University of Nevada, Reno

# LARGE-SCALE MULTI-AGENT DECISION-MAKING USING MEAN FIELD GAME THEORY AND REINFORCEMENT LEARNING

A Dissertation Submitted in Partial Fulfillment

of the Requirements for the Degree of Doctor of Philosophy in

Electrical Engineering

by

Zejian Zhou

Dr. Hao Xu / Dissertation Advisor

May 2021

THE GRADUATE SCHOOL

We recommend that the dissertation
prepared under our supervision by

**Zejian Zhou**

entitled

**Large-scale Multi-agent Decision-making Using Mean Field Game Theory and Reinforcement Learning**

be accepted in partial fulfillment of the
requirements for the degree of

**DOCTOR OF PHILOSOPHY**

Hao Xu, Ph.D.
*Advisor*

M. S. Fadali, Ph.D.
*Committee Member*

Xiaoshan Zhu, Ph.D.
*Committee Member*

Yantao Shen, Ph.D.
*Committee Member*

Hung (Jim) La, Ph.D.
*Graduate School Representative*

David W. Zeh, Ph.D., Dean
*Graduate School*

May, 2021

## ABSTRACT

The Multi-agent system (MAS) optimal control problem is a recently emerging research topic that benefits industries such as robotics, communication, and power systems. The traditional MAS control algorithms are developed by extending the single agent optimal controllers, requiring heavy information exchange. Moreover, the information exchanged within the MAS needs to be used to compute the optimal control resulting in the coupling between the computational complexity and the agent number. With the increasing need for large-scale MAS in practical applications, the existing MAS optimal control algorithms suffer from the "curse of dimensionality" problem and limited communication resources. Therefore, a new type of MAS optimal control framework that features a decentralized and computational friendly decision process is desperately needed. To deal with the aforementioned problems, the mean field game theory is introduced to generate a decentralized optimal control framework named the Actor-critic-mass (ACM). Moreover, the ACM algorithm is improved by eliminating constraints such as homogeneous agents and cost functions. Finally, the ACM algorithm is utilized in two applications.

## ACKNOWLEDGEMENTS

I would like to express my deep thanks to my advisor, Dr. Hao Xu for his valuable guidance, patience and support over the last four years. I also would like to thank Dr. Hung (Jim) La, Dr. M. S. Fadali, Dr. Xiaoshan Zhu, and Dr. Yantao Shen for serving on my doctoral committee.

I am greatly thankful to my parents, Ping Zhou and Fengmei Ye, my cousin Qin Liu, and the rest of my family for their love and support. Their encouragement and understanding over the course of my education has truly been a blessing. I also would like to thank all my fellow colleagues who made my time in Ph.D. program fun and exciting.

## TABLE OF CONTENTS

## LIST OF FIGURES

# CHAPTER 1
# INTRODUCTION

Pursuing a strategy that automatically generates the optimal decisions in various tasks has become one of the main pillars in artificial intelligence (AI) studies. As a general representation of different tasks, the intelligent agent-based system has been serving as the tool to study the theory of AI decision-making [80]. In the past decades, the intelligent agent-based system with only a single agent has incubated well-known reinforcement learning algorithms such as the approximate dynamic programming (ADP) [51], the deep Q network (DQN) [68], and the policy gradient descent algorithms [31] etc. However, with the development of the urgent need for a system with multiple agents [69], the single agent algorithms have been modified to adapt to the multi-agent systems (MAS) (see the survey paper [25]). The limit of such naive modifications has challenged these algorithms' future applications because the modern MAS contains more and more agents (see the survey paper [86]). The modern MAS is now named the large-scale MAS.

Due to enormous diversity gain from a larger population, large-scale multi-agent systems have recently attracted growing interests from academic research societies as well as industrial companies [103]. However, different than traditional optimal control methods [37], massive MAS optimal control is defined as a collective motion of a vast number of individuals such as schooling of fish and swarming of bacteria. There are two significant challenges, i.e., 1) how to break the well known "curse of dimensionality" [7] while optimizing tracking performance, and 2) how to share information under complex and uncertain environment, e.g., limited communication ability [24], uncertain wind and so on. Most recent multi-agent control algorithms [79] have stringent constraints, i.e., distributed agents

must share information with neighbors timely and accurately. However, this requirement could not be guaranteed in massive MAS optimal control since a complex and uncertain environment cannot support high-quality communication (e.g., less delay and packet dropouts) especially while the number of agents goes to infinity. This challenge has been recognized as the notorious **"Curse of Dimensionality"** problem [7].

To overcome the "Curse of dimensionality" problem, a decentralized solution is much more preferred since less communication is needed. Recently, A new type of decentralized multi-agent decision-making algorithm, named Mean Field Game (MFG), has been developed by Lasry and Lions [48], [34] under stochastic noncooperative games theory. A parallel work of MFG (or Nash Certainty Equivalence) has been provided by Huang et al. [39] independently. It has been applied in different areas successfully [23], [85].

The key idea of Mean Field Control Theory is to design decentralized control only based on local information and impact from mass, i.e., the entire other agents. [48] and [34] have proven that the individual agent's influence will be replaced by mass influence while the population of agents goes to infinity. When each agent plays the best response to the environment and population distribution, the solution of Mean Field Controller converges to $\varepsilon_N$-Nash equilibrium [39]. To obtain Mean Field optimal control solution, we need to minimize cost function as well as consider the practical influence from other agents that has been described as a probability density function (PDF) [34], [48]. Recall that for optimal control [53], the optimal cost function attained through solving Hamilton-Jacobi-Bellman (HJB) equation which is usually solved backward-in-time. Meanwhile, the massive multi-agent population distribution can also be represented through a new

type of partial differential equation, Fokker-Planck-Kolmogorov (FPK) equation, introduced by [34]. The FPK equation, however, must be solved forward-in-time.

In this chapter, we will elaborate in details to introduce the optimal control, reinforcement learning, and the motivation of introducing mean field games.

## 1.1 Optimal Control Problem and Reinforcement Learning Algorithms

To seek the automatic decision process that consumes less energy, the optimal control and decision-making methods have been studied for decades [78]. Although the optimal control problems and decision-making problems sometimes have different meanings, the control problems usually target the lower-level control adjustment of devices, while decision-making problems target solving decision-level tasks. In this dissertation, we refer to both as the same type of problems, i.e., finding the optimal action/control for a given task. An optimal control task is often defined in a given state and action space. Depending the type of the problem, an equation that describes the physics of the agent may be given. Finally, a cost function is defined to evaluate the behavior of the agent. The goal is to design an algorithm to automatically select the optimal control/action such that:

- The cost function is minimized.
- The physical system remains stable.

These two rules will be used to evaluate the different versions of the ACM algorithm through this dissertation.

Due to the high computational complexity, traditional optimizers [78] are not suitable in real-time systems. The dynamic programming method was developed by Richard Bellman [7] to break the computational intensive optimization problem into simpler sub-problems. The dynamic programming method requires recursively solving the Bellman equation in each computation step. And the resulting solution set is a sequence of parameters that minimize the cost function. Inspired by Bellman's work, researchers introduced this recursive optimizer into the optimal control problem and generated the reinforcement learning algorithm [47]. In reinforcement learning, the cost function is used to generate the Bellman equation since it is the objective function to minimize. Consequently, the parameters of the cost function, which are states and actions, become the subject of adjustment. Intuitively, the parameters are "reinforced" based on the stimuli (reward or punishment) received from the cost function.

A major development of reinforcement learning happened when a scheme called the Actor-critic is introduced by Andrew G. Barto et al. [6]. In the Actor-critic scheme, the actor applies actions, and the critic assesses the actions applied. The importance of this scheme is that it clearly describes the two-step frames (policy evaluation and policy improvement), which are now used in most modern reinforcement learning algorithms. Barto's work was adopted by both control and computer science societies and inspired the two leading branches of reinforcement learning. Both branches, however, propose to use neural networks in the Actor-critic framework. The major development of the reinforcement learning algorithm in computer science was contributed by Volodymyr Mnih et al. [68], where deep neural networks are introduced to approximate the optimal cost function. This algorithm has many developments such as double Q-learning [36], Deep Deterministic Policy Gradient (DDPG) [59], etc. On the other hand, Frank Lewis et al.

[50] introduced the approximate dynamic programming (ADP) based on optimal control problems. Although describing similar processes, the ADP allows rigorous derivations and Lyapunov stability analysis to provide a more reliable performance guarantee. Therefore, in this dissertation, the development of the ACM algorithm will follow Lewis's approach. However, it is worth noting that the ACM framework can also be easily transformed into the computer science's style.

In [2], the authors derived the Hamilton-Jacobi-Bellman (HJB) equation from the cost function. Moreover, the authors proposed to use two neural networks that fulfill the Actor-critic structure, i.e., the actor neural network to approximate the optimal control and the critic neural network to approximate the optimal cost function.

## 1.2 Multi-agent Reinforcement Learning and the Motivation of Mean Field Games

The multi-agent systems (MAS) with large scale agent population have benefited various industries, i.e., IoT [43], robotics [57], communication [83], etc. In the MAS control problem, all agents' real-time information in the team is required for coordination. To accomplish more complex tasks, the MAS's size has been brought to a very large-scale setup. However, it brings challenges.

The first is the communication difficulty that occurs because a very large-scale low latency communication network is required for the individual agent to exchange real-time information. For instance, the state-of-art consensus algorithms [72, 106] require a communication graph to be connected for ensuring that all

agents can share information. Similarly, in multi-agent reinforcement learning [69, 111], and the multi-agent optimal control [110], the requirement of the information from the other agents are stringent since it is often and nearly impossible to build and maintain a very large-scale communication mobile network, especially in uncertain environments [83, 29, 100]. Many types of research have been focusing on solving the unreliable communication issue in MAS games to deal with this issue. In [54], the authors developed two observers to estimate the state information of the followers with input delays caused by an unreliable communication network. Some researchers have found ways to reduce the communication effort among agents by transferring the all-time transmitting scenario into an on-demand communication scenario, e.g., the event-triggered consensus algorithm [27, 53]. Others tried to compress the information and set up multiple hop transmission via various advanced information flow techniques described in [12]. Despite the effort made to reduce communication traffic, the challenge of building and maintaining a massive network remains as the agent number goes to infinity. Therefore, a decentralized solution that requires no communication is much more preferred.

Secondly, traditional MAS algorithms suffered from the well-known "Curse of Dimensionality" problem since they demand each agent to include all other agents' states to compute the optimal strategy. Traditional reinforcement algorithms, like Q-learning, want to estimate each possible discrete state or state-action pairs' values for every other agent, and thus the complexity grows along with the increasing number of agents. Eventually, this increase leads directly to an exponential rise in the computational complexity [69, 10]. To tackle this challenge, the authors in [107] proposed the mean field reinforcement learning method which replaced the information of other agents by averaging the actions. Although this effectively solves the "Curse of Dimensionality" difficulty, acquiring the other

agents' real-time actions still requires accurate observations or reliable communication capability. Therefore, a **decentralized** MAS algorithm that **1) does not require observation or communication, 2) encodes the information of other agents to reduce the dimension** is desperately needed.

Recently, the mean-field game (MFG) theory is emerging to solve the multi-agent noncooperative games with infinity agents [34, 120]. The key idea of the mean-field game is to use a probability density function (PDF), named **"mass"**, among all agents' states to replace the required other agents' information in traditional non-cooperative games. Since the mass function's output has the same dimension with the state space, the computational complexity to compute optimal control is no longer related to the agent number. Therefore, the "curse of dimensionality" is tackled. Intuitively, the infinity players' non-cooperative game has been shifted into a two-player game for each agent, i.e., the agent himself versus the whole population (mass). Moreover, the mass is computed by a PDE named the Fokker-Planck-Kolmogorov (FPK), which is solved using local information only [34]. It is worth noting that all agents are assumed to be homogeneous in MFG and have the same objective function, so the time evolution of the global state PDF (mass) of all agents can be approximated by applying local control policy [34]. Therefore, no information exchange is required for an agent in the MAS compared to the aforementioned algorithms. To this end, the two major concerns for large scale MAS optimal tracking control is promising to be solved by formulating as an MFG. However, another major complication arises as the optimal tracking control brings another PDE, making the two coupled PDEs nearly impossible to solve in real-time.

Recall the optimal control and reinforcement learning [52, 92, 49], each agent's

objective is to minimize a given evaluation function by selecting the optimal control policy while considering the effect from other agents [94]. The optimal evaluation function, which is the evaluation function corresponding to the optimal control policy, can be computed by solving a PDE named Hamilton-Jacobian-Bellman (HJB) equation [92, 61, 22]. The HJB equation is solved reversely in the time since it comes from the Bellman equation [52] while the FPK is solved forward in time [34]. Moreover, both FPK and HJB are high dimensional nonlinear PDEs, which are also coupled by the optimal evaluation function and mass. Some researchers have provided offline empirical approximations such as [4, 15]. However, an online algorithm to solve the HJB-FPK needed in real applications is still lacking.

Inspired by the well-known actor-critic algorithm [92], which can online solve the multi-agent optimal control and reinforcement learning problem [94], we developed the actor-critic-mass (ACM) structure to solve the mean field game type optimal control problem. The ACM algorithm includes three neural networks, i.e., 1) the actor NN which approximates the optimal control, 2) the critic NN which approximates the optimal evaluation function by estimating the solution of the HJB equation, 3) the mass NN, which approximates the FPK equation's solution (mass).

## 1.3 Contributions of the Dissertation

In this dissertation, the large-scale multi-agent optimal control is discussed. The main contributions can be summarized as:

1. A decentralized algorithm that utilizes the mean field game theory has been

developed and named as the actor-critic-mass (ACM) algorithm. The developed algorithm can effectively reduce the computational complexity and communication burden in a MAS whose agent size goes to infinity.

2. The original mean field game theory's stringent assumptions on homogeneous agents has been relaxed. The ACM-Opponent algorithm has been designed to deal with agents with different physical dynamics in pursuit-evasion games.
3. The ACM algorithms have been applied to practical problems such as the electrical vehicle charging and communication optimal power allocation.

## 1.4 Organization of the Dissertation

In Chapter 2, the general ACM framework and theoretical foundations are introduced. Specifically, the large-scale MAS optimal control problem is formulated into the mean field type optimal control. Besides the Hamiltonian-Jacobi-Bellman (HJB) equation that is derived from the single agent optimal control scheme, an extra partial differential equation (PDE) named the Fokker-Plank-Kolmogorov (FPK) equation is introduced. To solve the coupled PDE system, referred to as the mean field equation system in this dissertation, a numerical approximation algorithm that includes three neural networks is introduced. Specifically, the actor, critic, and mass neural networks are designed to approximate the optimal control, optimal cost function, and the PDF of all agents' states, respectively.

The general ACM framework suffers from stringent constraints due to some unrealistic assumptions of the original mean field game theory. To release those constraints, the mean field game theory is improved to adapt to practical applica-

tions. In Chapter 3, the assumption that requires all agents to be homogeneous is lifted by considering the pursuer and evader agents with different system dynamics.

In Chapter 4, a case study based on an optimal electrical vehicle charging schedule is designed to verify the ACM algorithm's practical performance.

In Chapter 5, the proposed ACM framework is applied to optimal transmission power control in wireless networks to illustrate the practicability.

CHAPTER 2

# LARGE-SCALE MULTI-AGENT REINFORCEMENT LEARNING ALGORITHM AND STABILITY ANALYSIS [114, 118]

## 2.1 Introduction

During the past decade, the multi-agent systems (MAS) has attracted considerable amount of attention in complicated missions [113], [53]. Especially with the rapid development of game theory [87], the complex task-oriented decision making and control policy for MAS has been effectively investigated with profound mathematical support. Upon the traditional optimal control methods [52, 50], the differential game theory [93] has been thoroughly studied and applied to MAS by modeling the MAS control problem as a multi-player game. All players' states are integrated into one evaluation equation and maintained by each individual agent. To update that evaluation equation, the information is required to be exchanged among different agents. However, recall to latest researches [28], such multi-player game theory can effectively address the control problem in MAS with a small number of agents but not a very large-scale multi-agent system. It is because the increasing number of agents challenge the traditional MAS control algorithms in two aspects, i.e., 1) the difficulty of obtaining real-time information from all other agents due to unreliable wireless communication, 2) drastically increased dimension of the partial differential equations in optimal control methods due to the increasing augmented state dimension. Therefore, a new type of multi-agent control algorithm that is specially designed for a large size agent group is desperately needed.

In most MAS control and optimization researches [122], e.g., the consensus al-

gorithm [77], a reliable communication network is necessary for maintaining the basic information exchange among distributed agents, e.g., agent's states, control policies, etc. However, in large-scale multi-agent systems, building a reliable communication network is challenging and even impossible due to limited communication resources for the massive peer-to-peer wireless links [19, 112, 3]. Especially, the burden of large-scale communication network will significantly compromise the performance when the agent number goes to infinity. Some researchers are recently actively developing new methods to reduce the communication effort by introducing the on-demand style information exchange protocol. In [17, 53], the authors designed a new type of event-driven communication network by setting an appropriate threshold to trigger communication at the optimal time, which can effectively improve network usage efficiency. Other researchers have focused on the information reduction and further set up multi-hop transmission via efficient information flow design [74, 12]. Moreover, another type of algorithm focuses on solving the MAS control with less team information needed amid the difficulty of information exchange among large-scale MAS [60]. Despite the extreme effort made for the communication reduction in large-scale MAS, the huge information exchanging load continues if the agent number is ultra large. Besides the communication problem, the complicity of the individual agent's high-dimension state space can also lead to the notorious "***Curse of Dimensionality***" challenge in MAS design. For instance, in a vast majority of MAS intelligent control algorithms that feature deep reinforcement learning [45, 11], the finite discrete state space and/or action space have been considered [73, 33]. The drawbacks in those algorithms are that the complexity of the algorithm is coupled with the agent number [119]. Therefore, the computational effort increases drastically in the large-scale multi-agent systems.

To solve the above difficulties, the game theory [34, 41, 119] has been adopted. In the Mean Field Game (MFG) theory, each agent can model all the other agents' influence as a probability density function (PDF) to avoid the dimension explosion. More importantly, obtaining the PDF requires no communication but solving the Fokker-Planck-Kolmogorov (FPK) equation [34] locally. By assuming that all agents are homogeneous and share the same objective [9], each agent can estimate other agents' actions by substituting all agents' states (in the form of PDF) into its policy. Then, these actions can be used to update the PDF via the FPK equation. Since all information is generated locally (except the agents' initial distribution [9]), the communication traffic load has been relieved significantly.

In this chapter, the optimal control for large-scale MAS is investigated with the tracking control problem as an example. Each agent in the large-scale MAS must follow a reference trajectory and interact with other agents in the team. To deal with the communication and computation problems, a novel decentralized algorithm, i.e., the Actor-critic-mass (ACM) algorithm, has been proposed. Specifically, three neural networks are designed to solve the previously unsolvable coupled PDEs, i.e., the critic NN for HJB equation, actor NN for optimal control, and mass NN for FPK equation.

The contributions can be summarized as:

1. The Actor-critic-mass (ACM) algorithm has been developed to tackle the computation explosion and communication difficulties in the large-scale multi-agent optimal control problem. The dimension of the problem is reduced by introducing a probability function density (PDF) function to replace the effect of the team. Moreover, the PDF can be solved locally via the FPK equation
2. Three neural networks are designed to approximate the optimal control on-

line. Rigorous theoretical analysis is given to guarantee the performance of the neural networks.

## 2.2 Problem Statement

The large-scale tracking control problem is firstly defined. Then, the mean field game theory is introduced so that the original problem is reformulated into the mean field type tracking formulation.

### 2.2.1 The Original Multi-agent Tracking Control Problem

Let $N$ mobile agents move in a map with $l$ dimension. The noised motion dynamic equations of each agent can be written as:

$$d\rho_i(t) = [f_y(\rho_i(t)) + g_y(\rho_i(t))u_i]\,dt + \sigma dw_i \tag{2.1}$$

where $f_y(\rho_i(t))$ and $g_y(\rho_i(t))$ represent the intrinsic dynamic equations, $u_i \in \mathbb{R}^l$ is the deterministic control input, $\rho_i(t)$ is a $l$-dimensional stochastic vector which represents the position of agent $i$, $w_i$ denotes a the Brownian noise term, and $\sigma$ is the noise's coefficient matrix. A reference trajectory is provided and is known to all agents prior to the mission, i.e., $x_d(t)$. The trajectory should be tracked by all members in the team.

Therefore, we set a cost function to evaluate each agent's performance. And the lower the cost is, the better the agent performs.

$$J_i(\rho_i, \rho^{(N)}, u_i) = \mathbb{E}\Big\{ \int_0^\infty \big[\|\rho_i - x_d\|_Q^2 + \|u_i\|_R^2 + \Phi'(\rho^{(N)}, \rho_i)\big]dt\Big\} \tag{2.2}$$

where $\rho^{(N)}$ denotes the augmented states of all agents, and $\Phi'(\rho^{(N)}, \rho_i)$ evaluates the team's effect on agent $i$. The dimension of (2.2) is coupled with the agent number $N$, which goes to infinity in the very large-scale MAS case. The growing dimension of the cost function would cause computational explosion. To tackle this difficulty, the Mean Field Games (MFG) is embedded in the next subsection.

### 2.2.2 MFG-based Tracking Control

Mean Field Games [48] are designed to solve large-scale non-cooperative games. In the mean field games, the **global** information is encoded into a PDF, which can be measured by a local PDE named the FPK equation. The computed PDF overcomes the difficulty of collecting information from all other agents as well as reduces the dimension of the optimal control problem.

All agents are considered to track a reference trajectory which is denoted by $x_d(t) \in \mathbb{R}^l$. Therefore the tracking error is defined as :

$$x_i(t) = \rho_i(t) - x_d(t) \tag{2.3}$$

where $x_i$ represents the tracking error. Moreover, we obtain the tracking error as:

$$\begin{aligned} dx_i &= d\rho_i - dx_d \\ &= [f_y(\rho_i) + g_y(\rho_i)u_i - \frac{dx_d}{dt}]dt + \sigma dw_i \\ &= [f_y(x_i + x_d) + g_y(x_i + x_d)u_i - \frac{dx_d}{dt}]dt + \sigma dw_i \\ &= [f(x_i) + g(x_i)u_i]dt + \sigma dw_i \end{aligned} \tag{2.4}$$

where $f(x_i) = f_y(x_i + x_d) - \frac{dx_d}{dt}$, and $g(x_i) = g_y(x_i + x_d)$.

Similar to [39], in order to optimize the massive MAS performance, the cost

function (2.2) can be formalized as

$$J_i(x_i, m, u_i) = \mathbb{E}\Big\{ \int_0^\infty \big[L(x_i, u_i) + \Phi(m, x_i)\big] dt \Big\} \tag{2.5}$$

where the $m(x_i, t)$ denotes the probability density function that is used to replace $\rho^{(N)}$, $\Phi(m, x_i)$ is the corresponding new coupling function, $L(x_i, u_i) = \|x_i\|_Q^2 + \|u_i\|_R^2$.

Recall the Bellman equation [52]:

$$J_i^*(x_i, u_i^*(t)) = \min_{u_i} \{r(x_i, u_i(t)) + J_i^*(x_i, u_i(t + dt)\} \tag{2.6}$$

where $r(x_i, u_i(t)) = \|x_i\|_Q^2 + \|u_i(t)\|_R^2 + \Phi(m, x_i)$, and $u_i^*$ represents $i$-th agent's optimal control input.

According to the optimal control theory [52], we select the Hamiltonian as

$$H\left[x_i, \partial_x J_i(x_i, m, u_i)\right] = L(x_i, u_i) + \partial_x J_i(x_i, m, u_i)^T \left[f(x_i) + g(x_i)u_i\right] \tag{2.7}$$

Next, the HJB equation is obtained by substituting the Hamiltonian into the Bellman equation, i.e.,

$$\Phi(m, x_i) = -\partial_t J_i^*(x_i, m, u_i^*) - 0.5\sigma^2 \Delta J_i^*(x_i, m, u_i^*) + H\left[x_i, \partial_x J_i^*(x_i, m, u_i^*)\right] \tag{2.8}$$

Moreover, the optimal control can be derived as

$$u_i^*(x) = -\frac{1}{2} R^{-1} g^T(x_i) \partial_x J_i^*(x_i, m, t) \tag{2.9}$$

In the formulated game, the optimal control is the Nash equilibrium [8], i.e.

$$J_i(u_i; u_o^*) \geq J_i(u_i^*; u_o^*) \tag{2.10}$$

where $u_o^*$ denotes all other agents' control.

To calculate the optimal tracking control for individual agents in the large-scale MAS team, the PDF, i.e., $m(x_i, t)$ is needed. In the MFG [48, 34], this PDF can be measured through the locally solvable Fokker-Plank-Kolmogorov (FPK) equation.

$$\partial_t m(x_i, t) - \frac{\sigma^2}{2}\Delta m(x_i, t) - \operatorname{div}\left\{m D_p H\left[x_i, \partial_x J_i^*(x_i, m, t)\right]\right\} = 0 \tag{2.11}$$

The solution that satisfies the HJB and FPK equation at the same time is guaranteed to be unique [34], [48], [39]). Specifically, the solution is called the $\varepsilon_N$-Nash equilibrium [48]:

$$J_i(u_i; u_o^*) \geq J_i(u_i^*; u_o^*) - \varepsilon_N \tag{2.12}$$

where $\varepsilon_N$ is a constant and $\lim_{N\to\infty} \varepsilon_N = 0$.

**Remark 1.** In order to find the $\varepsilon_N$-Nash equilibrium, the HJB-FPK equation system has to be solved first. However, these two equations are coupled high-dimensional partial differential equations (PDEs) which are challenging to solve online. Hence, the reinforcement learning method [52] is introduced and extended to solve this problem.

## 2.3 Actor-critic-Mass Decentralized Algorithm

In this section, the actor-critic-mass algorithm is proposed. The main idea includes three coupled neural network estimators.

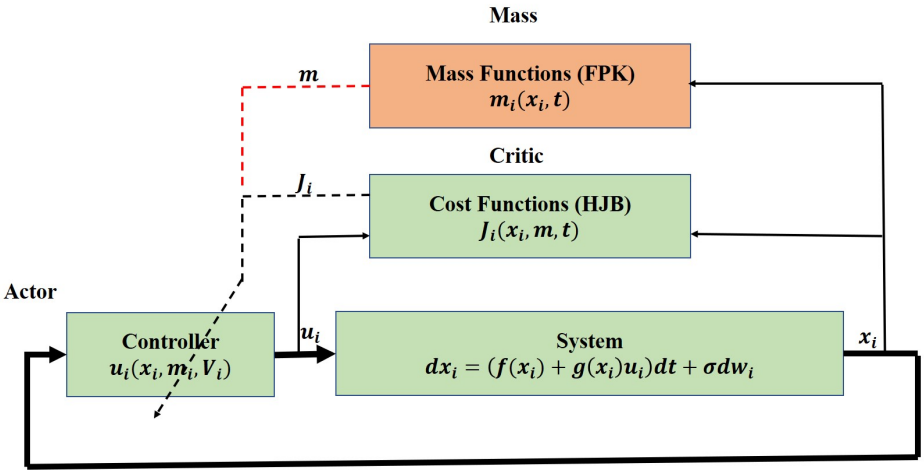



Figure 2.1: The proposed reinforcement learning structure

### 2.3.1 The Actor-critic-mass Estimator

To calculate the optimal control for each agent, the mean field equations (2.8), (2.11) needs to be solved. In reinforcement learning [52], two neural networks are designed to approximate the optimal solution. However, in the large-scale multi-agent system case, the PDF is also required to be approximated. Therefore, the proposed algorithm includes a new neural network, i.e., the mass NN.

Let $W_{J,i}^T$, $W_{u,i}^T$, and $W_{m,i}^T$ denote the weights vectors of the designed three neural networks, the critic, actor, and mass neural networks are

$$\begin{cases} J_i^*(x_i, m, t) = W_{J,i}^T \phi_{J,i}(x_i, m, t) + \varepsilon_{J,i} \\ u_i^*(x_i, m, t) = W_{u,i}^T \phi_{u,i}(x_i, m, t) + \varepsilon_{u,i} \\ m(x_i, t) = W_{m,i}^T \phi_{m,i}(x_i, \bar{m}_i, t) + \varepsilon_{m,i} \end{cases} \tag{2.13}$$

where $\bar{m}_i$ is defined as $\bar{m}_i(t) = \frac{1}{\hat{t}} \int_{[t-\hat{t}]_+}^{t} \mathbb{E}[m(x_i, \tau)] d\tau$, and $\hat{t}$ is a constant historical window, $\phi(\cdot)$ are activation functions, and $\varepsilon$ are the reconstruction errors.

Since the actual value of the weights are unknown, they have to be approximated. Let $\hat{W}_{J,i}^T$, $\hat{W}_{u,i}^T$, and $\hat{W}_{m,i}^T$ denote the approximated weights, the approxi-

mated optimal cost function, control and PDF are

$$\begin{cases} \hat{J}_i(x_i, \hat{m}_i, t) = \hat{W}_{J,i}^T(t)\phi_{J,i}(x_i, \hat{m}_i, t) \\ \hat{u}_i(x_i, \hat{m}_i, t) = \hat{W}_{u,i}^T(t)\phi_{u,i}(x_i, \hat{m}_i, t) \\ \hat{m}_i(x_i, \bar{m}_i, , t) = \hat{W}_{m,i}^T(t)\phi_{m,i}(x_i, \bar{m}_i, t) \end{cases} \tag{2.14}$$

When the approximated equations (2.14) are substituted into the mean field equations (2.8), (2.9) and (2.11), the generated errors are used to tune the neural networks, i.e.,

$$e_{HJBi} = \Phi(\hat{m}_i, x_i, t) + \hat{W}_{J,i}^T(t)(\partial_t \hat{\phi}_{J,i} + \frac{\sigma^2}{2}\Delta\hat{\phi}_{J,i} - \hat{H}_{WJ}) \tag{2.15}$$

$$e_{FPKi} = \hat{W}_{m,i}^T(t)\left[\partial_t \hat{\phi}_{m,i} - \frac{\sigma^2}{2}\Delta\hat{\phi}_{m,i} - \mathrm{div}(\hat{\phi}_{m,i}\,\mathrm{D_p}\,\hat{H})\right] \tag{2.16}$$

$$e_{ui} = \hat{W}_{u,i}^T(t)\phi_{u,i}(x_i, \hat{m}_i, t) + \frac{1}{2}R^{-1}g^T(x_i)\partial_x \hat{J}_i \tag{2.17}$$

where

$$\hat{\phi}_{J,i} = \phi_{J,i}(x_i, \hat{m}_i, t)$$

$$\hat{\phi}_{m,i} = \phi_{m,i}(x_i, \bar{m}_i, t)$$

$$\hat{\phi}_{u,i} = \phi_{u,i}(x_i, \hat{m}_i, t)$$

$$\hat{H}_{WJ} = H_{WJ}[x_i, \partial_x \hat{\phi}_{J,i}(x_i, \hat{m}_i, t)]$$

$$\hat{H} = H[x_i, \partial_x \hat{\phi}_{J,i}(x_i, \hat{m}_i, t)]$$

$\hat{H}_{WJ}$ satisfies

$$\hat{H}[x_i, \partial_x J_i(x_i, \hat{m}_i, t)] = \hat{W}_{J,i}^T(t)\hat{H}_{WJ}$$

Next, we simplify the notations of Eqs. 2.15 and 2.16

$$e_{HJBi} = \Phi(x_i, m, t) + \tilde{\Phi}_i(x_i, \tilde{m}_i, t) + \hat{W}_{J,i}^T(t)\Psi_{J,i}(x_i, \hat{m}_i, t) \tag{2.18}$$

$$e_{FPKi} = \hat{W}_m^T(t)\Psi_{m,i}(x_i, \bar{m}_i, \hat{J}_i, t) \tag{2.19}$$

with $\Psi_{J,i}(x_i, \hat{m}_i, t)$, and $\Psi_{m,i}(x_i, \bar{m}_i, \hat{J}_i, t)$ being defined as

$$\Psi_{J,i}(x_i, \hat{m}_i, t) = \partial_t \hat{\phi}_{J,i} + \frac{\sigma^2}{2}\Delta\hat{\phi}_{J,i} - H_{WJ}(x_i, \partial_x \hat{\phi}_{J,i}) \tag{2.20}$$

$$\Psi_{m,i}(x_i, \bar{m}_i, \hat{J}_i, t) = \partial_t \hat{\phi}_{m,i} - \frac{\sigma^2}{2}\Delta\hat{\phi}_{m,i} - \operatorname{div}\left\{\hat{\phi}_{m,i} D_p H\left[x_i, \partial_x \hat{J}(x_i, \hat{m}_i, t)\right]\right\} \tag{2.21}$$

and $\tilde{\Phi}_i(x_i, \tilde{m}_i, t)$ represents:

$$\tilde{\Phi}_i(x_i, \tilde{m}_i, t) = \Phi(x_i, \hat{m}_i, t) - \Phi(x_i, m, t) \tag{2.22}$$

Substitute 2.13 into 2.8 and 2.11, one obtains

$$\Phi(x_i, m, t) + W_{J,i}^T\{\partial_t \phi_{J,i} + \frac{\sigma^2}{2}\Delta\phi_{J,i}(x_i, m, t) - H_{WJ}\} + \varepsilon_{HJBi} = 0 \tag{2.23}$$

$$W_{m,i}^T\left\{\partial_t \phi_{m,i} - \frac{\sigma^2}{2}\Delta\phi_{m,i}(x_i, \bar{m}_i, t) - \operatorname{div}\left[\phi_{m,i} D_p H\left(x_i, \partial_x J(x_i, m, t)\right)\right]\right\} + \varepsilon_{FPKi} = 0 \tag{2.24}$$

Namely,

$$\Phi(x_i, m, t) + W_{J,i}^T \Psi_{J,i}(x_i, m, t) + \varepsilon_{HJBi} = 0 \tag{2.25}$$

$$W_{m,i}^T \Psi_{m,i}(x_i, \bar{m}_i, J_i, t) + \varepsilon_{FPKi} = 0 \tag{2.26}$$

where $\varepsilon_{HJBi}, \varepsilon_{FPKi}$ are the errors caused by neural networks' reconstruction errors. And they can be ignored when the reconstruction errors are negligible.

Substitute Eq,. 2.25 and 2.26 into Eqs. 2.18 and 2.19, one obtains the affect of the reconstruction errors on the neural networks' estimation errors.

$$\Phi(x_i, \tilde{m}_i, t) - \tilde{W}_{J,i}^T \Psi_{J,i}(x_i, \hat{m}_i, t) - W_{J,i}^T \tilde{\Psi}_{J,i}(x_i, \tilde{m}_i, t) - \varepsilon_{HJBi} = e_{HJBi} \tag{2.27}$$

$$- \tilde{W}_{m,i}^T \Psi_{m,i}(x_i, \bar{m}_i, \hat{J}_i, t) - W_{m,i}^T \tilde{\Psi}_{m,i}(x_i, \bar{m}_i, \tilde{J}_i, t) - \varepsilon_{FPKi} = e_{FPKi} \tag{2.28}$$

Similarly,

$$\begin{aligned} &- \tilde{W}_{u,i}^T \phi_{u,i}(x_i, \hat{m}_i, t) - W_{u,i}^T \tilde{\phi}_{u,i}(x_i, \tilde{m}_i, t) \\ &- \frac{1}{2} R^{-1} g^T(x_i) \partial_x \tilde{J}_i - \varepsilon_{ui} = e_{ui} \end{aligned} \tag{2.29}$$

with

$$\varepsilon_{ui} = \varepsilon_{mi} + \frac{1}{2}R^{-1}g^T(x_i)\partial_x\varepsilon_{HJBi}$$

$$\tilde{W}_{J,i} = W_{J,i} - \hat{W}_{J,i}(t)$$

$$\tilde{\Psi}_{J,i}(x_i, \tilde{m}_i, t) = \Psi_{J,i}(x_i, m, t) - \Psi_{J,i}(x_i, \hat{m}_i, t)$$

$$\tilde{\Psi}_{m,i}(x_i, \bar{m}_i, \tilde{J}_i, t) = \Psi_{m,i}(x_i, \bar{m}, J_i, t) - \Psi_{m,i}(x_i, \bar{m}_i, \hat{J}_i, t)$$

$$\tilde{\phi}_{u,i}(x_i, \tilde{m}_i, t) = \phi_{u,i}(x_i, m, t) - \phi_{u,i}(x_i, \hat{m}_i, t)$$

Apply the gradient descent algorithm on the neural networks' estimation errors, the update laws are

$$\dot{\hat{W}}_{J,i}(t) = -\alpha_{h,i}\frac{\Psi_{J,i}(x_i, \hat{m}_i, t)e_{HJBi}^T}{1 + \Psi_{J,i}^T(x_i, \hat{m}_i, t)\Psi_{J,i}(x_i, \hat{m}_i, t)} \tag{2.30}$$

$$\dot{\hat{W}}_{m,i}(t) = -\alpha_{m,i}\frac{\Psi_{m,i}(x_i, \bar{m}_i, \hat{J}_i, t)e_{FPKi}^T}{1 + \Psi_{m,i}^T(x_i, \bar{m}_i, \hat{J}_i, t)\Psi_{m,i}(x_i, \bar{m}_i, \hat{J}_i, t)} \tag{2.31}$$

$$\dot{\hat{W}}_{u,i}(t) = -\alpha_{u,i}\frac{\phi_{u,i}(x_i, \hat{m}_i, t)e_{ui}^T}{1 + \phi_{u,i}^T(x_i, \hat{m}_i, t)\phi_{u,i}(x_i, \hat{m}_i, t)} \tag{2.32}$$

where $\alpha_{h,i}$, $\alpha_{m,i}$, and $\alpha_{u,i}$ are the learning rates.

### 2.3.2 The Actor-critic-mass Algorithm Performance Analysis

Recall to (2.30), (2.31), and (2.32), the Actor-critic-mass NN weights estimation error can be represented as:

$$\dot{\tilde{W}}_{J,i}(t) = -\dot{\hat{W}}_{J,i}(t) = \alpha_{h,i}\frac{\Psi_{J,i}(x_i, \hat{m}_i, t)e_{HJBi}^T}{1 + \Psi_{J,i}^T(x_i, \hat{m}_i, t)\Psi_{J,i}(x_i, \hat{m}_i, t)} \tag{2.33}$$

$$\dot{\tilde{W}}_{m,i}(t) = -\dot{\hat{W}}_{m,i}(t) = \alpha_{m,i}\frac{\Psi_{m,i}(x_i, \bar{m}_i, \hat{J}_i, t)e_{FPKi}^T}{1 + \Psi_{m,i}^T(x_i, \bar{m}_i, \hat{J}_i, t)\Psi_{m,i}(x_i, \bar{m}_i, \hat{J}_i, t)} \tag{2.34}$$

$$\dot{\tilde{W}}_{u,i}(t) = -\dot{\hat{W}}_{u,i}(t) = \alpha_{u,i}\frac{\phi_{u,i}(x_i, \hat{m}_i, t)e_{ui}^T}{1 + \phi_{u,i}^T(x_i, \hat{m}_i, t)\phi_{u,i}(x_i, \hat{m}_i, t)} \tag{2.35}$$

Next, the bound of the approximated optimal cost function, PDF of all tracking errors, optimal control, the bound will depend on the NNs' reconstruction errors and the weights' approximation errors, i.e.,

**Cost function estimation error:**

$$\begin{aligned}
\|\tilde{J}_i(t)\| &= \|\tilde{W}_{J,i}^T(t)\phi_{J,i}(x_i,\hat{m}_i,t) + W_{J,i}^T\tilde{\phi}_{J,i}(x_i,\tilde{m}_i,t) + \varepsilon_{J,i}\| \\
&\leq \|\tilde{W}_{J,i}^T(t)\|\|\hat{\phi}_{J,i}\| + l_{\phi_{J,i}}\|W_{J,i}\|\|\tilde{m}_i\| + \|\varepsilon_{J,i}\| \\
&\leq b_{W_{v,i}}(t)\|\hat{\phi}_{J,i}\| + l_{\phi_{J,i}}\|W_{J,i}\|b_{m,i}(t) + \|\varepsilon_{J,i}\| \\
&\equiv b_{J,i}(t)
\end{aligned} \tag{2.36}$$

, where $\tilde{\phi}_{J,i}(x_i,\tilde{m}_i,t) = \phi_{J,i}(x_i,m,t) - \phi_{J,i}(x_i,\hat{m}_i,t)$

**Mass function estimation error:**

$$\begin{aligned}
\|\tilde{m}_i(t)\| &= \|\tilde{W}_{m,i}^T(t)\phi_{m.i}(x_i,\bar{m}_i,t) + \varepsilon_{m,i}|| \\
&\leq \|\tilde{W}_{m,i}\|\|\phi_{m,i}\| + \|\varepsilon_{m,i}\| \leq b_{W_{m,i}}(t)\|\phi_{m,i}\| + \|\varepsilon_{m,i}\| \\
&\equiv b_{m,i}(t)
\end{aligned} \tag{2.37}$$

**Optimal Mean Field type control estimation error:**

$$\begin{aligned}
\|\tilde{u}_i(t)\| &= \|\tilde{W}_{u,i}^T(t)\hat{\phi}_{u,i} + W_{u,i}^T(t)\tilde{\phi}_{u,i}(x_i,\tilde{m}_i,t)\varepsilon_{u,i}|| \\
&\leq \|\tilde{W}_{u,i}\|\|\hat{\phi}_{u,i}\| + \|W_{u,i}\|\|\tilde{\phi}_{u,i}(x_i,\tilde{m}_i,t)\| + \|\varepsilon_{u,i}\| \\
&\leq b_{W_{u,i}}(t)\|\hat{\phi}_{u,i}\| + l_\phi\|W_{u,i}\|\|\tilde{m}_i\| + \|\varepsilon_{u,i}\| \\
&\equiv b_{u,i}(t)
\end{aligned} \tag{2.38}$$

The convergence and stability of neural networks and the closed-loop system can be described using the following theorems.

**Theorem 1.** *(Actor neural network convergence)*: Let the actor neural network weights be updated following the updated law (2.32), and let the learning rate $\alpha_{u,i}$ be a positive number, the actor neural network's weight approximation error $\tilde{W}_{u,i}$ and the the optimal control approximation error $\tilde{u}_i = u_i - \hat{u}_i$ are uniformly ultimately

bounded (UUB) in the stochastic sense. The bounds are negligible if the reconstruction error is ignored.

*Proof.* Considering the Lyapunov function candidate selected as

$$L_{u,i}(t) = \frac{1}{2} tr\Big\{\tilde{W}_{u,i}^T \tilde{W}_{u,i}\Big\} \tag{2.39}$$

Take the first derivative of selected Lyapunov function candidate and substitute actor NN weights estimation error dynamics given in Eq. 2.35, Eq. 2.39 can be represented as

$$\dot{L}_{u,i}(t) = tr\Big\{\tilde{W}_{u,i}^T \dot{\tilde{W}}_{u,i}\Big\} = \alpha_{u,i} tr\Big\{\tilde{W}_{u,i}^T \frac{\hat{\phi}_{u,i} e_{ui}^T}{1 + \hat{\phi}_{u,i}^T \phi_{u,i}}\Big\} \tag{2.40}$$

Next, substituting Eq.2.29 into Eq. 2.40, Eq. 2.40 can be expressed as

$$\begin{aligned}
\dot{L}_{u,i}(t) = &- \alpha_{u,i} tr\{\tilde{W}_{u,i}^T \frac{\hat{\phi}_{u,i}\phi_{u,i}^T}{1 + \hat{\phi}_{u,i}^T \phi_{u,i}} \tilde{W}_{u,i}\} \\
&- \alpha_{u,i} tr\{\tilde{W}_{u,i}^T \frac{\hat{\phi}_{u,i}\tilde{\phi}_{u,i}^T}{1 + \hat{\phi}_{u,i}^T \phi_{u,i}} W_{u,i}\} \\
&- \alpha_{u,i} tr\{\tilde{W}_{u,i}^T \frac{\hat{\phi}_{u,i}[\frac{1}{2} R^{-1} g^T(e_i) \partial_e \tilde{V}_i]}{1 + \hat{\phi}_{u,i}^T \phi_{u,i}}\} \\
&- \alpha_{u,i} tr\{\tilde{W}_{u,i}^T \frac{\hat{\phi}_{u,i}\varepsilon_{u,i}^T}{1 + \hat{\phi}_{u,i}^T \phi_{u,i}}\Big\}
\end{aligned} \tag{2.41}$$

Let $b_{ui} = \frac{\hat{\phi}_{u,i}}{1+\hat{\phi}_{u,i}^T \phi_{u,i}}$, the triangle inequality properties (e.g. Cauchy-Schwarz inequality etc.) are applied for simplifying Eq. 2.41 as

$$\begin{aligned}
&\dot{L}_{u,i}(t) = -\alpha_{u,i} tr\{\tilde{W}_{u,i}^T b_{ui} \phi_{u,i}^T \tilde{W}_{u,i}\} \\
&- \alpha_{u,i} tr\{\tilde{W}_{u,i}^T b_{ui} \tilde{\phi}_{u,i}^T W_{u,i}\} - \alpha_{u,i} tr\{\tilde{W}_{u,i}^T b_{ui} \varepsilon_{ui}^T\} \\
&- \alpha_{u,i} tr\{\tilde{W}_{u,i}^T b_{ui} \frac{1}{2} R^{-1} g^T(e_i) \partial_e \tilde{V}_i\} \\
&\leq -\frac{\alpha_{u,i}}{4} \frac{\left\|\hat{\phi}_{u,i}\right\|^2}{1 + \left\|\hat{\phi}_{u,i}\right\|^2} \left\|\tilde{W}_{u,i}\right\|^2 - \frac{\alpha_{u,i}}{4} \frac{\left\|\phi_{u,i}\right\|^2}{1 + \left\|\phi_{u,i}\right\|^2} \left\|\tilde{W}_{u,i}\right\|^2
\end{aligned}$$

$$
\begin{aligned}
&- \alpha_{u,i} tr\{\tilde{W}_{u,i}^T \mathrm{b}_{ui} \tilde{\phi}_{u,i}^T W_{u,i}\} - \alpha_{u,i} \frac{\left\| W_{u,i}^T \tilde{\phi}_{u,i} \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \\
&+ \alpha_{u,i} \frac{\left\| W_{u,i}^T \tilde{\phi}_{u,i} \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} - \frac{\alpha_{u,i}}{4} \frac{\left\| \hat{\phi}_{u,i} \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \left\| \tilde{W}_{u,i} \right\|^2 \\
&- \alpha_{u,i} tr\{\tilde{W}_{u,i}^T \mathrm{b}_{1ui} \tilde{\phi}_{u,i}^T W_{u,i}\} - \alpha_{u,i} \frac{\left\| W_{u,i}^T \tilde{\phi}_{u,i} \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \\
&- \alpha_{u,i} tr\{\tilde{W}_{u,i}^T \frac{\hat{\phi}_{u,i} [\frac{1}{2} R^{-1} g^T(e_i) \partial_e \tilde{V}_i]}{1 + \hat{\phi}_{u,i}^T \phi_{u,i}}\} \\
&- \alpha_{u,i} \frac{\left\| \frac{1}{2} R^{-1} g^T(e_i) \partial_e \tilde{V}_i \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} + \alpha_{u,i} \frac{\left\| \frac{1}{2} R^{-1} g^T(e_i) \partial_e \tilde{V}_i \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \\
&- \frac{\alpha_{u,i}}{4} \frac{\left\| \hat{\phi}_{u,i} \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \left\| \tilde{W}_{u,i} \right\|^2 - \alpha_{u,i} tr\{\tilde{W}_{u,i}^T b_{ui} \varepsilon_{ui}^T\} \\
&- \alpha_{u,i} \frac{\left\| \varepsilon_{ui} \right\|^2}{1 + \hat{\phi}_{u,i}^T \phi_{u,i}} + \alpha_{u,i} \frac{\left\| \varepsilon_{ui} \right\|^2}{1 + \hat{\phi}_{u,i}^T \phi_{u,i}}
\end{aligned} \tag{2.42}
$$

Moreover, according to the $-\frac{1}{4}a^2 \pm ab - b^2 = -\left(\frac{1}{2}a \mp b\right)^2$, Eq. 2.42 can be represented as

$$
\begin{aligned}
\dot{L}_{u,i}(t) \leq & - \frac{\alpha_{u,i}}{4} \frac{\left\| \hat{\phi}_{u,i} \right\|^2 \left\| \tilde{W}_{u,i} \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \\
& - \frac{\alpha_{u,i}}{\left\| 1 + \phi_{u,i} \right\|^2} \left\| \frac{\tilde{W}_{u,i}^T \hat{\phi}_{u,i}}{2} - W_{u,i}^T \tilde{\phi}_{u,i} \right\|^2 \\
& - \frac{\alpha_{u,i}}{\left\| 1 + \phi_{u,i} \right\|^2} \left\| \frac{\tilde{W}_{u,i}^T \hat{\phi}_{u,i}}{2} - \frac{1}{2} R^{-1} g^T(e_i) \partial_e \tilde{V} \right\|^2 \\
& - \frac{\alpha_{u,i}}{\left\| 1 + \phi_{u,i} \right\|^2} \left\| \frac{\tilde{W}_{u,i}^T \hat{\phi}_{u,i}}{2} - \varepsilon_{ui} \right\|^2 + \frac{\alpha_{u,i} R^{-1} g^T(e_i) \partial_e \tilde{V}}{4(1 + \left\| \hat{\phi}_{u,i} \right\|^2)}
\end{aligned}
$$

$$+\underbrace{\frac{\alpha_{u,i}\|\varepsilon_{ui}\|^2}{1+\left\|\hat{\phi}_{u,i}\right\|^2}}_{\varepsilon_{Nui}} \tag{2.43}$$

To simplify the derivation in Eq. 2.43, several negative terms are dropped such as

$$\begin{aligned}
&\dot{L}_{u,i}(t) \leq \\
&-\frac{\alpha_{u,i}}{4}\frac{\left\|\hat{\phi}_{u,i}\right\|^2\left\|\tilde{W}_{u,i}\right\|^2}{1+\left\|\hat{\phi}_{u,i}\right\|^2}+\frac{\alpha_{u,i}R^{-1}g^T(e_i)\partial_e\tilde{V}}{4(1+\left\|\hat{\phi}_{u,i}\right\|^2)}+\varepsilon_{Nu,i} \\
&\leq -\frac{\alpha_{u,i}}{4}\frac{\left\|\hat{\phi}_{u,i}\right\|^2\left\|\tilde{W}_{u,i}\right\|^2}{1+\left\|\hat{\phi}_{u,i}\right\|^2}+B_{Wu,i}
\end{aligned} \tag{2.44}$$

where

$$B_{W_{u,i}}=\alpha_{u,i}\frac{\left\|R^{-1}g^T\left(e_i\right)\right\|^2\left\|\tilde{V}_i\right\|^2}{1+\left\|\hat{\phi}_{u,i}\right\|^2}+\varepsilon_{Nu,i}$$

and $\tilde{V}_i$ is the actor estimation error given in Eq. 2.36.

According to Lyapunov stability analysis and Eq. 2.44, the actor NN weight estimation error will be uniformly bounded with the bound given as

$$\left\|\tilde{W}_{ui}\right\| \leq \sqrt{\frac{4(1+\left\|\hat{\phi}_{ui}\right\|^2)}{\alpha_{u,i}\left\|\hat{\phi}_{ui}\right\|^2}B_{Wu,i}(t)} \equiv b_{Wu,i}(t) \tag{2.45}$$

This completes the proof. □

**Theorem 2.** *(Optimal cost function convergence)* Let the critic neural network weights be updated following the update law (2.30), and let the learning rate $\alpha_{h,i}$ be a positive number, the critic neural network's weight approximation error $\tilde{W}_{J,i}$ and the

cost function approximation error $\tilde{J}_i = J_i - \hat{J}_i$ are UUB in the stochastic sense. The bounds are negligible if the reconstruction error is ignored.

*Proof.* Consider the Lyapunov function candidate as

$$L_{J,i}(t) = \frac{1}{2} tr\Big\{\tilde{W}_{J,i}\tilde{W}_{J,i}\Big\} \tag{2.46}$$

According to the Lyapunov stability analysis method, take the first derivative of selected Lyapunov function candidate, we have

$$\dot{L}_{J,i}(t) = tr\{\tilde{W}_{J,i}^T\dot{\tilde{W}}_{J,i}\} \tag{2.47}$$

Substituting critic NN weights estimation error dynamics given in Eq. 2.33, Eq. 2.47 can be represented as

$$\dot{L}_{J,i}(t) = \alpha_{h,i} tr\Big\{\tilde{W}_{J,i}^T \frac{\Psi_{J,i}(e_i, \hat{m}_i, t)e_{HJBi}^T}{1 + \hat{\Psi}_{J,i}^T\Psi_{J,i}}\Big\} \tag{2.48}$$

Let $\hat{\Psi}_{J,i}$ denotes $\Psi_{J,i}(e_i, \hat{m}_i, t)$, and $\tilde{\Psi}_{J,i}$ denotes $\Psi_{J,i}(e_i, \tilde{m}_i, t)$. Substituting Eq.2.27 into Eq. 2.48, Eq. 2.48 can be expressed as

$$\begin{aligned}\dot{L}_{J,i}(t) =& \alpha_{h,i} tr\Big\{\tilde{W}_{J,i}^T \frac{\hat{\Psi}_{J,i}\tilde{\Phi}^T}{1 + \hat{\Psi}_{J,i}^T\Psi_{J,i}}\Big\} - \alpha_{h,i} tr\Big\{\tilde{W}_{J,i}^T \frac{\hat{\Psi}_{J,i}\Psi_{J,i}^T}{1 + \hat{\Psi}_{J,i}^T\Psi_{J,i}}\tilde{W}_{J,i}\Big\} \\ &- \alpha_{h,i} tr\Big\{\tilde{W}_{J,i}^T \frac{\hat{\Psi}_{J,i}\tilde{\Psi}_{J,i}^T}{1 + \hat{\Psi}_{J,i}^T\Psi_{J,i}}W_{J,i}\Big\} - \alpha_{h,i} tr\Big\{\tilde{W}_{J,i}^T \frac{\hat{\Psi}_{J,i}\varepsilon_{HJBi}^T}{1 + \hat{\Psi}_{J,i}^T\Psi_{J,i}}\Big\}\end{aligned} \tag{2.49}$$

Let $b_{1Vi} = \frac{\hat{\Psi}_{J,i}}{1+\hat{\Psi}_{J,i}^T\Psi_{J,i}}$, the triangle inequality properties (e.g. Cauchy-Schwarz inequality etc.) are applied for simplifying Eq. 2.49 as

$$\begin{aligned}&\dot{L}_{J,i}(t) \leq \alpha_{h,i} tr\{\tilde{W}_{J,i}^T b_{1Vi}\hat{\Phi}^T\} - \alpha_{h,i} tr\{\tilde{W}_{J,i}^T b_{1Vi}\Psi_{J,i}^T\tilde{W}_{J,i}\} \\ &- \alpha_{h,i} tr\{\tilde{W}_{J,i}^T \mathrm{b}_{1Vi}\tilde{\Psi}_{J,i}^T W_{J,i}\} - \alpha_{h,i} tr\{\tilde{W}_{J,i}^T \mathrm{b}_{1Vi}\varepsilon_{HJBi}^T\} \\ &\leq -\frac{\alpha_{h,i}}{4}\frac{\left\|\hat{\Psi}_{J,i}\right\|^2}{1 + \left\|\hat{\Psi}_{J,i}\right\|^2}\left\|\tilde{W}_{J,i}\right\|^2 - \frac{\alpha_{h,i}}{4}\frac{\left\|\hat{\Psi}_{J,i}\right\|^2}{1 + \left\|\hat{\Psi}_{J,i}\right\|^2}\left\|\tilde{W}_{J,i}\right\|^2\end{aligned}$$

$$
\begin{aligned}
&+\alpha_{h,i} tr\{\tilde{W}_{J,i}^T \mathrm{b}_{1Vi} \tilde{\Phi}^T\} - \alpha_{h,i} \frac{1}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} \left\|\tilde{\Phi}^T\right\|^2 \\
&+\alpha_{h,i} \frac{1}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} \left\|\tilde{\Phi}^T\right\|^2 - \frac{\alpha_{h,i}}{4} \frac{\left\|\hat{\Psi}_{J,i}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} \left\|\tilde{W}_{J,i}\right\|^2 \\
&- \alpha_{h,i} tr\{\tilde{W}_{J,i}^T \mathrm{b}_{1Vi} \tilde{\Psi}_{J,i}^T W_{J,i}\} - \alpha_{h,i} \frac{\left\|W_{J,i}^T \tilde{\Psi}_{J,i}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} \\
&+\alpha_{h,i} \frac{\left\|W_{J,i}^T \tilde{\Psi}_{J,i}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} - \frac{\alpha_{h,i}}{4} \frac{\left\|\hat{\Psi}_{J,i}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} \left\|\tilde{W}_{J,i}\right\|^2 \\
&- \alpha_{h,i} tr\{\tilde{W}_{J,i}^T b_{1Vi} \varepsilon_{HJBi}^T\} - \alpha_{h,i} \frac{\left\|\varepsilon_{HJBi}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} \\
&+\alpha_{h,i} \frac{1}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} \left\|\varepsilon_{HJBi}\right\|^2
\end{aligned} \tag{2.50}
$$

Moreover, according to the $-\frac{1}{4}a^2 \pm ab - b^2 = -\left(\frac{1}{2}a \mp b\right)^2$, Eq. 2.50 can be represented as

$$
\begin{aligned}
\dot{L}_{J,i}(t) \leq &-\frac{\alpha_{h,i}}{4} \frac{\left\|\hat{\Psi}_{J,i}\right\|^2 \left\|\tilde{W}_{J,i}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} - \frac{\alpha_{h,i} \left\|\frac{\tilde{W}_{J,i}^T \hat{\Psi}_{J,i}}{2} - \tilde{\Phi}\right\|^2}{\left\|1+\Psi_{J,i}\right\|^2} \\
&- \frac{\alpha_{h,i} \left\|\frac{\tilde{W}_{J,i}^T \hat{\Psi}_{J,i}}{2} - W_{J,i}^T \tilde{\Psi}_{J,i}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} - \alpha_{h,i} \frac{\left\|\frac{\tilde{W}_{J,i}^T \hat{\Psi}_{J,i}}{2} - \varepsilon_{HJBi}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} \\
&+ \frac{\alpha_{h,i} \left\|\tilde{\Phi}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} + \frac{\alpha_{h,i} \left\|W_{J,i}^T \tilde{\Psi}_{J,i}\right\|^2}{1+\left\|\Psi_{J,i}\right\|^2} + \underbrace{\frac{\alpha_{h,i} \left\|\varepsilon_{HJBi}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2}}_{\varepsilon_{VHJBi}}
\end{aligned} \tag{2.51}
$$

To simplify the derivation in Eq. 2.51, several negative terms are dropped such

as

$$\dot{L}_{J,i}(t) = -\frac{\alpha_{h,i}}{4}\frac{\left\|\hat{\Psi}_{J,i}\right\|^2\left\|\tilde{W}_{J,i}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} + \frac{\alpha_{h,i}\left\|\tilde{\Phi}\right\|^2}{1+\left\|\Psi_{J,i}\right\|^2}$$
$$+ \frac{\alpha_{h,i}\left\|W_{J,i}^T\tilde{\Psi}_{J,i}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} + \varepsilon_{VHJBi}$$
$$\leq -\frac{\alpha_{h,i}}{4}\frac{\left\|\hat{\Psi}_{J,i}\right\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2}\left\|\tilde{W}_{J,i}\right\|^2 + B_{WJ,i}(t) \tag{2.52}$$

where $B_{WJ,i}(t) = \frac{[l_{\Phi,i}+l_{\Psi_{J,i}}\left\|\tilde{W}_{J,i}\right\|^2]\|\tilde{m}_i\|^2}{1+\left\|\hat{\Psi}_{J,i}\right\|^2} + \varepsilon_{VHJBi}$, $l_{\Phi,i}$, $l_{\Psi_{J,i}}$ are Lipschitz constants of the coupling function $\Phi(\cdot)$ and $\Psi_{J,i}(\cdot)$ respectively, and $\tilde{m}_i$ is the mass estimation bound given in Eq. 2.37.

According to Lyapunov stability analysis and Eq. 2.52, the critic NN weight estimation error will be uniformly bounded with the bound given as

$$\left\|\tilde{W}_{J,i}\right\| \leq \sqrt{\frac{4(1+\left\|\hat{\Psi}_{J,i}\right\|^2)}{\alpha_{h,i}\left\|\hat{\Psi}_{J,i}\right\|^2}B_{WJ,i}(t)} \equiv b_{WJ,i}(t) \tag{2.53}$$

This completes the proof. □

**Theorem 3.** *(Mass neural network convergence)*: Let the mass neural network weights be updated following the updated law (2.31), and let the learning rate $\alpha_{m,i}$ be a positive number, the mass neural network's weight error $\tilde{W}_{m,i}$ and the PDF approximation error $\tilde{m}_i = m_i - \hat{m}_i$ are UUB in the stochastic sense. The bounds are negligible if the reconstruction error is ignored.

*Proof.* Considering the Lyapunov function candidate selected as

$$L_{m,i}(t) = \frac{1}{2}tr\left\{\tilde{W}_{m,i}^T\tilde{W}_{m,i}\right\} \tag{2.54}$$

Take the first derivative of selected Lyapunov function candidate and substitute mass NN weights estimation error dynamics given in Eq. 2.34, Eq. 2.54 can be represented as

$$\dot{L}_{m,i}(t) = tr\Big\{\tilde{W}_{m,i}^T \dot{\tilde{W}}_{m,i}\Big\} = \alpha_{m,i} tr\Big\{\tilde{W}_{m,i}^T \frac{\hat{\Psi}_{m,i} e_{FPKi}^T}{1+\hat{\Psi}_{m,i}^T \Psi_{m,i}}\Big\} \tag{2.55}$$

Next, substituting Eq.2.28 into Eq. 2.55, Eq. 2.55 can be expressed as

$$\begin{aligned}\dot{L}_{m,i}(t) = &- \alpha_{m,i} tr\{\tilde{W}_{m,i}^T \frac{\hat{\Psi}_{m,i}\Psi_{m,i}^T}{1+\hat{\Psi}_{m,i}^T\Psi_{m,i}} \tilde{W}_{m,i}\} \\ &- \alpha_{m,i} tr\{\tilde{W}_{m,i}^T \frac{\hat{\Psi}_{m,i}\tilde{\Psi}_{m,i}^T}{1+\hat{\Psi}_{m,i}^T\Psi_{m,i}} W_{m,i}\} \\ &- \alpha_{m,i} tr\{\tilde{W}_{m,i}^T \frac{\hat{\Psi}_{m,i}\varepsilon_{FPKi}^T}{1+\hat{\Psi}_{m,i}^T\Psi_{m,i}}\}\end{aligned} \tag{2.56}$$

Let $b_{mi} = \frac{\hat{\Psi}_{m,i}}{1+\hat{\Psi}_{m,i}^T\Psi_{m,i}}$, the triangle inequality properties (e.g. Cauchy-Schwarz inequality etc.) are applied for simplifying Eq. 2.56 as

$$\begin{aligned}&\dot{L}_{m,i}(t) = -\alpha_{m,i} tr\{\tilde{W}_{m,i}^T b_{mi} \Psi_{m,i}^T \tilde{W}_{m,i}\} \\ &- \alpha_{m,i} tr\{\tilde{W}_{m,i}^T b_{mi} \tilde{\Psi}_{m,i}^T W_{m,i}\} - \alpha_{m,i} tr\{\tilde{W}_{m,i}^T b_{mi} \varepsilon_{FPKi}^T\} \\ &\leq -\frac{\alpha_{m,i}}{2} \frac{\left\|\hat{\Psi}_{m,i}\right\|^2 \left\|\tilde{W}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2} - \frac{\alpha_{m,i}}{4} \frac{\|\Psi_{m,i}\|^2 \left\|\tilde{W}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2} \\ &- \alpha_{m,i} tr\{\tilde{W}_{m,i}^T b_{mi} \tilde{\Psi}_{m,i}^T W_{m,i}\} - \alpha_{m,i} \frac{\left\|W_{m,i}^T \tilde{\Psi}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2} \\ &+ \alpha_{m,i} \frac{\left\|W_{m,i}^T \tilde{\Psi}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2} - \frac{\alpha_{m,i}}{4} \frac{\left\|\hat{\Psi}_{m,i}\right\|^2 \left\|\tilde{W}_{m,i}\right\|^2}{1+\|\Psi_{m,i}\|^2} \\ &- \alpha_{m,i} tr\{\tilde{W}_{m,i}^T b_{mi} \varepsilon_{FPKi}^T\} - \frac{\alpha_{m,i}\|\varepsilon_{FPKi}\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2} \\ &+ \frac{\alpha_{m,i}\|\varepsilon_{FPKi}\|^2}{1+\|\Psi_{m,i}\|^2}\end{aligned} \tag{2.57}$$

Moreover, according to the $-\frac{1}{4}a^2 \pm ab - b^2 = -\left(\frac{1}{2}a \mp b\right)^2$, Eq. 2.57 can be represented as

$$\begin{aligned}\dot{L}_{m,i}(t) \leq & -\frac{\alpha_{m,i}}{2}\frac{\left\|\hat{\Psi}_{m,i}\right\|^2\left\|\tilde{W}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2} \\ & -\frac{\alpha_{m,i}}{1+\left\|\hat{\Psi}_{m,i}\right\|^2}\left\|\frac{\tilde{W}_{m,i}^T\hat{\Psi}_{m,i}}{2}-W_{m,i}^T\tilde{\Psi}_i\right\|^2 \\ & +\alpha_{m,i}\frac{\left\|W_{m,i}^T\tilde{\Psi}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2}+\varepsilon_{NFPKi}\end{aligned} \tag{2.58}$$

where

$$\varepsilon_{NFPKi} = \frac{\alpha_{m,i}\left\|\varepsilon_{FPKi}\right\|^2}{1+\left\|\Psi_{m,i}\right\|^2}$$

To simplify the derivation in Eq. 2.58, several negative terms are dropped such as

$$\dot{L}_{m,i}(t) \leq \tag{2.59}$$

$$\begin{aligned}& -\frac{\alpha_{m,i}}{2}\frac{\left\|\hat{\Psi}_{m,i}\right\|^2\left\|\tilde{W}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2}+\alpha_{m,i}\frac{\left\|W_{m,i}^T\tilde{\Psi}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2}+\varepsilon_{NFPKi} \\ & -\frac{\alpha_{m,i}}{2}\frac{\left\|\hat{\Psi}_{m,i}\right\|^2\left\|\tilde{W}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2}+B_{Wm,i}(t)\end{aligned} \tag{2.60}$$

where

$$B_{Wm,i}(t) = \alpha_{m,i}\frac{l_{\Psi_{m,i}}\|W_{v,i}\|^2\left\|\tilde{V}_i\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2}+\varepsilon_{NFPKi}$$

and $l_{\Psi_{m,i}}$ represents the Lipschitz constant of $\Psi_{m,i}(\cdot)$, $\tilde{V}_i$ is the critic estimation error given in Eq. 2.36.

According to Lyapunov stability analysis and Eq. 2.59, the mass NN weight

estimation error will be uniformly bounded with the bound given as

$$\left\|\tilde{W}_{m,i}\right\| \leq \sqrt{\frac{2(1+\left\|\hat{\Psi}_{m,i}\right\|^2)}{\alpha_{m,i}\left\|\hat{\Psi}_{m,i}\right\|^2}B_{Wm,i}(t)} \equiv b_{Wm,i}(t) \tag{2.61}$$

This completes the proof. □

**Theorem 4.** *(Mass neural network convergence)*: Let the mass neural network weights be updated following the updated law (2.31), and let the learning rate $\alpha_{m,i}$ be a positive number, the mass neural network's weight error $\tilde{W}_{m,i}$ and the PDF approximation error $\tilde{m}_i = m_i - \hat{m}_i$ are UUB in the stochastic sense. The bounds are negligible if the reconstruction error is ignored.

*Proof.* The details are presented in Appendix C. □

Finally, the closed-loop stability is given with an additional lemma.

**Lemma 1.** Given the system dynamics in (2.1), the optimal control $u_i^*$ satisfies,

$$x_i^T\left[f(x_i) + g(x_i)u_i^* + \sigma\frac{dw_i}{dt}\right] \leq -\gamma\left\|x_i\right\|^2 \tag{2.62}$$

where $\gamma > 0$.

**Theorem 5.** Let the critic, mass, and actor neural network weights are updated as Eqs. 2.30, 2.31, and 2.32. There exist constants $\alpha_{h,i} > 0$, $\alpha_{m,i} > 0$, and $\alpha_{u,i} > 0$ such that $x_i$, $\tilde{W}_{J,i}$, $\tilde{W}_{m,i}$, and $\tilde{W}_{u,i}$ are all UUB in the stochastic sense. The corresponding bounds are negligible if the reconstruction error is ignored.

*Proof.*

$$L_{sysm,i}(t) = \frac{\beta_1}{2}\operatorname{tr}\left\{e_i^T(t)e_i(t)\right\} + \frac{\beta_2}{2}\operatorname{tr}\left\{\tilde{W}_{V,i}^T(t)\tilde{W}_{V,i}(t)\right\}$$

$$+\frac{\beta_3}{2}\,\mathrm{tr}\left\{\tilde{W}_{m,i}^T(t)\tilde{W}_{m,i}(t)\right\}+\frac{\beta_4}{2}\,\mathrm{tr}\left\{\tilde{W}_{u,i}^T(t)\tilde{W}_{u,i}(t)\right\} \tag{2.63}$$

According the Lyapunov stability method, taking the first derivative of selected Lyapunov candidate, we have

$$\begin{aligned}
&\dot{L}_{sysm,i}(t)=\frac{\beta_1}{2}\mathrm{tr}\left\{e_i^T(t)\dot{e}_i(t)\right\}+\frac{\beta_1}{2}\mathrm{tr}\left\{\dot{e}_i^T(t)e_i(t)\right\}\\
&+\frac{\beta_2}{2}\mathrm{tr}\left\{\tilde{W}_{V,i}^T(t)\dot{\tilde{W}}_{V,i}(t)\right\}+\frac{\beta_2}{2}\mathrm{tr}\left\{\dot{\tilde{W}}_{V,i}^T(t)\tilde{W}_{V,i}(t)\right\}\\
&+\frac{\beta_3}{2}\mathrm{tr}\left\{\tilde{W}_{m,i}^T(t)\dot{\tilde{W}}_{m,i}(t)\right\}+\frac{\beta_3}{2}\mathrm{tr}\left\{\dot{\tilde{W}}_{m,i}^T(t)\tilde{W}_{m,i}(t)\right\}\\
&+\frac{\beta_4}{2}\mathrm{tr}\left\{\tilde{W}_{u,i}^T(t)\dot{\tilde{W}}_{u,i}(t)\right\}+\frac{\beta_4}{2}\mathrm{tr}\left\{\dot{\tilde{W}}_{u,i}^T(t)\tilde{W}_{u,i}(t)\right\}\\
&=\beta_1\mathrm{tr}\left\{e_i^T(t)\dot{e}_i(t)\right\}+\beta_2\mathrm{tr}\left\{\tilde{W}_{V,i}^T(t)\dot{\tilde{W}}_{V,i}(t)\right\}\\
&+\beta_3\mathrm{tr}\left\{\tilde{W}_{m,i}^T(t)\dot{\tilde{W}}_{m,i}(t)\right\}+\beta_4\mathrm{tr}\left\{\tilde{W}_{u,i}^T(t)\dot{\tilde{W}}_{u,i}(t)\right\}
\end{aligned} \tag{2.64}$$

Recall to Lemma 1 and Theorem 1, 2, and 3 given in Eq. 2.52, Eq. 2.59, and Eq. 2.44, Eq. 2.64 can be represented as:

$$\begin{aligned}
&\dot{L}_{sysm,i}(t)=\beta_1\mathrm{tr}\left\{e_i^T(t)\dot{e}_i(t)\right\}+\beta_2\mathrm{tr}\left\{\tilde{W}_{V,i}^T(t)\dot{\tilde{W}}_{V,i}(t)\right\}\\
&+\beta_3\mathrm{tr}\left\{\tilde{W}_{m,i}^T(t)\dot{\tilde{W}}_{m,i}(t)\right\}+\beta_4\mathrm{tr}\left\{\tilde{W}_{u,i}^T(t)\dot{\tilde{W}}_{u,i}(t)\right\}\\
&\leq\beta_1\mathrm{tr}\left\{e_i^T[f(e_I)+g(e_i)u_i^*-\frac{de_d}{dt}+\sigma_i\frac{dw_i}{dt}]\right\}\\
&-\beta_1\mathrm{tr}\left\{e_i^Tg(e_i)\tilde{u}_i\right\}-\frac{2\beta_1}{\gamma}\left\|g(e_i)\tilde{u}_i\right\|+\frac{2\beta_1}{\gamma}\left\|g(e_i)\tilde{u}_i\right\|\\
&-\frac{\alpha_{h,i}\beta_2}{4}\frac{\left\|\hat{\Psi}_{V,i}\right\|^2}{1+\left\|\hat{\Psi}_{V,i}\right\|^2}\left\|\tilde{W}_{V,i}\right\|^2\\
&+\alpha_{h,i}\frac{\beta_2\left[l_{\Phi,i}+l_{\Psi_{V,j}}\|W_{V,i}\|^2\right]\|\tilde{m}_i\|^2}{1+\left\|\hat{\Psi}_{V,i}\right\|^2}\\
&-\frac{\alpha_{m,i}}{2}\frac{\beta_3\left\|\hat{\Psi}_{m,i}\right\|^2\left\|\tilde{W}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2}+\alpha_{m,i}\frac{\beta_3\|W_{v,i}\|^2\left\|\tilde{V}_i\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2}
\end{aligned} \tag{2.65}$$

$$
\begin{aligned}
&- \frac{\alpha_{u,i}}{4} \frac{\beta_4 \left\| \hat{\phi}_{u,i} \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \left\| \tilde{W}_{u,i} \right\|^2 + \alpha_{u,i} \beta_4 \frac{\left\| R^{-1} g^T (e_i) \right\|^2 \left\| \tilde{V}_i \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \\
&+ \beta_2 \varepsilon_{VHB,i} + \beta_3 \varepsilon_{NFPKi} + \beta_4 \varepsilon_{Nu,i} \\
&\leq -\frac{\gamma \beta_1}{2} \left\| e_i \right\|^2 - \frac{\gamma \beta_1}{2} \left\| e_i \right\|^2 - \beta_1 \operatorname{tr} \left\{ e_i^T g (e_i) \tilde{u}_i \right\} \\
&- \frac{2\beta_1}{\gamma} \left\| g (e_i) \tilde{u}_i \right\|^2 + \frac{2\beta_1}{\gamma} \left\| g (e_i) \tilde{u}_i \right\|^2 \\
&- \frac{\alpha_{h,i} \beta_2}{4} \frac{\left\| \hat{\Psi}_{V,i} \right\|^2}{1 + \left\| \hat{\Psi}_{V,i} \right\|^2} \left\| \tilde{W}_{V,i} \right\|^2 \\
&+ \alpha_{h,i} \frac{\beta_2 \left[ l_{\Phi,i} + l_{\Psi_{V,j}} \| W_{V,i} \|^2 \right] \| \tilde{m}_i \|^2}{1 + \left\| \hat{\Psi}_{V,i} \right\|^2} \\
&- \frac{\alpha_{m,i}}{2} \frac{\beta_3 \left\| \hat{\Psi}_{m,i} \right\|^2 \left\| \tilde{W}_{m,i} \right\|^2}{1 + \left\| \hat{\Psi}_{m,i} \right\|^2} + \alpha_{m,i} \frac{\beta_3 \| W_{v,i} \|^2 \left\| \tilde{V}_i \right\|^2}{1 + \left\| \hat{\Psi}_{m,i} \right\|^2} \\
&- \frac{\alpha_{u,i}}{4} \frac{\beta_4 \left\| \hat{\phi}_{u,i} \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \left\| \tilde{W}_{u,i} \right\|^2 + \alpha_{u,i} \beta_4 \frac{\left\| R^{-1} g^T (e_i) \right\|^2 \left\| \tilde{V}_i \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \\
&+ \beta_2 \varepsilon_{VHB,i} + \beta_3 \varepsilon_{NFPKi} + \beta_4 \varepsilon_{Nu,i} \\
&\leq -\frac{\gamma \beta_1}{2} \left\| e_i \right\|^2 - \beta_1 \left[ \sqrt{\frac{\gamma}{2}} \left\| e_i \right\| + \sqrt{\frac{2}{\gamma}} \left\| g (e_i) \tilde{u}_i \right\| \right]^2 \\
&+ \frac{2 g_M^2 \beta_1}{\gamma} \left\| \tilde{u}_i \right\|^2 - \frac{\alpha_{h,i} \beta_2}{4} \frac{\left\| \hat{\Psi}_{V,i} \right\|^2}{1 + \left\| \hat{\Psi}_{V,i} \right\|^2} \left\| \tilde{W}_{V,i} \right\|^2 \\
&+ \alpha_{h,i} \frac{\beta_2 \left[ l_{\Phi,i} + l_{\Psi_{V,j}} \| W_{V,i} \|^2 \right] \| \tilde{m}_i \|^2}{1 + \left\| \hat{\Psi}_{V,i} \right\|^2} \\
&- \frac{\alpha_{m,i}}{2} \frac{\beta_3 \left\| \hat{\Psi}_{m,i} \right\|^2 \left\| \tilde{W}_{m,i} \right\|^2}{1 + \left\| \hat{\Psi}_{m,i} \right\|^2} + \alpha_{m,i} \frac{\beta_3 \| W_{v,i} \|^2 \left\| \tilde{V}_i \right\|^2}{1 + \left\| \hat{\Psi}_{m,i} \right\|^2} \\
&- \frac{\alpha_{u,i}}{4} \frac{\beta_4 \left\| \hat{\phi}_{u,i} \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2} \left\| \tilde{W}_{u,i} \right\|^2 + \alpha_{u,i} \beta_4 \frac{\left\| R^{-1} g^T (e_i) \right\|^2 \left\| \tilde{V}_i \right\|^2}{1 + \left\| \hat{\phi}_{u,i} \right\|^2}
\end{aligned}
$$

$$
\begin{aligned}
&+ \beta_2 \varepsilon_{VHB,i} + \beta_3 \varepsilon_{NFPKi} + \beta_4 \varepsilon_{Nu,i} \\
&\leq -\frac{\gamma}{2}\beta_1 \|e_i\|^2 + \frac{2g_M^2 \beta_1}{\gamma}\|\tilde{u}_i\|^2 - \frac{\alpha_{h,i}\beta_2}{4}\frac{\left\|\hat{\Psi}_{V,i}\right\|^2 \left\|\tilde{W}_{V,i}\right\|^2}{1+\left\|\hat{\Psi}_{V,i}\right\|^2} \\
&+ \alpha_{h,i}\frac{\beta_2\left[l_{\Phi,i} + l_{\Psi_{V,j}}\|W_{V,i}\|^2\right]\|\tilde{m}_i\|^2}{1+\left\|\hat{\Psi}_{V,i}\right\|^2} \\
&- \frac{\alpha_{m,i}}{2}\frac{\beta_3\left\|\hat{\Psi}_{m,i}\right\|^2\left\|\tilde{W}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2} - \frac{\alpha_{u,i}}{4}\frac{\beta_4\left\|\hat{\phi}_{u,i}\right\|^2}{1+\left\|\hat{\phi}_{u,i}\right\|^2}\left\|\tilde{W}_{u,i}\right\|^2 \\
&+ \beta_2 \varepsilon_{VHB,i} + \beta_3 \varepsilon_{NFPKi} + \beta_4 \varepsilon_{Nu,i} \\
&+ \frac{\alpha_{m,i}\beta_3 l_{\Psi_{mi}}\|W_{V,i}\|^2 + \alpha_{u,i}\beta_4\left\|R^{-1}g^T(e_i)\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2}\left\|\tilde{V}_i\right\|^2
\end{aligned}
\tag{2.66}
$$

where $g_M$ is the Lipschitz constant of the dynamic equation $g(e_i)$. Let

$$
\begin{aligned}
b_2 =& -\frac{\gamma}{2}\beta_1\|e_i\|^2 + \frac{2g_M^2\beta_1}{\gamma}\|\tilde{u}_i\|^2 - \frac{\alpha_{h,i}\beta_2}{4}\frac{\left\|\hat{\Psi}_{V,i}\right\|^2\left\|\tilde{W}_{V,i}\right\|^2}{1+\left\|\hat{\Psi}_{V,i}\right\|^2} \\
b_3 =& -\frac{\alpha_{m,i}}{2}\frac{\beta_3\left\|\hat{\Psi}_{m,i}\right\|^2\left\|\tilde{W}_{m,i}\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2} - \frac{\alpha_{u,i}}{4}\frac{\beta_4\left\|\hat{\phi}_{u,i}\right\|^2}{1+\left\|\hat{\phi}_{u,i}\right\|^2}\left\|\tilde{W}_{u,i}\right\|^2 \\
&+ \beta_2\varepsilon_{VHB,i} + \beta_3\varepsilon_{NFPKi} + \beta_4\varepsilon_{Nu,i} \\
b_4 =& \frac{\alpha_{m,i}\beta_3 l_{\Psi_{mi}}\|W_{V,i}\|^2 + \alpha_{u,i}\beta_4\left\|R^{-1}g^T(e_i)\right\|^2}{1+\left\|\hat{\Psi}_{m,i}\right\|^2} \\
b_5 =& \alpha_{h,i}\frac{\beta_2\left[l_{\Phi,i} + l_{\Psi_{V,i}}\|W_{V,i}\|^2\right]}{1+\left\|\hat{\Psi}_{V,i}\right\|^2}
\end{aligned}
$$

Substituting 2.36 into 2.65, Eq. 2.65 can be represented as:

$$
\begin{aligned}
&\dot{L}_{sysm,i}(t) \leq b_2 + b_5\|\tilde{m}_i\|^2 \\
&+ b_4[\|\tilde{W}_{V,i}^T(t)\|\|\hat{\Psi}_{v,i}\| + l_{\Psi_{V,i}}\|W_{V,i}\|\|\tilde{m}_i\| + \|\varepsilon_{V,i}\|]^2 \\
&\leq b_2 + b_3 + 3b_4\left\|\hat{\Psi}_{V,i}\right\|^2\left\|\tilde{W}_{V,i}\right\|^2
\end{aligned}
$$

$$+\left[3b_4 l_{\Psi_{V,i}}^2 \left\|W_{V,i}\right\| + b_5\right] \left\|\tilde{m}_i\right\|^2 + 3b_4 \left\|\varepsilon_{V,i}\right\|^2 \tag{2.67}$$

Furthermore, substitute Eq.2.37 into 2.67, Eq. 2.67 can be expressed as:

$$\begin{aligned}
\dot{L}_{sysm,i}(t) \leq\ & b_2 + b_3 + 3b_4 \left\|\hat{\Psi}_{V,i}\right\|^2 \left\|\tilde{W}_{V,i}\right\|^2 \\
& + 2\left[3b_4 l_{\Psi_{V,i}}^2 \left\|W_{V,i}\right\|^2 + b_5\right] \left\|\Psi_{m,i}\right\|^2 \left\|\tilde{W}_{m,i}\right\|^2 \\
& + 2\left[3b_4 l_{\Psi_{V,i}}^2 \left\|W_{V,i}\right\|^2 + b_5\right] \left\|\varepsilon_{m,i}\right\|^2 + 3b_4 \left\|\varepsilon_{V,i}\right\|^2
\end{aligned} \tag{2.68}$$

Furthermore, substitute Eq.2.38 into Eq. 2.68 and combine terms, Eq.2.68 can be expressed as:

$$\begin{aligned}
\dot{L}_{sysm,i}(t) \leq\ & -\frac{\gamma\beta_1}{2} \left\|e_i\right\|^2 \\
& - \left[\frac{\alpha_{u,i}\beta_4}{4} \frac{\left\|\hat{\phi}_{u,i}\right\|^2}{1 + \left\|\hat{\phi}_{u,i}\right\|^2} - \frac{6g_M^2\beta_1}{\gamma} \left\|\hat{\phi}_{u,i}\right\|^2\right] \left\|\tilde{W}_{u,i}\right\|^2 \\
& - \left[\frac{\alpha_{h,i}\beta_1}{4} \hat{\Psi}_{V,i} - b_6 \left\|\hat{\Psi}_{V,i}\right\|^2\right] \left\|\tilde{W}_{V,i}\right\|^2 \\
& - \left[\frac{\alpha_{m,i}\beta_3}{2} \hat{\Psi}_{m,i} - 2(b_6 l_{\Psi_{V,i}}^2 \left\|W_{V,i}\right\|^2 + b_7) \left\|\hat{\Psi}_{m,i}\right\|^2\right] \left\|\tilde{W}_{m,i}\right\|^2 \\
& + 2\left(b_6 l_{\Psi_{V,i}}^2 \left\|W_{V,i}\right\|^2 + b_7\right) \left\|\varepsilon_{m,i}\right\|^2 + \frac{6g_M^2\beta_1}{\gamma} \left\|\varepsilon_{u,i}\right\|^2 \\
& + b_6 \left\|\varepsilon_{V,i}\right\|^2 + \beta_4 \varepsilon_{Nu,i} + \beta_3 \mathcal{E}_{NFPK,i} + \beta_2 \varepsilon_{VHB,i}
\end{aligned} \tag{2.69}$$

where

$$b_6 = \frac{3\left[\alpha_{m,i} l_{\Psi_{m,i}} \beta_3 \left\|W_{V,i}\right\|^2 + \alpha_{u,i}\beta_4 \left\|R^{-1} g^T(e_i)\right\|^2\right]}{1 + \left\|\Psi_{m,i}\left(e_i, \overline{m}_i, \hat{V}_i, t\right)\right\|^2}$$

$$b_7 = \alpha_{H,i} \frac{\left[l_{hi} + l_{\Psi_{vi}} \left\|W_{V,i}\right\|^2\right] \beta_1}{1 + \left\|\Psi_{V,i}(e_i, \hat{m}_i, t)\right\|^2} + \frac{6g_M^2\beta_1}{\gamma} \left\|W_{u,i}\right\|^2 l_{\phi u,i}^2$$

where $l_{\phi_{u,i}}$ is the Lipschitz constant for the actor activation functions $\phi_{u,i}(\cdot)$. Eq. 2.69 can be rewrote as:

$$\dot{L}_{sysm,i}(t) \leq -\frac{\gamma\beta_1}{2}\|e_i\|^2 - \kappa_u\left\|\tilde{W}_{u,i}\right\|^2 - \kappa_m\left\|\tilde{W}_{m,i}\right\|^2 - \kappa_V\left\|\tilde{W}_{V,i}\right\|^2 + \varepsilon_{CLS}$$

Using the Lyapunov stability analysis, the derivative of selected Lyapunov function candidate $\dot{L}_{sysm,i}(t)$ is less than zero outside a compact set, i.e.

$$\|e_i\| > \sqrt{\frac{2}{\gamma\beta_1}\varepsilon_{CLS}} \quad \text{or} \quad \left\|\tilde{W}_{u,i}\right\| > \sqrt{\frac{1}{\kappa_u}\varepsilon_{CLS}}$$

or

$$\left\|\tilde{W}_{m,i}\right\| > \sqrt{\frac{1}{\kappa_m}\varepsilon_{CLS}} \quad \text{or} \quad \left\|\tilde{W}_{V,i}\right\| > \sqrt{\frac{1}{\kappa_V}\varepsilon_{CLS}}$$

with

$$\kappa_u = \frac{\alpha_{u,i}\beta_4}{4}\hat{\Psi}_{m,i} - \frac{6g_M^2\beta_1}{\gamma}\left\|\hat{\phi}_{u,i}\right\|^2]$$

$$\kappa_m = \frac{\alpha_{m,i}\beta_3}{2}\hat{\Psi}_{m,i} + 2(a_1 l_{\Psi_{V,i}}^2\|W_{V,i}\|^2 + a_2)\|\hat{\Psi}_{m,i}\|^2$$

$$\kappa_V = \frac{\alpha_{H,i}\beta_4}{4}\hat{\Psi}_{V,i} - a_1\|\hat{\Psi}_{V,i}\|^2$$

$$\varepsilon_{CLS} = 2(a_1 l_{\Psi_{V,i}}^2\|W_{V,i}\|^2 + a_2)\|\varepsilon_{m,i}\|^2 + \frac{6g_M^2\beta_1}{\gamma}\|\varepsilon_{u,i}\|^2 + a_1\|\varepsilon_{V,i}\|^2 + \beta_4\varepsilon_{Nu,i} + \beta_3\varepsilon_{NFPK,i} + \beta_2\varepsilon_{VHB,i}$$

This completes the proof. □

If the weights are perfectly approximated and assuming the reconstruction errors are negligible, the optimal cost function, PDF of all agents' tracking error, and the optimal control can be approximated correctly.

## 2.4 Numerical Experiments

In this section, the developed algorithm has been evaluated using a large-scale multi-agent control task. A 2-D map is scaled to $20 \times 20$ as the task's map with 1000

agents simulate and interact with each other. The agents' initial positions are randomly distributed and a planned reference trajectory is given to all agents. Meanwhile, initial distribution of all agents' tracking errors are measured and broadcasted to all agents prior to the mission. No communication is allowed after the task starts due to the harsh environment. To demonstrate the effectiveness, both linear and non-linear simulations are provided.

### 2.4.1 Linear System

In this subsection, a linear case of the tracking control problem is considered. The continuous-time linear second-order motion equations without noise for each agent are given by

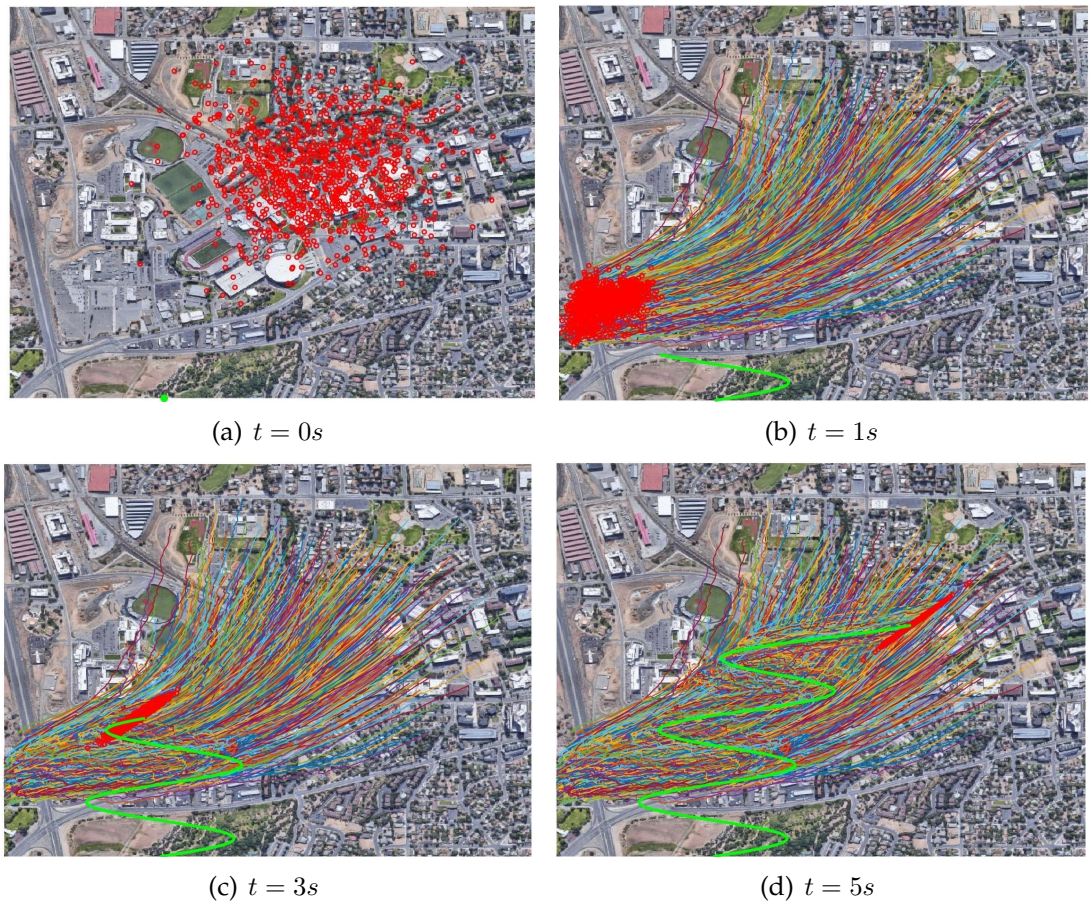
$$\begin{cases} dp_i = q_i dt \\ dq_i = u_i dt \end{cases} \tag{2.70}$$

where $p_i, q_i \in \mathbb{R}^2$. To test the robustness of the designed controller, a Brownian noise is added to the system. The corresponding stochastic state-space model with noise can be derived as

$$d\rho_i = \big[A\rho_i + Bu_i\big]dt + \sigma dw_i \tag{2.71}$$

where $\rho_i = [p_i, q_i] \in \mathbb{R}^4$ is the augmented state, $A, B$ can be computed from (2.70), $\sigma$ is set to 0.5 for all agents. The time-dependent reference trajectory is given as:

$$x_d(t) = \begin{bmatrix} 0.24\sin(4t) + 0.0034t^3 + 0.5 \\ 0.2t \\ 0.96\cos(4t) + 0.0102t^2 \\ 0.2 \end{bmatrix}$$



(a) $t = 0s$ (b) $t = 1s$

(c) $t = 3s$ (d) $t = 5s$

Figure 2.2: The overall position and trajectory of all agents in the MAS. The green curve represented the reference trajectory. Positions are marked with red dots. A linear system is used for simulation.

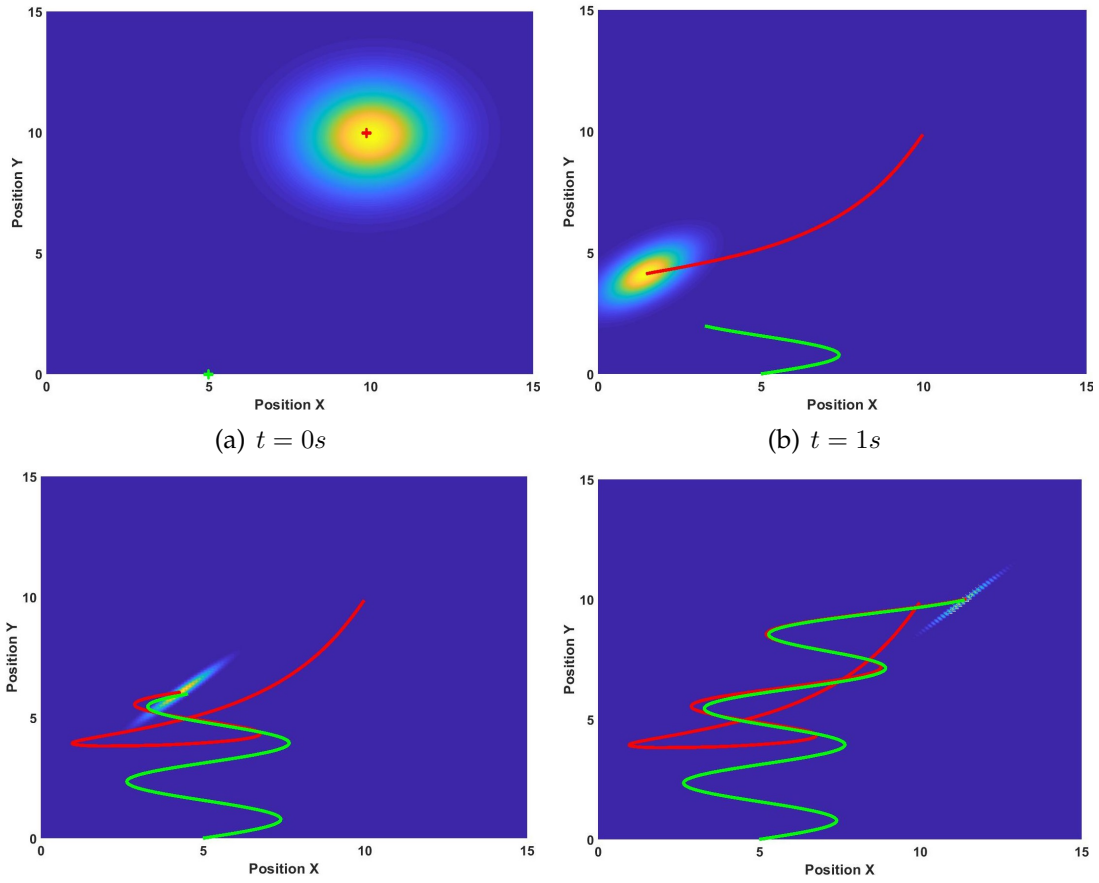
Thus, tracking error $x_i \in \mathbb{R}^4$ can be computed by (2.3).

For simplicity, $\Phi(m, x_i)$ is designed as

$$\Phi(m, x_i) = \|x_i - \mathbb{E}[m]\| \tag{2.72}$$

The initial state distribution of all agents are given as,

$$m_0 \sim \mathcal{N}(\begin{bmatrix} 10 \\ 10 \end{bmatrix}, \begin{bmatrix} 5^2 & 0 \\ 0 & 5^2 \end{bmatrix}) \tag{2.73}$$




(a) $t = 0s$ (b) $t = 1s$

(c) $t = 3s$ (d) $t = 5s$

Figure 2.3: The PDF of all agents in the MAS. The red curve represents the average trajectory of all agents, and the green curve denotes the reference trajectory. A linear system is used for simulation.

As illustrated by the key idea of the developed ACM algorithm, three neural networks are designed. The activation functions of these neural networks are vector of functions which are designed from the polynomial's expansion. Additionally, the time $t$ is first normalized to prevent output explosion.

The experiment results in the linear system are demonstrated in Fig. 2.2(a) to 2.8. In Fig. 2.2(a), all agents' positions are marked with the red dots. The green dot at the bottom denotes the initial position of the reference trajectory. All agents' trajectories at $1s$, $3s$, and $5s$ are plotted in Figs. 2.2(b), 2.2(c), and 2.2(d) where

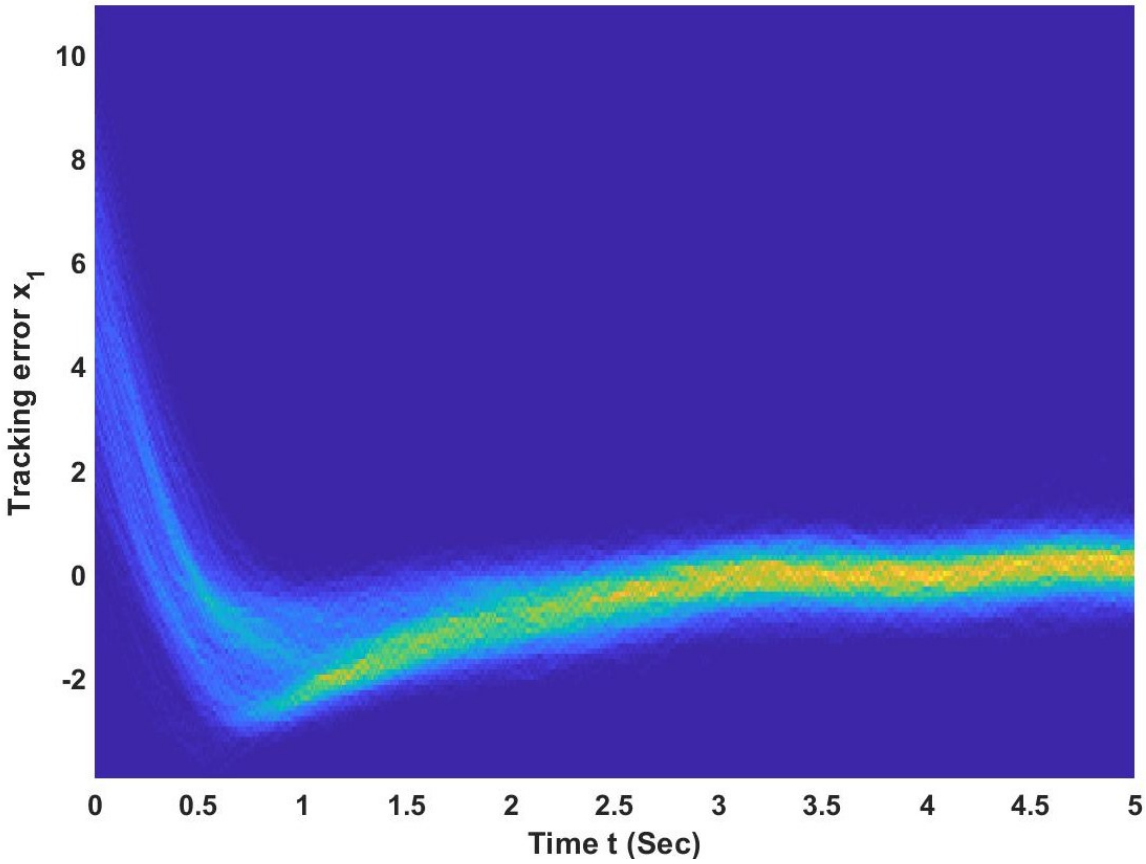



Figure 2.4: Linear case tracking error PDF of $x_1$ to time.

green curve marks the reference trajectory, and other curves denote the individule trajectories. It can be observed that all agents follow the reference trajectory successfully. To better analyze the whole population's mass behavior, we plot the distribution of the agents in Figs. 2.3(a), 2.3(b), 2.3(c), and 2.3(d) with respect to time. Note that the plots are the probability density function regressed from the agents' position distribution. The mean trajectory is also computed and marked as a red curve in those plots. The agents' trajectories and mass plot confirm that the reference trajectory can be tracked successfully.

Then, we plot the tracking errors. Fig. 2.4 and Fig. 2.5 demonstrate the distri-

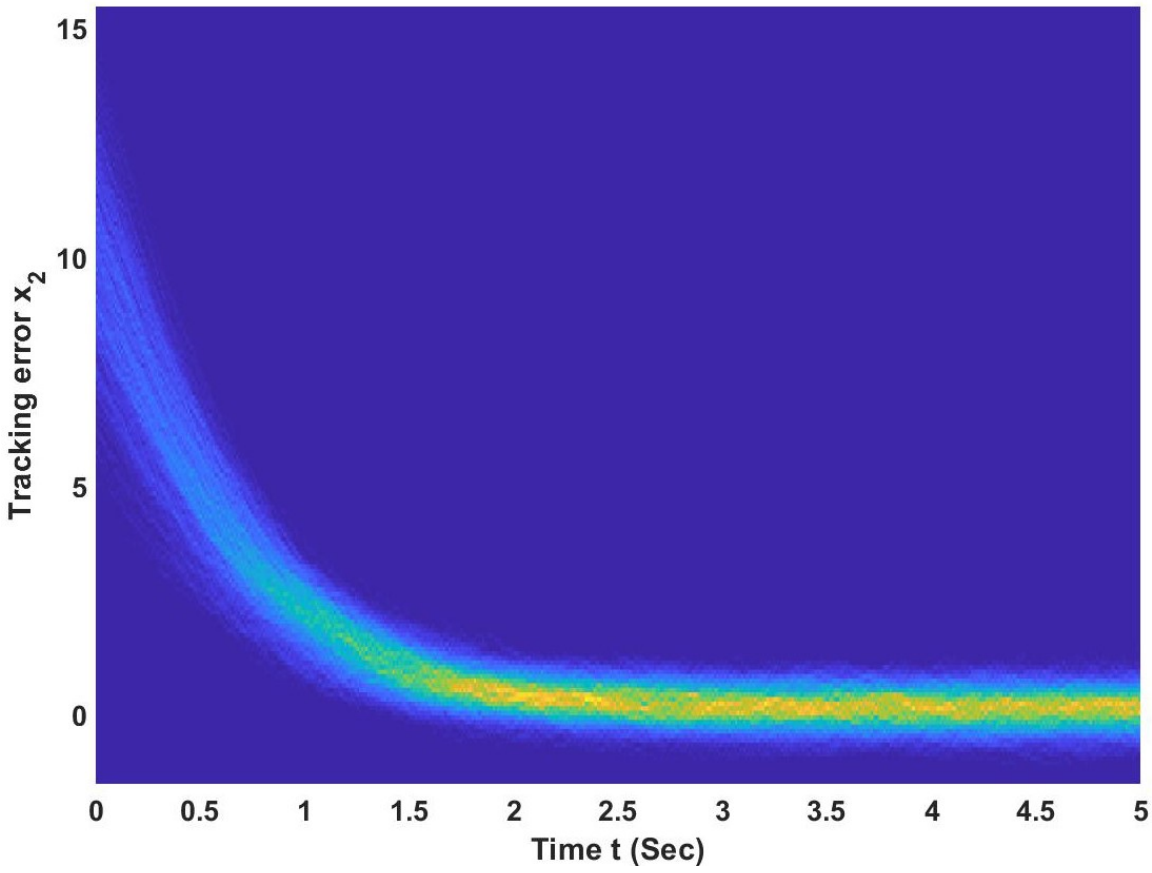



Figure 2.5: Linear case tracking error PDF of $x_2$ to time.

bution of agents tracking error $x$ with respect to time. We only plot the agent's first and second entry of tracking error states (i.e., $x_1$ and $x_2$) because they represent the positions of all agents. The plots show that the initial tracking error is high and randomly distributed. However, after 3 seconds, the agents' mean tracking errors are bounded near zero and the variance of tracking error PDF decreases to near zero. This also proves that the system is able to track the given trajectory.

Finally, the performance of three neural networks are studied. The HJB, FPK, and actor NN errors (i.e.$e_{HJBi}$ in (2.15), $e_{FPKi}$ in (2.16), and $e_{ui}$ in (2.17)) are plotted to show the convergence of 3 NNs. We plot agent 1's FPK equation error, HJB

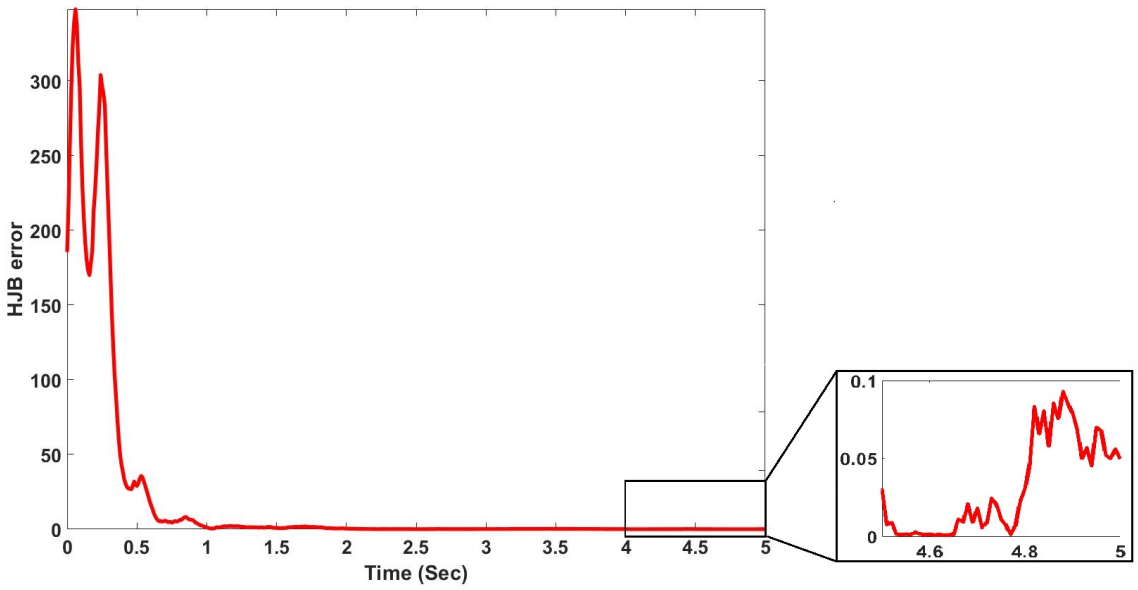



Figure 2.6: Linear case HJB equation error.

equation error, and actor NN approximation error in Figs. 2.8, 2.6, and 2.7, respectively. All the three figures show that the errors reduce to near zero after 4 seconds. Meanwhile, the convergence of both FPK equation error and HJB equation error confirm that the NNs reveal a correct approximation of the MFG equation's solution if the reconstruction error are ignored. And the convergence of the actor NN guaranteed that the computed control is the optimal control.

### 2.4.2 Nonlinear System

Now we consider a more complicate nonlinear dynamics with noise. The nonlinear stochastic state-space model is selected as:

$$d\rho_i = \big[f_y(\rho_i) + g_y(\rho_i)u_i\big]dt + \sigma dw_i \tag{2.74}$$

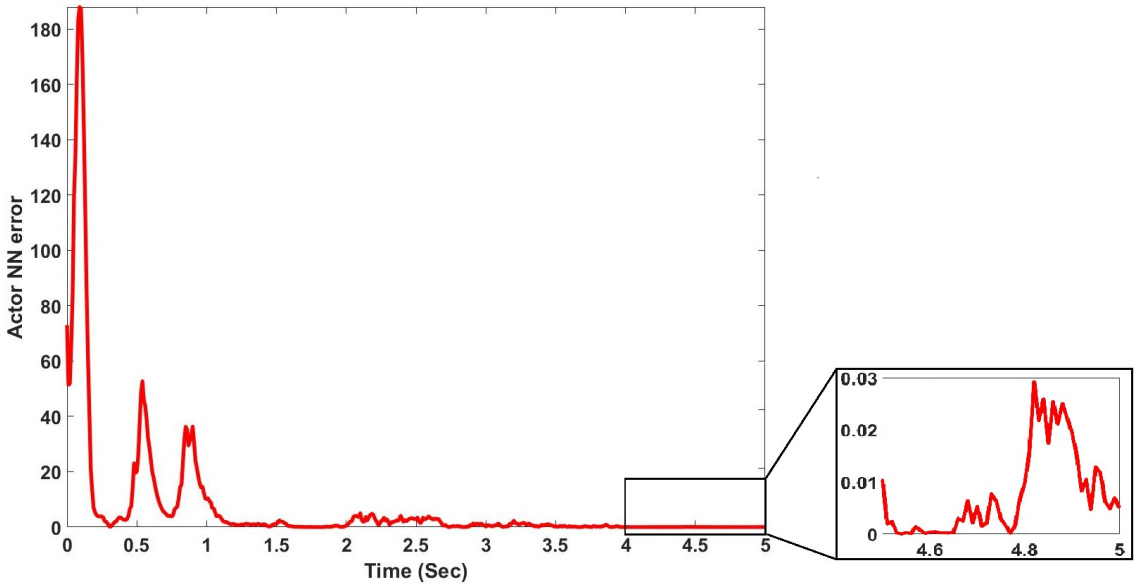



Figure 2.7: Linear case actor NN estimation error.

where $\rho_i = [p_i \quad q_i]^T \in \mathbb{R}^4$, and

$$f_y(\rho) = \begin{bmatrix} \rho_2 - \rho_1 \\ \rho_4 - \rho_3 \\ \frac{\rho_2}{2}[(\cos(2\rho_1 + 2)^2 - 1] - \frac{\rho_1}{2} \\ \frac{\rho_4}{2}[(\cos(2\rho_3 + 2)^2 - 1] - \frac{\rho_3}{2} \end{bmatrix}$$

$$g_y(\rho) = \begin{bmatrix} 0 \\ 0 \\ \cos(2\rho_1) + 2 \\ \cos(2\rho_3) + 2 \end{bmatrix}$$

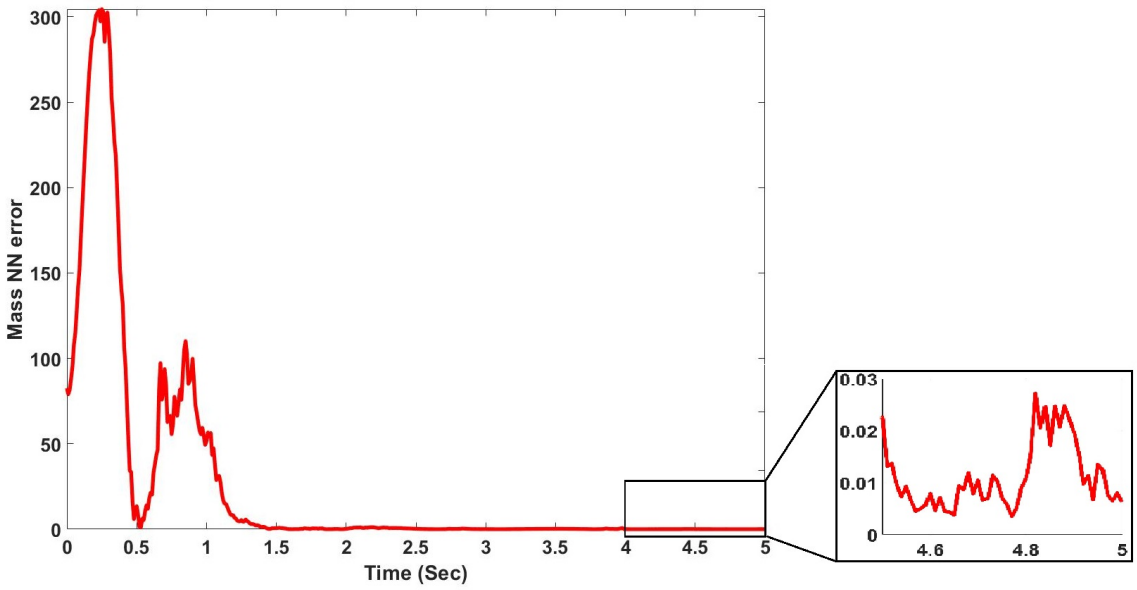



Figure 2.8: Linear case FPK equation error.

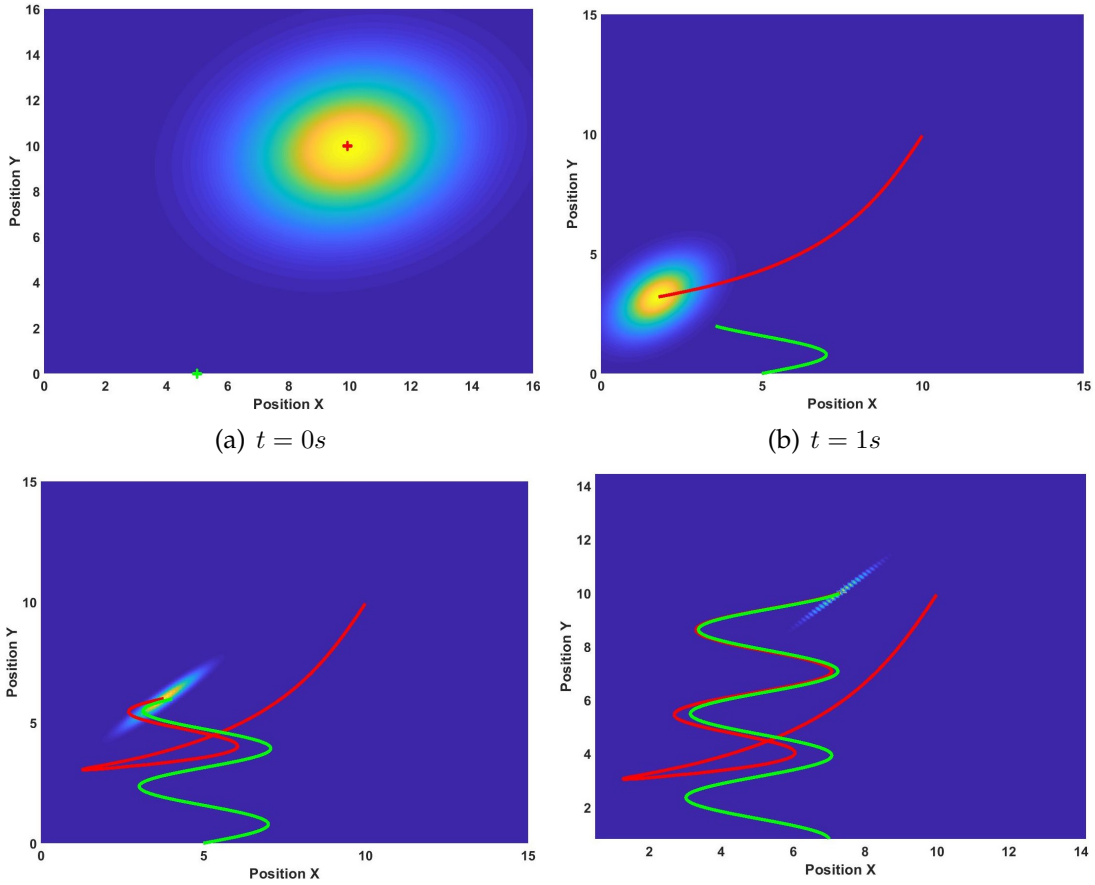
The coefficient of the noise $\sigma$ in (2.74) is set to $\sigma = 0.5$. The reference trajectory is given as:

$$x_d(t) = \begin{bmatrix} 0.2\sin(4t) + 0.002t^2 + 250 \\ 0.2 \\ 0.8\cos(4t) + 0.004t \\ 0.2 \end{bmatrix} \tag{2.75}$$

The initial agents' PDF is randomly selected using the following normal distribution:

$$m_0 \sim \mathcal{N}\left( \begin{bmatrix} 10 \\ 10 \end{bmatrix}, \begin{bmatrix} 8^2 & 0 \\ 0 & 8^2 \end{bmatrix} \right) \tag{2.76}$$

The activation functions of the neural networks are designed the same as the linear case.



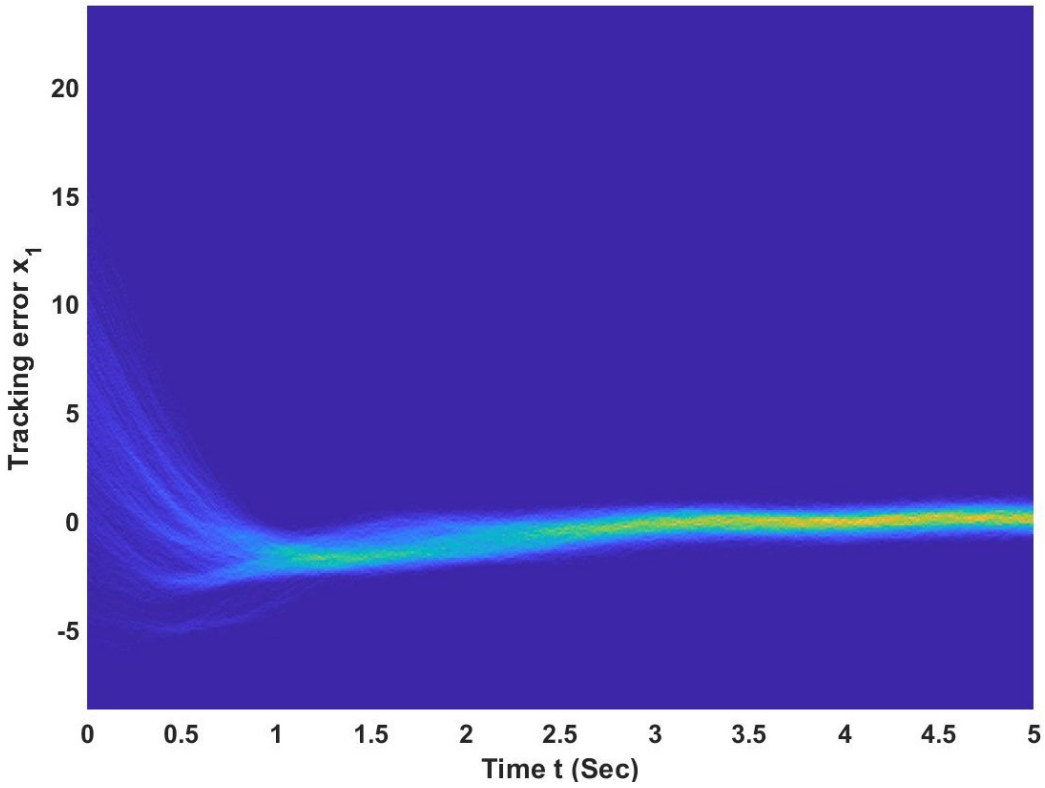

(a) $t = 0s$ (b) $t = 1s$

(c) $t = 3s$ (d) $t = 5s$

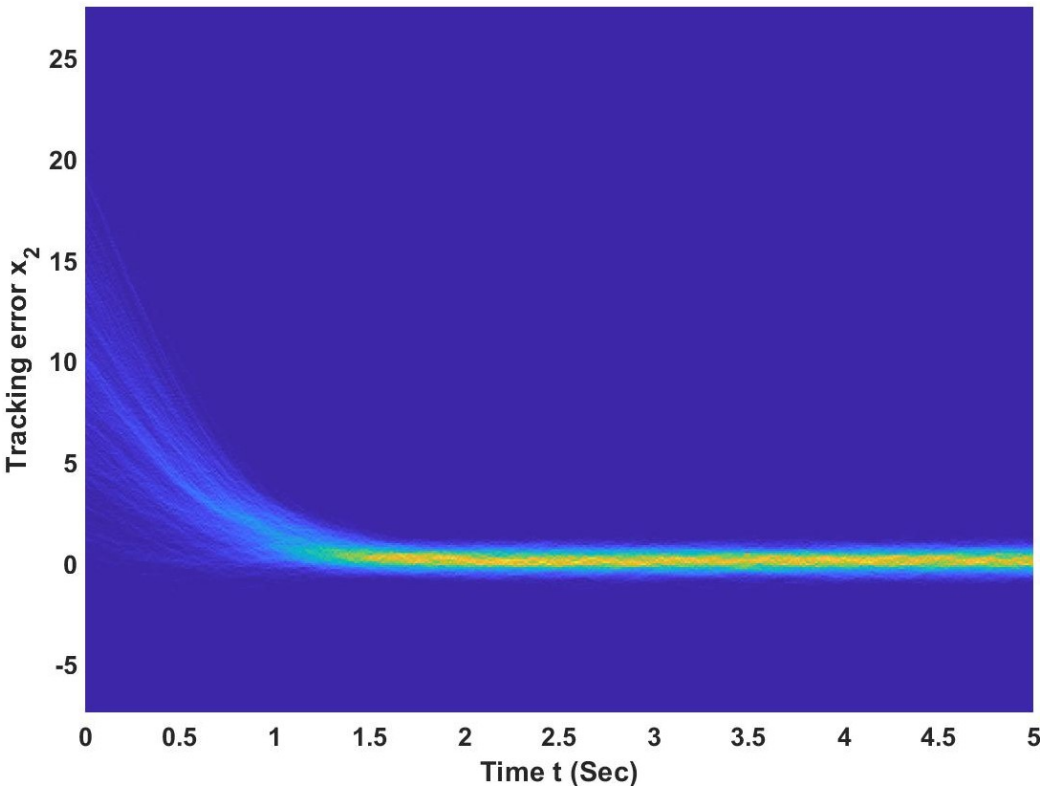
Figure 2.9: The density of all agents in the MAS. The green curve represents the reference trajectory. A nonlinear system is used for simulation.

Similar to the linear system case, we also plot the PDF of of all agents' position and the mean trajectory in Figs. 2.9(a) to 2.9(d) as a demonstration of the stability of the system. The legends are the same with the linear case simulation. We can observe that all agents can successfully follow the reference trajectory.

Next, the tracking error's PDF is plotted in Fig. 2.10(a) and 2.10(b). Similar to the linear case, the mean tracking error converges to zero in Fig. 2.10(a) and 2.10(b). This confirms that the developed ACM algorithm is a stable controller in nonlinear systems.

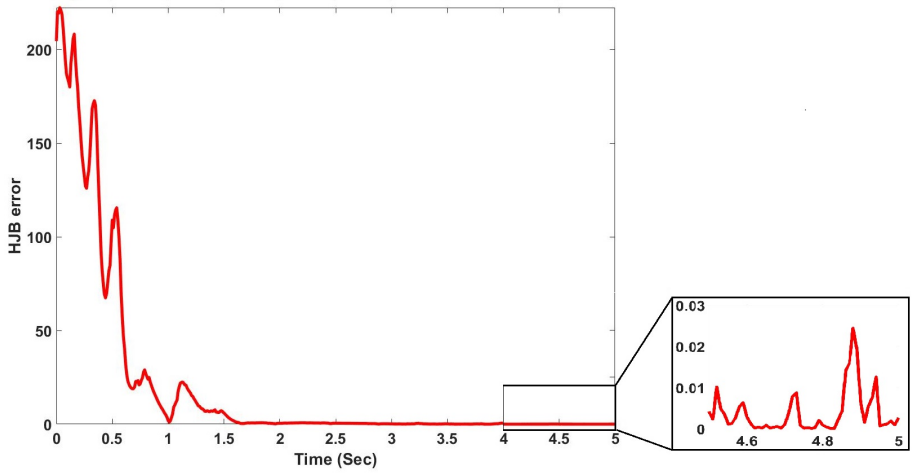



(a) Nonlinear case tracking error PDF of $x_1$ .

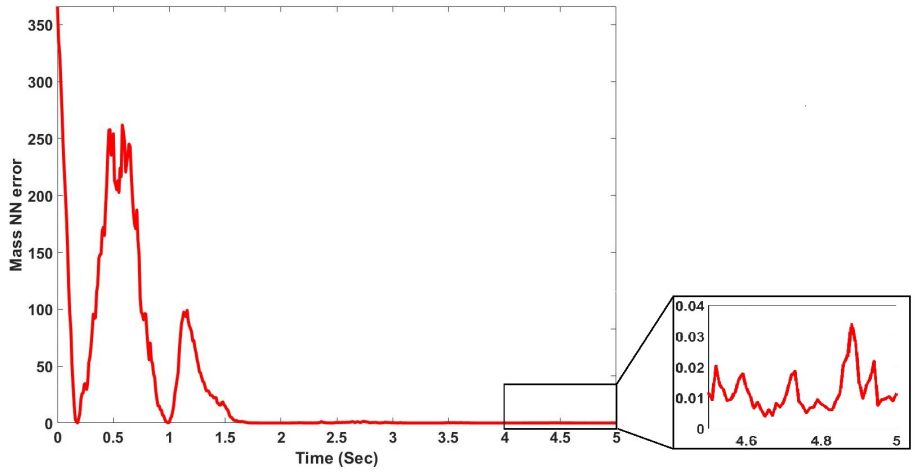



(b) Nonlinear case tracking error PDF of $x_2$

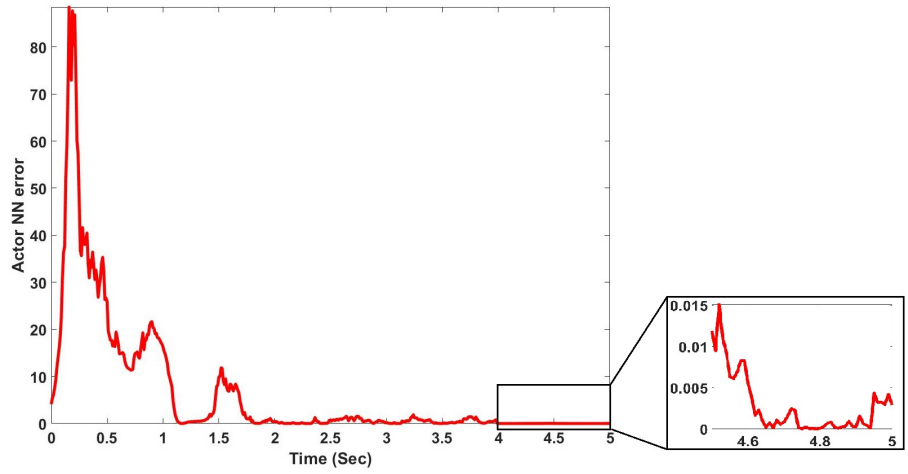
Figure 2.10: Neural network approximation errors.




(a) Nonlinear case HJB equation error.




(b) Nonlinear case FPK equation error.




(c) Nonlinear case actor neural network estimation error.

Figure 2.11: Neural network approximation errors.

Finally, the performance of the neural networks are evaluated through the error of the FPK and HJB equations. The time evolution of the errors are shown in Figs. 2.11(b), 2.11(a), and 2.11(c).

We can see that the HJB and FPK equation error converge near zero after 4 seconds. This confirms that the approximation of the mean field equations is accurate. Recall the original mean field game theory [48], the solution to the mean field equations is the $\varepsilon_N-$Nash equilibrium, which is the optimal solution of the large-scale tracking control problem.

## 2.5 Conclusions

this chapter has developed a novel decentralized large-scale MAS control algorithm based on approximate dynamic programming and reinforcement learning. The mean field games theory are embedded to solve the large-scale multi-agent tracking control problem in a decentralized and computational efficient manner. Specifically, the complexity of the infinite number non-cooperative game is decoupled with the agent number. Moreover, the actor-critic reinforcement learning method is extended to a actor-critic-mass algorithm where three neural networks are employed to approximate the solution, i.e., the $\epsilon_N$-Nash equilibrium. The stability and the convergence of the states and neural networks are guaranteed by the Lyapunov stability analysis. Then the near optimality can be achieved provided by the proof in traditional mean field games. Finally, the numerical simulations on both linear and nonlinear cases are given to evaluate the algorithm in practical scenarios.

CHAPTER 3

# LARGE-SCALE MULTI-AGENT REINFORCEMENT LEARNING FOR PURSUIT-EVASION GAMES [117, 120]

## 3.1 Introduction

During the past decades, the multi-agent systems (MAS) have attracted increasing attention since MAS brings the new capability to achieve a series of complex missions that cannot be accomplished using the single agent/system [108, 84, 58, 55]. Among the various MAS applications [98, 32], the multi-player pursuit-evasion game is one of most important problem that has profound influence to both civilian and military industries, i.e., autonomous drone control [14], the tracking control for autonomous vehicles [95], guided missile and defense system [90], etc. To obtain the optimal multi-player pursuit-evasion strategies, some nontrivial issues arise, especially while the agent's number is continuously increasing larger and even close to infinite.

The first challenge is that the limited communication resource cannot ensure reliable information exchange among ultra-large numbers of agents [19, 112, 3]. According to most conventional MAS designs, e.g., the consensus algorithms [64, 122], formation control [71] etc., computing control policy at local agent requires the relevant system state/control information from the other agents. To maintain the information exchange among large scale multiple agents, a reliable and high-quality communication network is critical. However, when the number of agents goes to infinity, the large scale communication network's burden will be simultaneously increased. Hence, more researchers are actively developing a series of new methods to reduce the communication effort by introducing a new type of

on-demand style. In [17, 53], the authors redesigned the communication network as an event-driven system by setting an appropriate threshold and communication resource requests. Other researchers proposed reducing the communication burden by compressing the information and building multi-hop transmissions as [74, 12]. Another new type of technique focus on solving the multi-player pursuit-evasion game with reduced information exchange [60]. Besides the challenge from the communication network as well as information exchange, the notorious "***Curse of Dimensionality***" will lead to another issue while solving optimal large-scale multi-player pursuit-evasion strategy at the individual agent level, i.e., the computational complexity arising drastically along with the increasing number of agents. In many recent reinforcement learning-based intelligent MAS designs, including learning-based intelligent pursuit-evasion strategy development, only the finite discrete state space and/or action space has been considered [45, 11, 73, 33]. Those approaches cannot be directly utilized to large scale multi-player pursuit-evasion games since the dimension of agent/player's state space and/or action space may be exploded even close to infinite as the number of agents continuously grows large. For example, [67] introduced the states and control augmentation algorithm which transform the multi-agent tracking control problem into a regular optimal control problem. The dimension of the augmented states, however, increase with respect to the number of agents, which cause the increment of complexity. Similarly, Wang et al. [102] proposed a reinforcement learning method to solve the multi-pursuers versus single superior evader game. In [102], a ring topology and leader-follower line topology communication network is developed to further reduce communication and computational complexity. However, the computational complexity of the centralized global critic neural network in [102]'s design is still related to the agent number.


Figure 3.1: Problem formulation and challenges of multi-agent pursuit-evasion games.

To overcome these challenges, the Mean Field Games (MFG) [34, 48] has been utilized and further modified to apply to the multi-player pursuit-evasion game. In the MFG, individual agents will encode all other agents' states and actions as a form of probability density function (PDF) to avoid the dimension explosion while the number of agents is increasing. More importantly, the new form of PDF does not need to be acquired through communication. It can be obtained by solving a new type of partial differential equation (PDE) named the Fokker-Planck-Kolmogorov (FPK) equation [34] with local information. In large scale multi-player pursuit-evasion games, assuming that all agents/players with the same objective are homogeneous, the individual agent can estimate other agents' actions by substituting all agents' states in the new form of PDF into corresponding policy. Then,

these actions can be utilized further to update the time-varying PDF via the FPK equation. Since all agents' state distribution can be generated locally (except the initial distribution), the communication burden can be released significantly. In our previous work on the tracking control design [120], the general mean field game theory was applied to generate a decentralized online algorithm. However, while considering the large scale multi-player pursuit-evasion game, the previous algorithms are challenged in two different aspects, i.e. 1) The MFG cannot handle the case where the pursuers and the evaders have different dynamics, and 2) Unlike the traditional mean field game where all agents' dynamics are transparent, the pursuer team and the evader group formulate a differential game where no opponents' information is provided.

Recently, reinforcement learning and approximate dynamic programming (ADP) has been integrated appropriately to solve non-linear HJB and HJI equations forward-in-time, e.g., [51], [97]. In [66], an identifier-critic neural networks approach has been proposed to effectively solve the optimal control problem while estimating the unknown system dynamics. In [2], Abu-Khalaf, Murad, and F.L. Lewis have developed the actor-critic algorithm and proved the possibility of using it to approximate the solution of the HJB equation [2]. Similar to the HJI equation, the FPK equation is a partial differential equation as well. Therefore, a novel actor-critic-mass algorithm has been designed to obtain the decentralized optimal tracking control for massive MAS by online learning HJI as well as FPK using neural networks simultaneously.

In this paper, the multi-player pursuit-evasion game with massive pursuers and evaders has been considered. Specifically, the pursuers are operating non-cooperatively within the pursuers' group and so as the evaders. Meanwhile, the

pursuer group aims to intercept the evader group without knowing their dynamics and strategies. The Hamilton-Jacobi-Isaacs (HJI) equation is defined for both the pursuit and evasion groups to solve the large scale multi-player pursuit-evasion game. Moreover, each agent maintains a local FPK equation to approximate its own team's entire state PDF. By solving these coupled equations, the individual agent's optimal control strategy can be obtained, which features the Nash Equilibrium. However, solving those coupled partial differential equations (PDEs) analytically is nearly impossible [119, 35]. Therefore, we proposed the Actor-Critic-Mass-Opponent (ACMO) algorithm based on the emerging approximate dynamic programming (ADP) [51, 52] method to approximate the optimal solutions online.

The main contribution of this paper can be summarized as:

1. The traditional Mean Field Game theory has been extended and further apply to the large-scale multi-player pursuit-evasion game where the opponent group dynamics are heterogeneous and unknown.
2. The proposed Actor-Critic-Mass-Opponent algorithm is a decentralized algorithm that can solve the two common problems in the large-scale pursuit-evasion game as well as other multi-player games, i.e., the ***"Curse of Dimensionality"*** and the communication burden, by coding all other agents' information into two PDFs that are solvable locally.

## 3.2 Problem Formulation

The multi-player pursuit-evasion game with very large scale pursuit and evader groups is first introduced in this section. Then, the Mean Field Games theory has

been formulated.

### 3.2.1 Multi-Player Pursuit-evasion Game

In this subsection, the large scale multi-player pursuit-evasion game includes two competitive large groups of players, i.e., the pursuer group $\mathcal{G}_1$ and the evader group $\mathcal{G}_2$. Consider group $\mathcal{G}_1$ and $\mathcal{G}_2$ having $N_1$ and $N_2$ agents, respectively. The numbers $N_1$ and $N_2$ are two countably infinite numbers in the ideal case [34]. Next, the agents/players in two groups have heterogeneous stochastic nonlinear dynamics defined as:

Pursuer group $\mathcal{G}_1$ agents:

$$dx_{j,1}(t) = \begin{bmatrix} f_1\left(x_{j,1}\right) + l_1(x_{j,1})\bar{x}_2 \\ +g_1\left(x_{j,1}\right) u_{j,1}(t) \end{bmatrix} dt + D_1 dw_{j,1}(t), \quad 1 \leq j \leq N_1 \tag{3.1}$$

Evader group $\mathcal{G}_2$ agents:

$$dx_{i,2}(t) = \begin{bmatrix} f_2\left(x_{i,2}\right) + l_2(x_{i,2})\bar{x}_1 \\ +g_2\left(x_{i,2}\right) u_{i,2}(t) \end{bmatrix} dt + D_2 dw_{i,2}(t), \quad 1 \leq i \leq N_2 \tag{3.2}$$

where $x_{j,1} \in \mathbb{R}^l$ denotes the state of the $j$-th agent in group 1, $x_{i,2} \in \mathbb{R}^l$ denotes the state of the $i$-th agent in group 2, $x_1^{(N_1)}$ and $x_2^{(N_2)}$ are the set of all pursuers' and evaders' states respectively, $l_1(x_{j,1})$ and $l_2(x_{i,2})$ are smooth functions, $u_{j,1}$ and $u_{i,2}$ are the corresponding control input, $w_{j,1}$ and $w_{i,2}$ mark independent Brownian noise, $D_1$ and $D_2$ are the coefficient matrices for the noise term. The functions $f_1(\cdot)$, $f_2(\cdot)$, $g_1(\cdot)$, $g_2(\cdot)$ are the nonlinear intrinsic dynamic equations of the pursuers and evaders.

The goal of the pursuers is to capture the evaders while the evaders are trying

their best run away from. In previous pursuit-evasion game studies [28, 65], the authors defined point capture, where the positions of the pursuers and evaders are same, as the sign of a success capture. Considering the large scale pursuit and evader agents in this paper, the point capture can be extended to the *mass capture*, where the pursuers' the average position, instead of every single points, has to be the same with the average evaders' position. In other words, when the condition $1/N_1 \sum x_1 = 1/N_2 \sum x_2$ is satisfied, the pursuers' group has been considered as capturing the evaders' group successfully. Following this goal, a cost function can be defined to evaluate the performance of each agents in both pursuit and evader groups, i.e.,

$$\mathcal{G}_1\text{: } V_{j,1}\left(x_{j,1}, x_1^{(N_1)}, x_2^{(N_2)}\right) = \mathbb{E}\left\{\int_0^\infty \left[\begin{array}{c} \Phi_{Q1}(x_{j,1}, x_1^{(N_1)}) + \Phi_{Q2}(x_{j,1}, x_2^{(N_2)}) \\ + \|u_{j,1}(t)\|_{R_1}^2 \end{array}\right] dt\right\} \tag{3.3}$$

$$\mathcal{G}_2\text{: } V_{i,2}\left(x_{i,2}, x_1^{(N_1)}, x_2^{(N_2)}\right) = \mathbb{E}\left\{\int_0^\infty \left[\begin{array}{c} \Phi_{Q3}(x_{i,2}, x_2^{(N_2)}) - \Phi_{Q4}(x_{i,2}, x_1^{(N_1)}) \\ + \|u_{i,2}(t)\|_{R_2}^2 \end{array}\right] dt\right\} \tag{3.4}$$

where the function $\Phi(\cdot)$ denotes the coupling effect between the pursuit and evader groups, $\|Z\|_C$ denotes the quadratic term, i.e., $\|Z\|_C = Z^T C Z$ . In the case of large scale multi-player pursuit-evasion game, we consider the coupling functions as the euclidean distances [28]. Guided by the *mass capture* principle, the group effect functions are defined as the group state average, i.e., $1/N \sum_i x_i$. Then, the coupling function can be represented as:

$$\Phi_Q(Z_i, Z^{(N)}) = \left\| Z_i - \frac{1}{N}\sum_{i=1}^{N} Z_i \right\|_Q$$

with the notation $\|Z\|_Q$ being the weighted norm of a vector, i.e., $Z^T Q Z$. The other terms are quadratic terms similarly designed as other differential game researches,

e.g. [97, 93].

To achieve the optimal strategy, each agent/player in pursuit-evasion game aims to minimize its cost function while the opponents are maximizing it, i.e.,

For pursuers in $\mathcal{G}_1$ :

$$V_{j,1}(x_{j,1}, x_1^{(N_1)}, x_2^{(*N_2)}) = \min_{u_{j,1}} \max_{\Phi_{Q2}(x_{j,1}, x_2^{(*N_2)})} \mathbb{E}\left\{ \int_0^\infty \begin{bmatrix} \Phi_{Q1}(x_{j,1}, x_1^{(N_1)}) \\ +\Phi_{Q2}(x_{j,1}, x_2^{(*N_2)}) \\ +\|u_{j,1}(t)\|_{R_1}^2 \end{bmatrix} dt \right\} \quad (3.5)$$

For evaders in $\mathcal{G}_2$ :

$$V_{i,2}(x_{i,2}, x_1^{(*N_1)}, x_2^{(N_2)}) = \min_{u_{i,2}} \max_{\Phi_{Q4}(x_{i,2}, x_1^{(*N_1)})} \mathbb{E}\left\{ \int_0^\infty \begin{bmatrix} \Phi_{Q3}(x_{i,2}, x_2^{(N_2)}) \\ -\Phi_{Q4}(x_{i,2}, x_1^{(*N_1)}) \\ +\|u_{i,2}(t)\|_{R_2}^2 \end{bmatrix} dt \right\} \quad (3.6)$$

where the terms $x_1^{(*N_1)}$ and $x_2^{(*N_2)}$ represent the average states with $V_{j,1}$ and $V_{i,2}$ are being maximized, respectively. The coupling functions $\Phi_{Q3}(x_{i,2}, x_2^{(N_1)})$ and $\Phi_{Q1}(x_{j,1}, x_1^{(N_1)})$ denote the influence level from agent's own group to keep agent stay in the same group and gather around the group mass while the agent minimizing the cost. The function $\Phi_{Q2}(x_{j,1}, x_2^{(N_2)})$ in (3.3) leads each pursuer to reduce the distance between itself and evaders' mass center. On the other hand, the function $\Phi_{Q4}(x_{i,2}, x_1^{(N_1)})$ in (3.4) forces the evaders to keep away from the pursuers' mass center to prevent being captured. Next, two sets of optimal control strategies can be defined as $\Omega_1(t) = \{u_{1,1}^*(t), u_{2,1}^*(t), \cdots, u_{N_1,1}^*(t)\}$ and $\Omega_2(t) = \{u_{1,2}^*(t), u_{2,2}^*(t), \cdots, u_{N_1,2}^*(t)\}$ as the solution to the cost functions (3.5) or (3.6). The goal of pursuit-evasion game is to find such optimal control strategy sets.

However, the main issue arises with the coupling term in the cost functions (3.3) and (3.4) where large scale agents' information $x_1^{(N_1)}$ and $x_2^{(N_2)}$ will increase the dimension of coupling function. When the number of agents goes to infinity, the

computational complexity of cost function arises drastically due to the coupling function. Moreover, the coupling function also requires other agents' real-time information in order to evaluate the cost function that is another stringent constraint since large-scale real-time communication network is not always available and reliable [99]. To overcome these practical challenges, the Mean Field Game (MFG) [30, 48, 34] is utilized to form a mean field type of decentralized control.

### 3.2.2 Mean Field Games Formulation

The main characteristics of the Mean Field Game theory include: 1) Reduce the dimension of the information flow between each agent by encoding the state information of large scale agents as a time-varying probability density function (PDF), and 2) Eliminate the communications by computing the time-varying PDF locally.

Recall the Mean Field Game theory [30], let $m_1(x_1, t)$ and $m_2(x_{i,2}, t)$ denote the time-varying PDF of the agents states in both pursuit and evader teams as $\mathcal{G}_1$ and $\mathcal{G}_2$ respectively. Moreover, denote $\mathbb{E}\{m_2^*\}$ and $\mathbb{E}\{m_1^*\}$ as the average states of opponents when the team reaches maximum cost, i.e., the worst case to the team. Then, the cost functions (3.3) and (3.4) can be loosely rewritten as:

$$\mathcal{G}_1\text{: } V_{j,1}\left(x_{j,1}, m_1, \mathbb{E}\{m_2^*\}\right) = \mathbb{E}\left\{\int_0^\infty \left[\begin{array}{c} \Phi_{Q1}(x_{j,1}, m_1) + \Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\}) \\ + \|u_{j,1}(t)\|_{R_1}^2 \end{array}\right] dt\right\} \tag{3.7}$$

$$\mathcal{G}_2\text{: } V_{i,2}\left(x_{i,2}, \mathbb{E}\{m_1^*\}, m_2\right) = \mathbb{E}\left\{\int_0^\infty \left[\begin{array}{c} \Phi_{Q3}(x_{i,2}, m_2) - \Phi_{Q4}(x_{i,2}, \mathbb{E}\{m_1^*\}) \\ + \|u_{i,2}(t)\|_{R_2}^2 \end{array}\right] dt\right\} \tag{3.8}$$

where the group average states can be represented through the PDF $m_1$ and $m_2$ as the expected value as

$$\frac{1}{N_1}\sum_{j=1}^{N_1} x_{j,1} = \mathbb{E}\{m_1(x_1,t)\}$$
$$\frac{1}{N_2}\sum_{i=1}^{N_2} x_{i,2} = \mathbb{E}\{m_2(x_{i,2},t)\}$$

The replaced coupling functions are now named as the mean field coupling functions where the computational complexity is greatly reduced since the value of the introduced PDF has the same dimension as system states. Intuitively, the Mean Field Games transformed the large scale multi-player pursuit-evasion (PE) game into two *mass players* PE game where each mass player has a mean-field type of stochastic states. More importantly, the PDF used in the coupling function can be computed through a locally solvable partial differential equation (PDF), i.e. Fokker Planck Kolmogorov (FPK) equation [34], which will be introduced in the next section. Since the PDF can be calculated by using local information only, the communication among large scale pursuers and evaders can be eliminated as well.

## 3.3 Methodologies

Due to the difficulties from the coupling terms introduced in the last section, the Mean Field type of control is introduced first. Moreover, a neural network based online solver for the Mean Field type of control will also be derived. Since the pursuers and evaders share the similar cost function and dynamics, we will introduce the Actor-Critic-Mass-Opponent (ACMO) control for pursuers only in this section. Without loss of generality, the evaders' control can be obtained by easily replacing relevant parameters in pursuer design.

$$\text{HJI-}\mathcal{G}_1: \begin{array}{l} -\dfrac{\partial V_{j,1}\left(x_{j,1}, m_1, \mathbb{E}\{m_2^*\}\right)}{\partial t} - \dfrac{D_1^2}{2}\Delta V_{j,1}\left(x_{j,1}, m_1, \mathbb{E}\{m_2^*\}\right) \\ + H_{j,1}\left(x_{j,1}, \dfrac{\partial V_{j,1}\left(x_{j,1}, m_1, \mathbb{E}\{m_2^*\}\right)}{\partial x_{j,1}}\right) = 0 \end{array} \tag{3.9}$$

$$\text{FPK-}\mathcal{G}_1: \begin{array}{l} \dfrac{\partial m_1\left(x_1, t\right)}{\partial t} - \dfrac{D_1^2}{2}\Delta m_1\left(x_1, t\right) \\ -\operatorname{div}\left[m_1 D_p H_{j,1}\left(x_{j,1}, \dfrac{\partial V_{j,1}\left(x_{j,1}, m_1, \mathbb{E}\{m_2^*\}\right)}{\partial x_{j,1}}\right)\right] = 0 \end{array} \tag{3.10}$$

$$\text{HJI-}\mathcal{G}_2: \begin{array}{l} -\dfrac{\partial V_{i,2}\left(x_{i,2}, \mathbb{E}\{m_1^*\}, m_2\right)}{\partial t} - \dfrac{D_2^2}{2}\Delta V_{i,2}\left(x_{i,2}, \mathbb{E}\{m_1^*\}, m_2\right) \\ + H_{i,2}\left(x_{i,2}, \dfrac{\partial V_{i,2}\left(x_{i,2}, \mathbb{E}\{m_1^*\}, m_2\right)}{\partial x_{i,2}}\right) = 0 \end{array} \tag{3.11}$$

$$\text{FPK-}\mathcal{G}_2: \begin{array}{l} \dfrac{\partial m_2\left(x_2, t\right)}{\partial t} - \dfrac{D_2^2}{2}\Delta m_2\left(x_2, t\right) \\ -\operatorname{div}\left[m_2 D_p H_{i,2}\left(x_{i,2}, \dfrac{\partial V_{i,2}\left(x_{i,2}, \mathbb{E}\{m_1^*\}, m_2\right)}{\partial x_{i,2}}\right)\right] = 0 \end{array} \tag{3.12}$$

$$m_1\left(x_1, 0\right) = m_{1,0}\left(x_1\right)$$
$$m_2\left(x_2, 0\right) = m_{2,0}\left(x_2\right)$$

---

### 3.3.1 The Mean Field Type of Optimal Control for Pursue Evasion Games

The objective of a mean field type of optimal control for a pursuer is to obtain a set of control $\Omega(t) = \{u_{1,1}^*(t), u_{2,1}^*(t), \cdots, u_{N_1,1}^*(t)\}$ such that pursuer group's cost functions (3.7) can no longer be minimized. This equilibrium is also known as the Nash Equilibrium in game theory [9]. According to the traditional optimal control theory [50] and differential games [97], one can derive the Hamiltonian-Jacobi-Isaacs (HJI) equation for individual agent in the group $\mathcal{G}_1$ from the cost function (3.7) as (3.9) where $H_{j,1}(\cdot)$ is the Hamiltonian defined as:

$$H_{j,1}\left[x_{j,1}, \partial_{x_{j,1}} V_{j,1}(x_{j,1}, m_1, \mathbb{E}\{m_2^*\})\right] = \|u_{j,1}\|_{R_1} + \Phi_{Q1}(x_{j,1}, m_1) + \Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\})$$

$$+ \partial_{x_{j,1}} V_{j,1}(x_{j,1}, m_1, \mathbb{E}\{m_2^*\})^T \begin{bmatrix} f_1\left(x_{j,1}\right) \\ +g_1\left(x_{j,1}\right) u_{j,1} \\ +l_1(x_{j,1})\mathbb{E}\{m_2^*\} \end{bmatrix}$$

with the $m_2^*$ being evaders' states' PDF that aims to cause the greatest cost on pursuers.

The solution to the group $\mathcal{G}_1$'s HJI equation yields the minimum cost $V_{j,1}(x_{j,1}, m_1, \mathbb{E}\{m_2^*\})$ and the corresponding optimal control strategy which can be derived as:

$$u_{j,1}^*(x_{j,1}, m_1, \mathbb{E}\{m_2^*\}) = -\frac{1}{2} R_1^{-1} g_1^T(x_{j,1}) \frac{\partial V_{j,1}^*(x_{j,1}, m_1, \mathbb{E}\{m_2^*\})}{\partial x_{j,1}} \tag{3.13}$$

The optimal solution to the cost function (3.5) features a saddle point where the maxima is the optimal control (3.13) and the minima represents $m_2^*$. According to the mean field coupling function, the expected value $\mathbb{E}\{m_2\}$ instead the PDF $m_2$ is used. Thus, to calculate the minima $m_2^*$, we take partial derivative on the Hamiltonian with respect to $\mathbb{E}\{m_2\}$ and let it equal to zero, which yields:

$$\Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\}) - \frac{1}{2} Q_2^{-1} l_1^T(x_{j,1}) \frac{V_{j,1}(x_{j,1}, m_1, \mathbb{E}\{m_2^*\})}{\partial x_{j,1}} = 0 \tag{3.14}$$

Similarly, one obtains the HJI equation for the evaders as (3.11). Moreover, the corresponding Hamiltonian is defined as:

$$H_{i,2}\left[x_{i,2}, \partial_{x_{i,2}} V_{i,2}(x_{i,2}, \mathbb{E}\{m_1^*\}, m_2)\right] = \|u_{i,2}\|_{R_2} + \Phi_{Q3}(x_{i,2}, m_2) - \Phi_{Q4}(x_{i,2}, \mathbb{E}\{m_1^*\})$$
$$+ \partial_{x_{i,2}} V_{i,2}(x_{i,2}, \mathbb{E}\{m_1^*\}, m_2)^T \begin{bmatrix} f_2\left(x_{i,2}\right) \\ +g_2\left(x_{i,2}\right) u_{i,2} \\ +l_2(x_{i,2})\mathbb{E}\{m_1^*\} \end{bmatrix}$$

The saddle point can be computed similar to the pursuers.

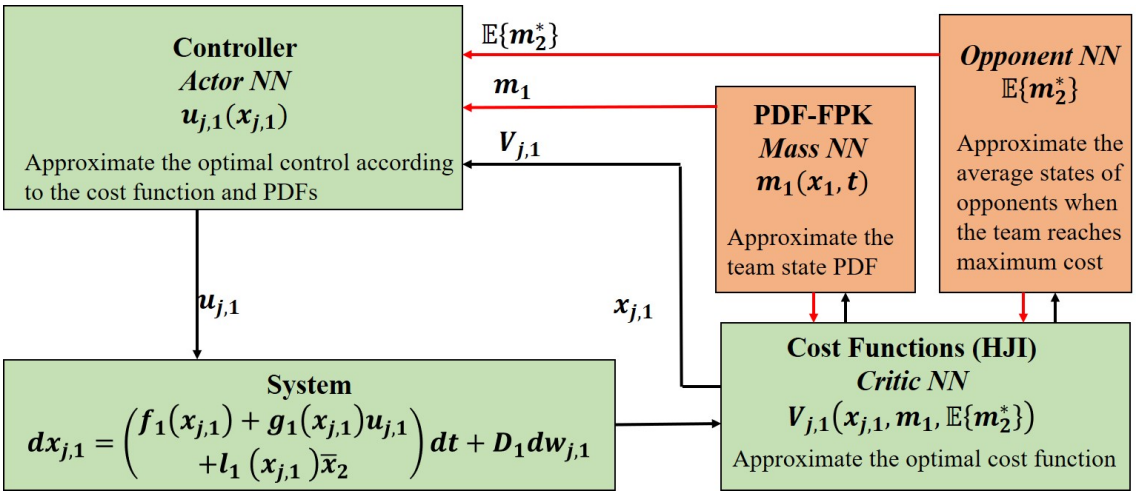



Figure 3.2: The structure of the proposed Actor-Critic-Mass-Opponent algorithm. The four neural networks are presented.

In the optimal control (3.13) for any pursuer, the PDF of the pursuer group $m_1$ remains unknown. In the MFG [13], the group PDF can be computed by the Fokker-Plank-Kolmogorov equation. In this problem, the FPK equations for both groups are given in (3.10) and (3.12). For a pursuer, the HJI (3.9) and FPK (3.9) equation form a coupled partial differential equation (PDE) named the mean field equation systems whose solutions provide the Nash Equilibrium (NE) control strategy for all players, i.e., $\Omega_1$. However, the mean field equation systems are difficult and nearly impossible to be solved analytically due to the complexity of two coupled high-dimensional PDEs [35]. Therefore, in this paper, we extend the Actor-Critic structure from approximate dynamic programming (ADP) [52] to a novel Actor-Critic-Mass-Opponent (ACMO) structure that can learn the solutions numerically.

### 3.3.2 The ACMO Neural Network Estimators

As the challenges rising in solving two coupled PDEs, i.e., HJI-FPK, the emerging neural network estimators are proposed to overcome this difficulty. In this section, the proposed Actor-Critic-Mass-Opponent (ACMO) algorithm for the pursuers' group $\mathcal{G}_1$ is explained in detail. The evaders' group is similarly developed and thus omitted. In the proposed ACMO, each agent maintains four neural networks, i.e., the actor neural network to approximate the optimal control, critic neural network to approximate the optimal cost function, and two mass neural networks to approximate the PDF for group $\mathcal{G}_1$ and $\mathcal{G}_2$ respectively. Figure 3.2 demonstrates the structure of the ACMO algorithm as well. In Fig. 3.2, four neural networks are constructed, i.e.,

- the actor neural network is designed to approximate the optimal control.
- the critic neural network is designed to approximate the optimal cost function.
- the mass neural network is designed to approximate the PDF of the pursuers' team.
- the opponent neural network is designed to approximate the average states of the opponents when the team reaches maximum cost, i.e., the worst case for the team.

The four neural networks are coupled through the HJI and FPK equations. With the mild assumption that there exists constant neural network weights, i.e., $W_{V,j,1}$, $W_{u,j,1}$, $W_{m,1}$, $W_{m,2}$, and activation functions $\phi_{V,j,1}, \phi_{u,j,1}, \phi_{m,1}, \phi_{m,j,2}$ such that the optimal cost function, optimal control, and the PDF for pursuer and evader groups

can be written as:

$$
\begin{aligned}
&V_{j,1}(x_{j,1}, m_1, \Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\})) = W_{V,j,1}^T \phi_{V,j,1}(x_{j,1}, m_1, \Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\})) + \varepsilon_{V,j,1} \\
&u_{j,1}(x_{j,1}, m_1, \Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\})) = W_{u,j,1}^T \phi_{u,j,1}(x_{j,1}, m_1, \Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\})) + \varepsilon_{u,j,1} \\
&m_1(x_1, t) = W_{m,j,1}^T \phi_{m,j,1}(x_{j,1}, t) + \varepsilon_{m,j,1} \\
&\Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\}) = W_{m,j,2}^T \phi_{m,j,2}(x_{j,1}, m_1) + \varepsilon_{m,j,2}
\end{aligned}
\tag{3.15}
$$

where the activation functions $\phi_{V,j,1}, \phi_{u,j,1}, \phi_{m,1}, \phi_{m,j,2}$ are bounded and continuous, $\varepsilon_{V,j,1}, \varepsilon_{u,j,1}, \varepsilon_{m,j,1}, \varepsilon_{m,j,2}$ are the reconstruction error of the corresponding neural networks. The weights $W_{V,j,1}^T, W_{u,j,1}^T, W_{m,j,1}^T, W_{m,j,2}^T$ are unknown and expected to be learnt. Let the approximated weights be denoted as $\hat{W}_{V,j,1}^T, \hat{W}_{u,j,1}^T, \hat{W}_{m,j,1}^T, \hat{W}_{m,j,2}^T$, the optimal control and cost functions can be approximated as:

$$
\begin{aligned}
&\hat{V}_{j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2}) = \hat{W}_{V,j,1}^T \phi_{V,j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2}) \\
&\hat{u}_{j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2}) = \hat{W}_{u,j,1}^T \phi_{u,j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2}) \\
&\hat{m}_{j,1}(x_{j,1}, t) = \hat{W}_{m,j,1}^T \phi_{m,j,1}(x_{j,1}, t)\big|_{V_{j,1}=\hat{V}_{j,1}} \\
&\hat{x}_{j,2} = \hat{W}_{m,j,2}^T \phi_{m,j,2}(x_{j,1}, \hat{m}_{j,1})\big|_{V_{j,1}=\hat{V}_{j,1}}
\end{aligned}
\tag{3.16}
$$

where $\hat{x}_{j,2}$ represents $j$-th agent's estimated coupling cost, i.e., $\hat{x}_{j,2} = \Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\})$.

Next, substituting the estimation equations (3.16), the mean field equations, i.e., (3.9), (3.10) as well as the optimal control equation (3.13), the worst mass from the evaders (3.14) will not hold. The resulting errors are used to tune the neural network weights, i.e.,

$$
e_{HJI,j} = \Phi_{Q1}(x_{j,1}, \hat{m}_{j,1}) + \hat{x}_{j,2} + \hat{W}_{V,j,1}^T(t) \begin{bmatrix} \frac{\partial \phi_{V,j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2})}{\partial t} - \hat{H}_{WV} \\ + \frac{D_1^2}{2} \Delta \phi_{V,j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2}) \end{bmatrix} \tag{3.17}
$$

$$
e_{FPK,j} = \hat{W}_{m,j,1}^T(t) \begin{bmatrix} \frac{\partial \phi_{m,j,1}(x_{j,1}, t)}{\partial t} - \frac{D_1^2}{2} \Delta \phi_{m,j,1}(x_{j,1}, t) \\ - \operatorname{div}(\phi_{m,j,1}(x_{j,1}, t)\, \mathrm{D_p}\, \hat{H}_j) \end{bmatrix} \tag{3.18}
$$

where

$$\hat{H}_j = H_{j,1}\left[x_{j,1}, \partial_{x_{j,1}} V(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2})\right] - \Phi_{Q1}(x_{j,1}, \hat{m}_{j,1}) - \hat{x}_{j,2}$$

and

$$\hat{W}_{V,j,1}(t)\hat{H}_{WV} = \hat{H}_j$$

Moreover, substituting the estimation equation (3.16) into the optimal control equation (3.13) and the worst evaders' expected states (3.14), one obtains:

$$e_{u,j} = \hat{W}_{u,j,1}^T(t)\phi_{u,j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2}) + \frac{1}{2}R_1^{-1}g_1^T(x_{j,1})\frac{\partial V_{j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2})}{\partial x_{j,1}} \tag{3.19}$$

$$e_{m2,j} = \hat{x}_{j,2} - \frac{1}{2}Q_2^{-1}l_1^T(x_{j,1})\frac{\partial V_{j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{W}_{m,j,2}^T\phi_{m,j,2}(x_{j,1}, t))}{\partial x_{j,1}} \tag{3.20}$$

Denote

$$\Psi_{V,j}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2}) = \begin{bmatrix} \frac{\partial \phi_{V,j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2})}{\partial t} - \hat{H}_{WV} \\ +\frac{D_1^2}{2}\Delta\phi_{V,j,1}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2}) \end{bmatrix}$$

$$\Psi_{m,j}(x_{j,1}, \hat{V}_{j,1}, \hat{x}_{j,2})) = \begin{bmatrix} \frac{\partial \phi_{m,j,1}(x_{j,1}, t)}{\partial t} - \frac{D_1^2}{2}\Delta\phi_{m,j,1}(x_{j,1}, t) \\ -\operatorname{div}(\phi_{m,j,1}(x_{j,1}, t)\,\mathrm{D_p}\,\hat{H}_j) \end{bmatrix}$$

$$\tilde{\Phi} = \Phi_{Q1}(x_{j,1}, m_1) - \Phi_{Q1}(x_{j,1}, \hat{m}_{j,1}) + \Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\}) - \hat{x}_{j,2}$$

The neural networks' estimation errors (3.17), (3.18), (3.19), (3.20) can be written as:

$$e_{HJI,j} = \Phi_{Q1}(x_{j,1}, \hat{m}_{j,1}) + \hat{x}_{j,2} + \hat{W}_{V,j,1}^T(t)\Psi_{V,j}(x_{j,1}, \hat{m}_{j,1}, \hat{x}_{j,2}) \tag{3.21}$$

$$e_{FPK,j} = \hat{W}_{m,j,1}^T(t)\Psi_{m,j}(x_{j,1}, \hat{V}_{j,1}, \hat{x}_{j,2}) \tag{3.22}$$

Finally, the update law for the four neural networks can be derived by applying gradient descent at the error functions (3.17), (3.18), (3.19), and (3.20) as,

$$\dot{\hat{W}}_{V,j,1} = -\alpha_{h,j}\frac{\Psi_{V,j}(x_{j,1},\hat{m}_{j,1},\hat{x}_{j,2})e_{HJI,j}^T}{1+\|\Psi_{V,j}(x_{j,1},\hat{m}_{j,1},\hat{x}_{j,2})\|^2} \tag{3.23}$$

$$\dot{\hat{W}}_{u,j,1} = -\alpha_{u,j}\frac{\phi_{u,j}(x_{j,1},\hat{m}_{j,1},\hat{x}_{j,2})e_{u,j}^T}{1+\|\phi_{u,j}(x_{j,1},\hat{m}_{j,1},\hat{x}_{j,2})\|^2} \tag{3.24}$$

$$\dot{\hat{W}}_{m,j,1} = -\alpha_{m,j}\frac{\Psi_{m,j}(x_{j,1},\hat{V}_{j,1},\hat{x}_{j,2})e_{FPK,j}^T}{1+\|\Psi_{m,j}(x_{j,1},\hat{V}_{j,1},\hat{x}_{j,2})\|^2} \tag{3.25}$$

$$\dot{\hat{W}}_{m,j,2} = -\alpha_{m2,j}\frac{\phi_{m,j,2}(x_{j,1},\hat{m}_{j,1})e_{m2,j}^T}{1+\|\phi_{m,j,2}(x_{j,1},\hat{V}_{j,1},\hat{x}_{j,2})\|^2} \tag{3.26}$$

where $\alpha_{h,j}$, $\alpha_{u,j}$, $\alpha_{m,j}$, $\alpha_{m2,j}$ are learning rates.

### 3.3.3 The ACMO Neural Networks' Performance Analysis

According to the weight update laws (3.23), (3.24), (3.25), and (3.26), we obtain the first derivative of neuron networks' estimation errors as:

$$\dot{\tilde{W}}_{V,j,1} = -\dot{\hat{W}}_{V,j,1} = \alpha_{h,j}\frac{\Psi_{V,j}(x_{j,1},\hat{m}_{j,1},\hat{x}_{j,2})e_{HJI,j}^T}{1+\|\Psi_{V,j}(x_{j,1},\hat{m}_{j,1},\hat{x}_{j,2})\|^2} \tag{3.27}$$

$$\dot{\tilde{W}}_{u,j,1} = -\dot{\hat{W}}_{u,j,1} = \alpha_{u,j}\frac{\phi_{u,j}(x_{j,1},\hat{m}_{j,1},\hat{x}_{j,2})e_{u,j}^T}{1+\|\phi_{u,j}(x_{j,1},\hat{m}_{j,1},\hat{x}_{j,2})\|^2} \tag{3.28}$$

$$\dot{\tilde{W}}_{m,j,1} = -\dot{\hat{W}}_{m,j,1} = \alpha_{m,j}\frac{\Psi_{m,j}(x_{j,1},\hat{V}_{j,1},\hat{x}_{j,2})e_{FPK,j}^T}{1+\|\Psi_{m,j}(x_{j,1},\hat{V}_{j,1},\hat{x}_{j,2})\|^2} \tag{3.29}$$

$$\dot{\tilde{W}}_{m,j,2} = -\dot{\hat{W}}_{m,j,2} = \alpha_{m2,j}\frac{\phi_{m,j,2}(x_{j,1},\hat{m}_{j,1})e_{m2,j}^T}{1+\|\phi_{m,j,2}(x_{j,1},\hat{V}_{j,1},\hat{x}_{j,2})\|^2} \tag{3.30}$$

Next, the performance of all neural networks is given in the following theorems.

**Theorem 6.** *(Critic NN's convergence)* Let $\hat{W}_{V,j,1}(t)$ be updated as (3.23) shows and assume the learning rate $\alpha_{h,j} > 0$, then the critic NN's weights estimation error

$\tilde{W}_{V,j,1}$, and the optimal cost function approximation error, i.e., $\tilde{V}_{j,1} = V_{j,1} - \hat{V}_{j,1}$ are uniformly ultimately bounded (UUB). The corresponding bounds $b_{W_{V,j}}$, $b_{V,j}$ are trivial when the reconstruction error is sufficiently small [91]. Moreover, $\tilde{W}_{V,j,1}$ and $\tilde{V}_{j,1}$ are also asymptotically stable if the neuron network structure is selected perfectly.

*Proof.* Because all agents are homogeneous in the same group, we drop the subscript of the agent number $i$ and make the following simplifications on the notations:

$x_{j,1} \to x_1$, $f_1(x_{j,1}) \to f_1(x_1)$, $g_1(x_{gj,1}) \to g_1(x_1)$, $u_{j,1} \to u$, $W_{V,j,1} \to W_{V1}$, $W_{m,j,1} \to W_{m1}$, $W_{u,j,1} \to W_{u1}$, $\alpha_{h,j} \to \alpha_h$, $\hat{m}_{j,1} \to \hat{m}_1$, $\hat{V}_{j,1} \to \hat{V}_1$, $V_{j,1} \to V_1$, $\hat{u}_{j,1} \to \hat{u}_1$, $u_{j,1} \to u_1$, $e_{HJI,j} \to e_{HJI1}$, $e_{u,j} \to e_{u1}$, $e_{FPK,j} \to e_{FPK1}$, $\varepsilon_{V,j,1} \to \varepsilon_{HJI1}$, $\varepsilon_{u,j,1} \to \varepsilon_{u1}$, $\varepsilon_{m,j,1} \to \varepsilon_{FPK1}$, $D_1 \to \sigma_1$, $dw_{j,1} \to dw_1$,

The following notation simplifications are similar to above but for the evaders' approximators. These simplifications are valid through all proofs in this chapter.

$f_2(x_{i,2}) \to f_2(x_2)$, $g_2(x_{i,2}) \to g_2(x_2)$, $u_{i,2} \to u$, $x_{i,2} \to x_2$, $W_{V,i,2} \to W_{V2}$, $W_{m,i,2} \to W_{m2}$, $W_{u,i,2} \to W_{u2}$, $\alpha_{h,i} \to \alpha_h$, $\hat{m}_{i,2} \to \hat{m}_2$, $m_{i,2} \to m_2$, $\hat{V}_{i,2} \to \hat{V}_2$, $V_{i,2} \to V_2$, $\hat{u}_{i,2} \to \hat{u}_2$, $u_{i,2} \to u_2$, $e_{HJI2,i} \to e_{HJI2}$, $e_{u2,i} \to e_{u2}$, $e_{FPK2,i} \to e_{FPK2}$, $\varepsilon_{V,i,2} \to \varepsilon_{HJI2}$, $\varepsilon_{u,i,2} \to \varepsilon_{u2}$, $\varepsilon_{m,i,2} \to \varepsilon_{FPK2}$, $D_2 \to \sigma_2$, $dw_{i,2} \to dw_2$.

The above notations are also valid through the rest of proofs in this chapter.

Consider the following Lyapunov function candidate as:

$$L_{V1}(t) = \frac{1}{2} \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\tilde{W}_{V1}(t)\right\} \tag{3.31}$$

Take the first derivative on the Lyapunov function candidate, one obtains:

$$\dot{L}_{V1}(t) = \frac{1}{2}\,\mathrm{tr}\left\{\tilde{W}_{V1}^{T}(t)\dot{\tilde{W}}_{V1}(t)\right\} + \frac{1}{2}\,\mathrm{tr}\left\{\dot{\tilde{W}}_{V1}^{T}(t)\tilde{W}_{V1}(t)\right\} = \mathrm{tr}\left\{\tilde{W}_{V1}^{T}(t)\dot{\tilde{W}}_{V1}(t)\right\} \tag{3.32}$$

Substitute the critic NN weights update law into (3.32), we get

$$\dot{L}_{V}1(t) = \alpha_h\,\mathrm{tr}\left\{\tilde{W}_{V1}^{T}(t)\frac{\hat{\Psi}_{V1}\left(x_1,\hat{m}_1,\hat{m}_2\right)e_{HJI1}^{T}}{1+\hat{\Psi}_{V1}^{T}\left(x_1,\hat{m}_1,\hat{m}_2\right)\hat{\Psi}_{V1}\left(x_1,\hat{m}_1,\hat{m}_2\right)}\right\} \tag{3.33}$$

Let

$$\Phi(x_1,\hat{m}_1,\hat{m}_2) = \Phi_{Q1}\left(\hat{m}_1,x_1\right) + \hat{x}_2$$

$$\tilde{\Phi}(x_1,m_1,m_2,\hat{m}_1,\hat{m}_2) = \hat{\Phi}(x_1,\hat{m}_1,\hat{m}_2) - \Phi(x_1,m_1,m_2)$$

Substitute $\tilde{\Phi}(x_1,m_1,m_2,\hat{m}_1,\hat{m}_2)$ into critic NN's error function (3.17), we get

$$\Phi(x_1,m_1,m_2) + \tilde{\Phi}(x_1,m_1,m_2,\hat{m}_1,\hat{m}_2) + \hat{W}_{V1}^{T}(t)\hat{\Psi}_{V1}\left(x_1,\hat{m}_1,\hat{m}_2\right) = e_{HJI1} \tag{3.34}$$

Since the correct estimated optimal cost function leads to the HJI equation equals zero, we have

$$\Phi(x_1,m_1,m_2) + W_{V}^{T}(t)\Psi_{V1}\left(x_1,m_1,m_2\right) = 0 \tag{3.35}$$

Substitute (3.35) into (3.34), we have

$$\begin{aligned} e_{HJI1} = &- W_{V}^{T}(t)\Psi_{V1}\left(x_1,m_1,m_2\right) - \varepsilon_{HJI1} \\ &+ \tilde{\Phi}(x_1,m_1,m_2,\hat{m}_1,\hat{m}_2) - \hat{W}_{V1}^{T}(t)\hat{\Psi}_{V1}\left(x_1,\hat{m}_1,\hat{m}_2\right) \end{aligned} \tag{3.36}$$

Let $\tilde{W}_{V1}(t) = W_V(t) - \hat{W}_{V1}(t)$, and $\tilde{\Psi}_{V1}(x_1,m_1,m_2,\hat{m}_1,\hat{m}_2) = \Psi_V(x_1,m_1,m_2) - \hat{\Psi}_{V1}(x_1,\hat{m}_1,\hat{m}_2)$. After manipulating terms in (3.36), we obtain

$$e_{HJI1} = - W_{V}^{T}(t)\left(\hat{\Psi}_{V1}\left(x_1,\hat{m}_1,\hat{m}_2\right) + \tilde{\Psi}_{V1}(x_1,m_1,m_2,\hat{m}_1,\hat{m}_2)\right)$$

$$- \varepsilon_{HJI1} + \tilde{\Phi}(x_1, m_1, m_2, \hat{m}_1, \hat{m}_2) + \hat{W}_{V1}^T(t)\hat{\Psi}_{V1}(x_1, \hat{m}_1, \hat{m}_2)$$

where $\varepsilon_{HJI1}$ is the error resulted from the reconstruction error. Next, the error can be written as

$$\begin{aligned} e_{HJI1} =& \tilde{\Phi}(x_1, m_1, m_2, \hat{m}_1, \hat{m}_2) - \tilde{W}_V^T\hat{\Psi}_{V1}(x_1, \hat{m}_1, \hat{m}_2) \\ & - W_V^T\tilde{\Psi}_{V1}(x_1, m_1, m_2, \hat{m}_1, \hat{m}_2) - \varepsilon_{HJI1} \end{aligned} \tag{3.37}$$

Let's further simplify the notations as: $\hat{\Psi}_{V1}(x_1, \hat{m}_1, \hat{m}_2) \to \hat{\Psi}_{V1}$, $\tilde{\Psi}_{V1}(x_1, m_1, m_2, \hat{m}_1, \hat{m}_2) \to \tilde{\Psi}_{V1}$, $\Psi_{V1}(x_1, m_1, m_2) \to \Psi_{V1}$, $\tilde{\Phi}(x_1, m_1, m_2, \hat{m}_1, \hat{m}_2) \to \tilde{\Phi}$

Substitute (3.37) into (3.33),

$$\begin{aligned} \dot{L}_{V1}(t) &= \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\left[\tilde{\Phi} - \tilde{W}_V^T\hat{\Psi}_{V1} - W_V^T\tilde{\Psi}_{V1} - \varepsilon_{HJI1}\right]^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}\right\} \\ &= \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\tilde{\Phi}^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}\right\} - \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\hat{\Psi}_{V1}^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}\tilde{W}_{V1}(t)\right\} \\ &- \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\tilde{\Psi}_{V1}^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}W_{V1}^T(t)\right\} - \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\varepsilon_{HJI1}^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}\right\} \end{aligned} \tag{3.38}$$

Apply Cauchy-Schwarz inequality on (3.38),

$$\begin{aligned} \dot{L}_{V1}(t) &= \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\tilde{\Phi}^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}\right\} - \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\hat{\Psi}_{V1}^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}\tilde{W}_{V1}(t)\right\} \\ &- \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\tilde{\Psi}_{V1}^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}W_{V1}(t)\right\} - \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\varepsilon_{HJI1}^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}\right\} \\ &\leq -\frac{\alpha_h}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1 + \left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}(t)\right\|^2 - \frac{\alpha_h}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1 + \left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}(t)\right\|^2 \\ &+ \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\frac{\hat{\Psi}_{V1}\tilde{\Phi}^T}{1 + \hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}\right\} - \alpha_h\frac{\left\|\tilde{\Phi}\right\|^2}{1 + \left\|\hat{\Psi}_{V1}\right\|^2} + \alpha_h\frac{\left\|\tilde{\Phi}\right\|^2}{1 + \left\|\hat{\Psi}_{V1}\right\|^2} \end{aligned}$$

$$
\begin{aligned}
&- \frac{\alpha_h}{4} \frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \left\|\tilde{W}_{V1}(t)\right\|^2 - \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t) \frac{\hat{\Psi}_{V1}\tilde{\Psi}_{V1}^T}{1+\hat{\Psi}_{V1}^T\hat{\Psi}_{V1}} W_{V1}(t)\right\} \\
&- \alpha_h \frac{\left\|W_V^T(t)\tilde{\Psi}_{V1}\right\|}{1+\left\|\hat{\Psi}_{V1}\right\|} + \alpha_h \frac{\left\|W_V^T(t)\tilde{\Psi}_{V1}\right\|}{1+\left\|\hat{\Psi}_{V1}\right\|} - \frac{\alpha_h}{4} \frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \left\|\tilde{W}_{V1}(t)\right\|^2 \\
&- \alpha_h \operatorname{tr}\left\{\tilde{W}_{V1}^T(t) \frac{\hat{\Psi}_{V1}\varepsilon_{HJI1}^T}{1+\hat{\Psi}_{V1}^T\hat{\Psi}_{V1}}\right\} - \alpha_h \frac{\|\varepsilon_{HJI1}\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} + \alpha_h \frac{\|\varepsilon_{HJI1}\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}
\end{aligned} \tag{3.39}
$$

Combining terms in (3.39),

$$
\begin{aligned}
\dot{L}_{V1}(t) \leq &- \frac{\alpha_h}{4} \frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \left\|\tilde{W}_{V1}(t)\right\|^2 - \frac{\alpha_h}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \left\|\frac{\tilde{W}_{V1}(t)\hat{\Psi}_{V1}}{2} - \tilde{\Phi}\right\|^2 \\
&- \frac{\alpha_h}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \left\|\frac{\tilde{W}_{V1}(t)\hat{\Psi}_{V1}}{2} - W_V^T(t)\tilde{\Psi}_{V1}\right\|^2 - \frac{\alpha_h}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \left\|\frac{\tilde{W}_{V1}(t)\hat{\Psi}_{V1}}{2} - \varepsilon_{HJI1}\right\|^2 \\
&+ \alpha_h \frac{\left\|\tilde{\Phi}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} + \alpha_h \frac{\left\|\tilde{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} + \underbrace{\alpha_h \frac{\|\varepsilon_{HJI1}\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}}_{\varepsilon_{VHJI}}
\end{aligned} \tag{3.40}
$$

Drop the negative terms in the right side of the inequality yields,

$$
\dot{L}_{V1}(t) \leq - \frac{\alpha_h}{4} \frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \left\|\tilde{W}_{V1}(t)\right\|^2 + \alpha_h \frac{\left\|\tilde{\Phi}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} + \alpha_h \frac{\left\|\tilde{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} + \varepsilon_{VHJI} \tag{3.41}
$$

Assume that the coupling function $\phi(x_1, m_1, m_2)$, and the function $\Psi_V(x_1, m_1, m_2)$ are Lipschitz and the Lipschitz constant are $L_\Phi$, $L_{\Psi V}$. (3.41) can be simplified as

$$
\dot{L}_{V1}(t) \leq - \frac{\alpha_h}{4} \frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \left\|\tilde{W}_{V1}(t)\right\|^2 + \alpha_h \frac{\left[L_\Phi + L_{\Psi V}\|W_V\|^2\right] \|\tilde{m}_1\tilde{m}_2\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} + \varepsilon_{VHJI}
$$

$$\leq -\frac{\alpha_h}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}(t)\right\|^2 + B_V(t) \tag{3.42}$$

According to the Lyapunov stability analysis, the critic NN weight estimation error will be Uniformly Ultimately Bounded (UUB) with the bound given as

$$\|\tilde{W}_{V1}\| \leq \sqrt{\frac{4(1+\|\hat{\Psi}_{V1}\|^2)}{\alpha_h\|\hat{\Psi}_{V1}\|^2}B_V(t)} \equiv b_{WV}(t) \tag{3.43}$$

□

We also derive the bound of estimated optimal cost function as follows:

Let $\tilde{V}_1 = V_1 - \hat{V}_1$, and substitute (3.15), one obtains,

$$\begin{aligned}\tilde{V}_1(t) &= W_{V1}^T(t)\phi_{V1} - \hat{W}_{V1}^T(t)\hat{\phi}_{V1} + \varepsilon_{HJI1} \\ &= W_{V1}^T(t)(\tilde{\phi}_{V1} + \hat{\phi}_{V1}) - \hat{W}_{V1}(t)^T\hat{\phi}_{V1} + \varepsilon_{HJI1} \\ &= \tilde{W}_{V1}^T(t)\hat{\phi}_{V1} + W_{V1}^T(t)\tilde{\phi}_{V1} + \varepsilon_{HJI1}\end{aligned} \tag{3.44}$$

Assume the critic NN activation function is Lipschitz, and the Lipschitz constant is denoted as $L_{\phi v}$. The value function estimation error can be represented as:

$$\begin{aligned}\|\tilde{V}_1(t)\| &= \|\tilde{W}_{V1}^T(t)\hat{\phi}_{V1} + W_{V1}^T(t)\tilde{\phi}_{V1} + \varepsilon_{HJI1}\| \\ &\leq \|\tilde{W}_{V1}(t)\|\|\hat{\phi}_{V1}\| + L_{\phi v}\|W_{V1}(t)\|\|\tilde{m_1}\tilde{m_2}\| + \|\varepsilon_{HJI1}\| \\ &\leq b_{WV}(t)\|\hat{\phi}_{V1}\| + L_{\phi v}\|W_{V1}(t)\|\|\tilde{m_1}\tilde{m_2}\| + \|\varepsilon_{HJI1}\| \equiv b_{V1}(t)\end{aligned} \tag{3.45}$$

**Theorem 7.** *(Mass NN's convergence)*: Let $\hat{W}_{m,j,1}(t)$ be updated as (3.25) shows and assume the learning rate $\alpha_{m,j} > 0$, then the mass NN's weights estimation error $\tilde{W}_{m,j,1}$, the mass function approximation error, i.e., $\tilde{m}_{j,1} = m_1 - \hat{m}_{j,1}$ are uniformly

ultimately bounded (UUB). The corresponding bound $b_{W_{m,j}}$, $b_{m,j}$ is trivial when the reconstruction error is sufficiently small [91]. Moreover, $\tilde{W}_{m,j,1}$ and $\tilde{m}_{j,1}$ are also asymptotically stable if the neuron network structure is selected perfectly.

*Proof.* Consider the following Lyapunov function

$$L_{m1}(t) = \frac{1}{2}\operatorname{tr}\left\{\tilde{W}_{m1}^T(t)\tilde{W}_{m1}(t)\right\} \tag{3.46}$$

Take the first derivative on the Lyapunov function candidate, one obtains:

$$\dot{L}_{m1}(t) = \frac{1}{2}\operatorname{tr}\left\{\tilde{W}_{m1}^T(t)\dot{\tilde{W}}_{m1}(t)\right\} + \frac{1}{2}\operatorname{tr}\left\{\dot{\tilde{W}}_{m1}^T(t)\tilde{W}_{m1}(t)\right\} = \operatorname{tr}\left\{\tilde{W}_{m1}^T(t)\dot{\tilde{W}}_{m1}(t)\right\} \tag{3.47}$$

Since the correct estimated optimal cost function leads to the FPK equation equals zero, we have

$$W_{m1}^T(t)\Psi_{m1}\left(x_1, V_1\right) + \varepsilon_{FPK1} = 0 \tag{3.48}$$

Combine (3.48) and (3.18), we have

$$-W_{m1}^T(t)\Psi_{m1}\left(x_1, V_1\right) - \varepsilon_{FPK1} - \hat{W}_{m1}^T(t)\hat{\Psi}_{m1}\left(x_1, \hat{V}_1\right) = e_{FPK1} \tag{3.49}$$

Let $\tilde{W}_{m1}(t) = W_{m1}(t) - \hat{W}_{m1}(t)$, and $\tilde{\Psi}_{m1}(x_1, V_1, \hat{V}_1) = \Psi_{m1}(x_1, V_1) - \hat{\Psi}_{m1}(x_1, \hat{V}_1)$. After manipulating terms in (3.49), we obtain

$$\begin{aligned} &- W_{m1}^T(t)\left(\hat{\Psi}_{m1}\left(x_1, \hat{V}_1\right) + \tilde{\Psi}_{m1}(x_1, V_1, \hat{V}_1)\right) - \varepsilon_{FPK1} + \hat{W}_{m1}^T(t)\hat{\Psi}_{m1}\left(x_1, \hat{V}_1\right) = e_{FPK1} \\ &- \tilde{W}_{m1}^T\hat{\Psi}_{m1}\left(x_1, \hat{V}_1\right) - W_{m1}^T\tilde{\Psi}_{m1}(x_1, V_1, \hat{V}_1) - \varepsilon_{FPK1} = e_{FPK1} \end{aligned} \tag{3.50}$$

where $\varepsilon_{FPK1}$ is the error resulted from the reconstruction error.

Let's further simplify the notations as: $\hat{\Psi}_{m1}\left(x_1, \hat{V}_1\right) \to \hat{\Psi}_{m1}$, $\tilde{\Psi}_{m1}(x_1, V_1, \hat{V}_1) \to \tilde{\Psi}_{m1}$, $\Psi_{m1}\left(x_1, V_1\right) \to \Psi_{m1}$

Substitute (3.50) into (3.47),

$$\dot{L}_{m1}(t) = \alpha_m \operatorname{tr}\left\{ \tilde{W}_{m1}^T(t) \frac{\hat{\Psi}_{m1}\left[-\tilde{W}_{m1}^T\hat{\Psi}_{m1} - W_{m1}^T\tilde{\Psi}_{m1} - \varepsilon_{FPK1}\right]^T}{1+\hat{\Psi}_{m1}^T\hat{\Psi}_{m1}} \right\}$$
$$= -\alpha_m \operatorname{tr}\left\{ \tilde{W}_{m1}^T(t) \frac{\hat{\Psi}_{m1}\hat{\Psi}_{m1}^T}{1+\hat{\Psi}_{m1}^T\hat{\Psi}_{m1}} \tilde{W}_{m1}(t) \right\} - \alpha_m \operatorname{tr}\left\{ \tilde{W}_{m1}^T(t) \frac{\hat{\Psi}_{m1}\tilde{\Psi}_{m1}^T}{1+\hat{\Psi}_{m1}^T\hat{\Psi}_{m1}} W_{m1}(t) \right\}$$
$$- \alpha_m \operatorname{tr}\left\{ \tilde{W}_{m1}^T(t) \frac{\hat{\Psi}_{m1}\varepsilon_{FPK1}^T}{1+\hat{\Psi}_{m1}^T\hat{\Psi}_{m1}} \right\}$$

Apply Cauchy-Schwarz inequality on the above equation,

$$\dot{L}_{m1}(t) \tag{3.51}$$
$$= -\alpha_m \operatorname{tr}\left\{ \tilde{W}_{m1}^T(t) \frac{\hat{\Psi}_{m1}\hat{\Psi}_{m1}^T}{1+\hat{\Psi}_{m1}^T\hat{\Psi}_{m1}} \tilde{W}_{m1}(t) \right\} - \alpha_m \operatorname{tr}\left\{ \tilde{W}_{m1}^T(t) \frac{\hat{\Psi}_{m1}\tilde{\Psi}_{m1}^T}{1+\hat{\Psi}_{m1}^T\hat{\Psi}_{m1}} W_{m1}(t) \right\}$$
$$- \alpha_m \operatorname{tr}\left\{ \tilde{W}_{m1}^T(t) \frac{\hat{\Psi}_{m1}\varepsilon_{FPK1}^T}{1+\hat{\Psi}_{m1}^T\hat{\Psi}_{m1}} \right\}$$
$$\leq -\frac{\alpha_m}{2} \frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \left\|\tilde{W}_{m1}(t)\right\|^2 - \frac{\alpha_m}{4} \frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \left\|\tilde{W}_{m1}(t)\right\|^2$$
$$- \alpha_m \operatorname{tr}\left\{ \tilde{W}_{m1}^T(t) \frac{\hat{\Psi}_{m1}\tilde{\Psi}_{m1}^T}{1+\hat{\Psi}_{m1}^T\hat{\Psi}_{m1}} W_{m1}(t) \right\} + \alpha_m \frac{\left\|W_{m1}^T(t)\tilde{\Psi}_{m1}\right\|}{1+\left\|\hat{\Psi}_{m1}\right\|}$$
$$- \frac{\alpha_m}{4} \frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \left\|\tilde{W}_{m1}(t)\right\|^2 - \alpha_m \operatorname{tr}\left\{ \tilde{W}_{m1}^T(t) \frac{\hat{\Psi}_{m1}\varepsilon_{FPK1}^T}{1+\hat{\Psi}_{m1}^T\hat{\Psi}_{m1}} \right\}$$
$$- \alpha_m \frac{\left\|\varepsilon_{FPK1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} + \alpha_m \frac{\left\|\varepsilon_{FPK1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \tag{3.52}$$

Combining terms in (3.51),

$$\dot{L}_{m1}(t) \leq -\frac{\alpha_m}{2} \frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \left\|\tilde{W}_{m1}(t)\right\|^2$$
$$- \frac{\alpha_m}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \left\| \frac{\tilde{W}_{m1}(t)\hat{\Psi}_{m1}}{2} - W_{m1}^T(t)\tilde{\Psi}_{m1} \right\|^2$$

$$-\frac{\alpha_m}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\frac{\tilde{W}_{m1}(t)\hat{\Psi}_{m1}}{2}-\varepsilon_{FPK1}\right\|^2+\alpha_m\frac{\left\|\tilde{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\underbrace{\alpha_m\frac{\left\|\varepsilon_{FPK1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}}_{\varepsilon_{NFPK1}} \tag{3.53}$$

Drop the negative terms in the right side of the inequality yields,

$$\dot{L}_{m1}(t)\leq-\frac{\alpha_m}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}(t)\right\|^2+\alpha_m\frac{\left\|\tilde{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\varepsilon_{NFPK1} \tag{3.54}$$

Assume that the function $\Psi_{m1}(x_1,V_1)$ are Lipschitz and the Lipschitz constant is $L_{\Psi m}$. (3.54) can be simplified as

$$\begin{aligned}\dot{L}_{m1}(t)&\leq-\frac{\alpha_m}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}(t)\right\|^2+\alpha_m\frac{L_{\Psi m}\|W_{m1}\|^2\|\tilde{V}_1\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\varepsilon_{NFPK1}\\&\leq-\frac{\alpha_m}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}(t)\right\|^2+B_{m1}(t)\end{aligned} \tag{3.55}$$

According to the Lyapunov stability analysis, the mass NN weight estimation error will be Uniformly Ultimately Bounded (UUB) with the bound given as

$$\|\tilde{W}_{m1}\|\leq\sqrt{\frac{2(1+\|\hat{\Psi}_{m1}\|^2)}{\alpha_m\|\hat{\Psi}_{m1}\|^2}B_{m1}(t)}\equiv b_{Wm}(t) \tag{3.56}$$

□

We also derive the bound of estimated mass function as follows:

Let $\tilde{m}_1=m_1-\hat{m}_1$, and substitute (3.15), one obtains,

$$\tilde{m}_1(t)=W_{m1}^T(t)\phi_{m1}-\hat{W}^T\phi_{m1}+\varepsilon_{FPK1}$$

$$= \tilde{W}_{m1}^T(t)\phi_{m1} + \varepsilon_{FPK1} \tag{3.57}$$

The PDF estimation error can be represented as:

$$\begin{aligned} \|\tilde{m}_1(t)\| &= \|\tilde{W}_{m1}^T(t)\phi_{m1} + \varepsilon_{FPK1}\| \\ &\leq \|\tilde{W}_{m1}(t)\|\|\phi_{m1}\| + \|\varepsilon_{FPK1}\| \\ &\leq b_{Wm}(t)\|\hat{\phi}_{m1}\| + \|\varepsilon_{FPK1}\| \equiv b_{m1}(t) \end{aligned} \tag{3.58}$$

**Theorem 8.** *(Actor NN's convergence)*: Let $\hat{W}_{u,j,1}(t)$ be updated as (3.24) shows and assume the learning rate $\alpha_{u,j} > 0$, then the actor NN's weights estimation error $\tilde{W}_{u,j,1}$, and the optimal control approximation error, i.e., $\tilde{u}_{j,1} = u_{j,1} - \hat{u}_{j,1}$ are said to be uniformly ultimately bounded (UUB). The corresponding bound $b_{W_{u,j}}$, $b_{u,j}$ are trivial when the reconstruction error is sufficiently small. Moreover, $\tilde{W}_{u,j,1}$ and $\tilde{u}_{j,1}$ are also asymptotically stable if the neuron network structure is selected perfectly.

*Proof.* Consider the following Lyapunov function

$$L_{u1}(t) = \frac{1}{2}\operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\tilde{W}_{u1}(t)\right\} \tag{3.59}$$

Take the first derivative on the Lyapunov function candidate, one obtains:

$$\dot{L}_{u1}(t) = \frac{1}{2}\operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\dot{\tilde{W}}_{u1}(t)\right\} + \frac{1}{2}\operatorname{tr}\left\{\dot{\tilde{W}}_{u1}^T(t)\tilde{W}_{u1}(t)\right\} = \operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\dot{\tilde{W}}_{u1}(t)\right\} \tag{3.60}$$

Since the correct estimated optimal cost function leads to the optimal control equation equals zero, we have

$$W_{u1}^T(t)\phi_{u1}\left(x_1, m_1, m_2\right) + \frac{1}{2}R_{g1}^{-1}g_1(x_1)\frac{\partial \hat{V}_1(x_1, \hat{m}_1, \hat{m}_2)}{\partial x_1} + \varepsilon_{u1} = 0 \tag{3.61}$$

Let $\tilde{W}_{u1}(t) = W_{u1}(t) - \hat{W}_{u1}(t)$, and $\tilde{\phi}_{u1}(x_1, m_1, m_2, \hat{m}_1, \hat{m}_2) = \phi_{u1}(x_1, m_1, m_2) - \hat{\phi}_{u1}(x_1, \hat{m}_1, \hat{m}_2)$. Similar to the critic and actor NNs, after manipulating terms, we

obtain

$$\begin{aligned} e_{u1} = &- \tilde{W}_{u1}^T \hat{\phi}_{u1}\left(x_1, \hat{m}_1, \hat{m}_2\right) - W_{u1}^T \tilde{\phi}_{u1}(x_1, m_1, m_2, \hat{m}_1, \hat{m}_2) \\ &- \frac{1}{2} R_{g1}^{-1} g_1(x_1) \frac{\partial \hat{V}_1(x_1, \hat{m}_1, \hat{m}_2)}{\partial x_1} - \varepsilon_{u1} \end{aligned} \tag{3.62}$$

where $\varepsilon_{u1}$ is the error resulted from the reconstruction error.

Let's further simplify the notations as:

$\hat{\phi}_{u1}\left(x_1, \hat{m}_1, \hat{m}_2\right) \rightarrow \hat{\phi}_{u1}$, $\tilde{\phi}_{u1}(x_1, m_1, m_2, \hat{m}_1, \hat{m}_2) \rightarrow \tilde{\phi}_{u1}$, $\phi_{u1}\left(x_1, m_1, m_2\right) \rightarrow \phi_{u1}$

Substitute (3.62) into (3.60),

$$\begin{aligned} \dot{L}_{u1}(t) &= \alpha_u \operatorname{tr}\left\{ \tilde{W}_{u1}^T(t) \frac{\hat{\phi}_{u1}\left[-\tilde{W}_{u1}^T \hat{\phi}_{u1} - W_{u1}^T \tilde{\phi}_{u1} - \frac{1}{2} R_{g1}^{-1} g_1(x_1) \frac{\partial \hat{V}_1}{\partial x_1} - \varepsilon_{u1}\right]^T}{1 + \hat{\phi}_{u1}^T \hat{\phi}_{u1}} \right\} \\ &= -\alpha_u \operatorname{tr}\left\{ \tilde{W}_{u1}^T(t) \frac{\hat{\phi}_{u1} \hat{\phi}_{u1}^T}{1 + \hat{\phi}_{u1}^T \hat{\phi}_{u1}} \tilde{W}_{u1}(t) \right\} - \alpha_u \operatorname{tr}\left\{ \tilde{W}_{u1}^T(t) \frac{\hat{\phi}_{u1} \tilde{\phi}_{u1}^T}{1 + \hat{\phi}_{u1}^T \hat{\phi}_{u1}} W_{u1}(t) \right\} \\ &- \alpha_u \operatorname{tr}\left\{ \tilde{W}_{u1}^T(t) \frac{\hat{\phi}_{u1}\left[\frac{1}{2} R_{g1}^{-1} g_1(x_1) \frac{\partial \hat{V}_1}{\partial x_1}\right]^T}{1 + \hat{\phi}_{u1}^T \hat{\phi}_{u1}} \right\} - \alpha_u \operatorname{tr}\left\{ \tilde{W}_{u1}^T(t) \frac{\hat{\phi}_{u1} \varepsilon_{u1}^T}{1 + \hat{\phi}_{u1}^T \hat{\phi}_{u1}} \right\} \end{aligned} \tag{3.63}$$

Apply Cauchy-Schwarz inequality on (3.63),

$$\dot{L}_{u1}(t) \tag{3.64}$$

$$\begin{aligned} &= - \alpha_u \operatorname{tr}\left\{ \tilde{W}_{u1}^T(t) \frac{\hat{\phi}_{u1} \hat{\phi}_{u1}^T}{1 + \hat{\phi}_{u1}^T \hat{\phi}_{u1}} \tilde{W}_{u1}(t) \right\} - \alpha_u \operatorname{tr}\left\{ \tilde{W}_{u1}^T(t) \frac{\hat{\phi}_{u1} \tilde{\phi}_{u1}^T}{1 + \hat{\phi}_{u1}^T \hat{\phi}_{u1}} W_{u1}(t) \right\} \\ &- \alpha_u \operatorname{tr}\left\{ \tilde{W}_{u1}^T(t) \frac{\hat{\phi}_{u1}\left[\frac{1}{2} R_{g1}^{-1} g_1(x_1) \frac{\partial \hat{V}_1}{\partial x_1}\right]^T}{1 + \hat{\phi}_{u1}^T \hat{\phi}_{u1}} \right\} - \alpha_u \operatorname{tr}\left\{ \tilde{W}_{u1}^T(t) \frac{\hat{\phi}_{u1} \varepsilon_{u1}^T}{1 + \hat{\phi}_{u1}^T \hat{\phi}_{u1}} \right\} \end{aligned} \tag{3.65}$$

$$
\begin{aligned}
&\leq -\frac{\alpha_u}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}(t)\right\|^2 - \frac{\alpha_u}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}(t)\right\|^2 \\
&\quad - \alpha_u \operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\frac{\hat{\phi}_{u1}\left[\frac{1}{2}R_{g1}^{-1}g_1(x_1)\frac{\partial \hat{V}_1}{\partial x_1}\right]^T}{1+\hat{\phi}_{u1}^T\hat{\phi}_{u1}}\right\} - \alpha_u\frac{\left\|\frac{1}{2}R_{g1}^{-1}g_1(x_1)\frac{\partial \hat{V}_1}{\partial x_1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} \\
&\quad + \alpha_u\frac{\left\|\frac{1}{2}R_{g1}^{-1}g_1(x_1)\frac{\partial \hat{V}_1}{\partial x_1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} - \frac{\alpha_u}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}(t)\right\|^2 \\
&\quad - \alpha_u \operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\frac{\hat{\phi}_{u1}\tilde{\phi}_{u1}^T}{1+\hat{\phi}_{u1}^T\hat{\phi}_{u1}}W_{u1}(t)\right\} - \alpha_u\frac{\left\|W_{u1}^T(t)\tilde{\phi}_{u1}\right\|}{1+\left\|\hat{\phi}_{u1}\right\|} + \alpha_u\frac{\left\|W_{u1}^T(t)\tilde{\phi}_{u1}\right\|}{1+\left\|\hat{\phi}_{u1}\right\|} \\
&\quad - \frac{\alpha_u}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}(t)\right\|^2 - \alpha_u \operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\frac{\hat{\phi}_{u1}\varepsilon_{u1}^T}{1+\hat{\phi}_{u1}^T\hat{\phi}_{u1}}\right\} \\
&\quad - \alpha_u\frac{\left\|\varepsilon_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} + \alpha_u\frac{\left\|\varepsilon_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}
\end{aligned}
\tag{3.66}
$$

Combining terms in (3.64),

$$
\begin{aligned}
\dot{L}_{u1}(t) \leq &-\frac{\alpha_u}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}(t)\right\|^2 - \frac{\alpha_u}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\frac{\tilde{W}_{u1}(t)\hat{\phi}_{u1}}{2} - W_{u1}^T(t)\tilde{\phi}_{u1}\right\|^2 \\
&- \frac{\alpha_u}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\frac{\tilde{W}_{u1}(t)\hat{\phi}_{u1}}{2} - \frac{1}{2}R_{g1}^{-1}g_1(x_1)\frac{\partial \hat{V}_1}{\partial x_1}\right\|^2 \\
&- \frac{\alpha_u}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\frac{\tilde{W}_{u1}(t)\hat{\phi}_{u1}}{2} - \varepsilon_{u1}\right\|^2 + \frac{\alpha_u}{4}\frac{\left\|R_{g1}^{-1}g_1(x_1)\frac{\partial \hat{V}_1}{\partial x_1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} + \underbrace{\alpha_u\frac{\left\|\varepsilon_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}}_{\varepsilon_{Nu1}}
\end{aligned}
\tag{3.67}
$$

Drop the negative terms in the right side of the inequality yields,

$$
\dot{L}_{u1}(t) \leq -\frac{\alpha_u}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}(t)\right\|^2 + \frac{\alpha_u}{4}\frac{\left\|R_{g1}^{-1}g_1(x_1)\frac{\partial \hat{V}_1}{\partial x_1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} + \varepsilon_{Nu1}
$$

$$\leq -\frac{\alpha_u}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}(t)\right\|^2 + \alpha_u\frac{\|R_{g1}^{-1}g_1(x_1)\|^2\|\tilde{V}_1\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} + \varepsilon_{Nu1}$$

$$\leq -\frac{\alpha_u}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}(t)\right\|^2 + B_{u1}(t) \tag{3.68}$$

According to the Lyapunov stability analysis, the actor NN weight estimation error will be Uniformly Ultimately Bounded (UUB) with the bound given as

$$\|\tilde{W}_{u1}\| \leq \sqrt{\frac{4(1+\|\hat{\phi}_{u1}\|^2)}{\alpha_u\|\hat{\phi}_{u1}\|^2}B_{u1}(t)} \equiv b_{Wm}(t) \tag{3.69}$$

□

We also derive the bound of estimated optimal control function as follows:

Similarly, let $\tilde{u}_1 = m_1 - \hat{u}_1$, and substitute (3.15), (3.16), one obtains,

$$\tilde{u}_1(t) = \tilde{W}_{u1}^T(t)\phi_{u1} + W_{u1}^T(t)\tilde{\phi}_{u1} + \varepsilon_{u1} \tag{3.70}$$

The optimal control estimation error can be represented as:

$$\begin{aligned}\|\tilde{u}_1(t)\| &= \|\tilde{W}_{u1}^T(t)\phi_{u1} + W_{u1}^T(t)\tilde{\phi}_{u1} + \varepsilon_{u1}\| \\ &\leq \|\tilde{W}_{u1}(t)\|\|\hat{\phi}_{u1}\| + L_{\phi u}\|W_{u1}\|\|\tilde{m_1}\tilde{m_2}\| + \|\varepsilon_{u1}\| \\ &\leq b_{Wu}(t)\|\hat{\phi}_{u1}\| + L_{\phi u}\|W_{u1}\|\|\tilde{m_1}\tilde{m_2}\| + \|\varepsilon_{u1}\| \equiv b_{u1}(t)\end{aligned} \tag{3.71}$$

where $L_{\phi u}$ is the Lipschitz constant of the actor NN's activation function.

**Theorem 9.** *(Opponent NN's convergence)*: Let $\hat{W}_{m,j,2}(t)$ be updated as (3.26) shows and assume the learning rate $\alpha_{m2,j} > 0$, then the opponent NN's weights estimation error, $\tilde{W}_{m,j,2}$, and the estimation error of the coupling cost $\tilde{x}_{j,2} = \Phi_{Q2}(x_{j,1}, \mathbb{E}\{m_2^*\}) - \hat{x}_{j,2}$ are said to be uniformly ultimately bounded (UUB). The corresponding bound

$b_{W_{m2,j}}$, $b_{m2,j}$ are trivial when the reconstruction error is sufficiently small. Moreover, $\tilde{W}_{m,j,2}$ and $\tilde{x}_{j,2}$ are also asymptotically stable if the neuron network structure is selected perfectly.

*Proof.* Similar to above. □

**Remark 2.** While the actor, critic, mass, and opponent NNs are learning the optimal control, evaluation function, and PDF of all agents' tracking error respectively, the bound will reduce significantly and only depends on the NNs' reconstruction error, which can be ignored when the perfect neuron numbers are selected [91, 2].

Eventually, we analyze the closed-loop stability. Before that, similar to , a necessary lemma is introduced.

**Lemma 2.** Given the stochastic system dynamic equations in (3.1), (3.2), there exist optimal control policies for pursuers $u_{j,1}^*$ which satisfy

$$x_{j,1}^T \left[ f_1(x_{j,1}) + g_1(x_{j,1}) u_{j,1}^* + \frac{D_1^2}{2} \frac{dw_{j,1}}{dt} \right] \leq -\gamma_1 \left\| x_j \right\|^2 \tag{3.72}$$

where $\gamma_1 > 0$.

**Theorem 10.** *(Closed-loop Stability)* Let the pursuers' critic, actor, mass, and opponent NNs' synaptic weights being updates as (3.23), (3.24), (3.25), (3.26), and assume the learning rates $\alpha_{h,j}, \alpha_{m,j}, \alpha_{u,j}, \alpha_{m2,j}$, then $\tilde{W}_{V,j,1}$, $\tilde{W}_{m,j,1}$, $\tilde{W}_{u,j,1}, \tilde{W}_{m,j,2}$, $\tilde{x}_{j,1}, \tilde{m}_{j,1}, \tilde{u}_{j,1}$ are all UUB. Moreover, if all neuron networks' structures are selected perfectly, $\tilde{W}_{V,j,1}$, $\tilde{W}_{m,j,1}$, $\tilde{W}_{u,j,1}, \tilde{W}_{m,j,2}, \tilde{x}_{j,1}, \tilde{m}_{j,1}, \tilde{u}_{j,1}$ are all asymptotically stable. Meanwhile, if the evaders follow the ACMO algorithm, their neural networks and system states are asymptotically stable as well.

*Proof.* Consider the Lyapunov function candidate as:

$$L_{sysm}(t) = \frac{\beta_1}{2} \operatorname{tr} \left\{ x_1^T(t) x_1(t) \right\} + \frac{\beta_2}{2} \operatorname{tr} \left\{ \tilde{W}_{V1}^T(t) \tilde{W}_{V1}(t) \right\}$$

$$+\frac{\beta_3}{2}\operatorname{tr}\left\{\tilde{W}_{m1}^T(t)\tilde{W}_{m1}(t)\right\}+\frac{\beta_4}{2}\operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\tilde{W}_{u1}(t)\right\}$$
$$+\frac{\beta_5}{2}\operatorname{tr}\left\{x_2^T(t)x_2(t)\right\}+\frac{\beta_6}{2}\operatorname{tr}\left\{\tilde{W}_{V2}^T(t)\tilde{W}_{V2}(t)\right\}+\frac{\beta_7}{2}\operatorname{tr}\left\{\tilde{W}_{m2}^T(t)\tilde{W}_{m2}(t)\right\}$$
$$+\frac{\beta_8}{2}\operatorname{tr}\left\{\tilde{W}_{u2}^T(t)\tilde{W}_{u2}(t)\right\} \tag{3.73}$$

According to the Lyapunov stability method, taking the first derivative of the selected Lyapunov function candidate

$$\begin{aligned}\dot{L}_{sysm}(t)=&\frac{\beta_1}{2}\operatorname{tr}\left\{x_1^T(t)\dot{x}_1(t)\right\}+\frac{\beta_1}{2}\operatorname{tr}\left\{\dot{x}_1^T(t)x_1(t)\right\}+\frac{\beta_2}{2}\operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\dot{\tilde{W}}_{V1}(t)\right\}\\&+\frac{\beta_2}{2}\operatorname{tr}\left\{\dot{\tilde{W}}_{V1}^T(t)\tilde{W}_{V1}(t)\right\}+\frac{\beta_3}{2}\operatorname{tr}\left\{\tilde{W}_{m1}^T(t)\dot{\tilde{W}}_{m1}(t)\right\}\\&+\frac{\beta_3}{2}\operatorname{tr}\left\{\dot{\tilde{W}}_{m1}^T(t)\tilde{W}_{m1}(t)\right\}+\frac{\beta_4}{2}\operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\dot{\tilde{W}}_{u1}(t)\right\}+\frac{\beta_4}{2}\operatorname{tr}\left\{\dot{\tilde{W}}_{u1}^T(t)\tilde{W}_{u1}(t)\right\}\\&+\frac{\beta_5}{2}\operatorname{tr}\left\{x_2^T(t)\dot{x}_2(t)\right\}+\frac{\beta_5}{2}\operatorname{tr}\left\{\dot{x}_2^T(t)x_2(t)\right\}+\frac{\beta_6}{2}\operatorname{tr}\left\{\tilde{W}_{V2}^T(t)\dot{\tilde{W}}_{V2}(t)\right\}\\&+\frac{\beta_6}{2}\operatorname{tr}\left\{\dot{\tilde{W}}_{V2}^T(t)\tilde{W}_{V2}(t)\right\}+\frac{\beta_7}{2}\operatorname{tr}\left\{\tilde{W}_{m2}^T(t)\dot{\tilde{W}}_{m2}(t)\right\}\\&+\frac{\beta_7}{2}\operatorname{tr}\left\{\dot{\tilde{W}}_{m2}^T(t)\tilde{W}_{m2}(t)\right\}+\frac{\beta_8}{2}\operatorname{tr}\left\{\tilde{W}_{u2}^T(t)\dot{\tilde{W}}_{u2}(t)\right\}+\frac{\beta_8}{2}\operatorname{tr}\left\{\dot{\tilde{W}}_{u2}^T(t)\tilde{W}_{u2}(t)\right\}\\=&\beta_1\operatorname{tr}\left\{x_1^T(t)\dot{x}_1(t)\right\}+\beta_2\operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\dot{\tilde{W}}_{V1}(t)\right\}+\beta_3\operatorname{tr}\left\{\tilde{W}_{m1}^T(t)\dot{\tilde{W}}_{m1}(t)\right\}\\&+\beta_4\operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\dot{\tilde{W}}_{u1}(t)\right\}+\beta_5\operatorname{tr}\left\{\tilde{x}_2^T(t)x_2(t)\right\}+\beta_6\operatorname{tr}\left\{\dot{\tilde{W}}_{V2}^T(t)\tilde{W}_{V2}(t)\right\}\\&+\beta_7\operatorname{tr}\left\{\tilde{W}_{m2}^T(t)\dot{\tilde{W}}_{m2}(t)\right\}+\beta_8\operatorname{tr}\left\{\tilde{W}_{u2}^T(t)\dot{\tilde{W}}_{u2}(t)\right\}\end{aligned} \tag{3.74}$$

Recall to the Lemma 2, Theorems 6-8, and equations (3.42), (3.55), (3.68), (3.74) can be represented as:

$$\begin{aligned}&\dot{L}_{sysm}(t)\\=&\beta_1\operatorname{tr}\left\{x_1^T(t)\dot{x}_1(t)\right\}+\beta_2\operatorname{tr}\left\{\tilde{W}_{V1}^T(t)\dot{\tilde{W}}_{V1}(t)\right\}+\beta_3\operatorname{tr}\left\{\tilde{W}_{m1}^T(t)\dot{\tilde{W}}_{m1}(t)\right\}\\&+\beta_4\operatorname{tr}\left\{\tilde{W}_{u1}^T(t)\dot{\tilde{W}}_{u1}(t)\right\}+\beta_5\operatorname{tr}\left\{\tilde{x}_2^T(t)x_2(t)\right\}+\beta_6\operatorname{tr}\left\{\dot{\tilde{W}}_{V2}^T(t)\tilde{W}_{V2}(t)\right\}\\&+\beta_7\operatorname{tr}\left\{\tilde{W}_{m2}^T(t)\dot{\tilde{W}}_{m2}(t)\right\}+\beta_8\operatorname{tr}\left\{\tilde{W}_{u2}^T(t)\dot{\tilde{W}}_{u2}(t)\right\}\end{aligned}$$

$$
\begin{aligned}
\leq & \beta_1 \operatorname{tr}\left\{x_1^T\left[f_1(x_1)+g_1(x_1)u_1^*+\sigma_1\frac{dw_1}{dt}\right]\right\}-\beta_1\operatorname{tr}\left\{x_1^T g_1(x_1)\tilde{u}_1\right\} \\
& -\frac{2\beta_1}{\gamma_1}\left\|g_1(x_1)\tilde{u}_1\right\|^2+\frac{2\beta_1}{\gamma_1}\left\|g_1(x_1)\tilde{u}_1\right\|^2-\frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}\right\|^2 \\
& +\alpha_h\frac{\beta_2\left[L_\Phi+L_{\Psi V1}\left\|W_{V1}\right\|^2\right]\left\|\tilde{m}_1\tilde{m}_2\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}+\beta_2\varepsilon_{VHJI1} \\
& -\frac{\alpha_m\beta_3}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}\right\|^2+\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\left\|W_{m1}\right\|^2\left\|\tilde{V}_1\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\beta_3\varepsilon_{NFPK1} \\
& -\frac{\alpha_u\beta_4}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}\right\|^2+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2\left\|\tilde{V}_1\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}+\beta_4\varepsilon_{Nu1} \\
& +\beta_5\operatorname{tr}\left\{x_2^T\left[f_2(x_2)+g_2(x_2)u_2^*+\sigma_2\frac{dw_2}{dt}\right]\right\}-\beta_5\operatorname{tr}\left\{x_2^T g_2(x_2)\tilde{u}_2\right\} \\
& -\frac{2\beta_5}{\gamma_2}\left\|g_2(x_2)\tilde{u}_2\right\|^2+\frac{2\beta_5}{\gamma_2}\left\|g_2(x_2)\tilde{u}_2\right\|^2-\frac{\alpha_h\beta_6}{4}\frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\left\|\tilde{W}_{V2}\right\|^2 \\
& +\alpha_h\frac{\beta_6\left[L_\Phi+L_{\Psi V2}\left\|W_{V2}\right\|^2\right]\left\|\tilde{m}_1\tilde{m}_2\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}+\beta_6\varepsilon_{VHJI2} \\
& -\frac{\alpha_m\beta_7}{2}\frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\left\|\tilde{W}_{m2}\right\|^2+\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\left\|W_{m2}\right\|^2\left\|\tilde{V}_2\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\beta_7\varepsilon_{NFPK2} \\
& -\frac{\alpha_u\beta_8}{4}\frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\left\|\tilde{W}_{u2}\right\|^2+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2\left\|\tilde{V}_2\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}+\beta_8\varepsilon_{Nu1}
\end{aligned}
$$

$$
\begin{aligned}
\leq & -\frac{\gamma_1\beta_1}{2}\left\|x_1\right\|^2-\frac{\gamma_1\beta_1}{2}\left\|x_1\right\|^2-\beta_1\operatorname{tr}\left\{x_1^T g_1(x_1)\tilde{u}_1\right\}-\frac{2\beta_1}{\gamma_1}\left\|g_1(x_1)\tilde{u}_1\right\|^2 \\
& +\frac{2\beta_1}{\gamma_1}\left\|g_1(x_1)\tilde{u}_1\right\|^2-\frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}\right\|^2
\end{aligned}
$$

$$
\begin{aligned}
&+ \alpha_h \frac{\beta_2 \left[ L_\Phi + L_{\Psi V1} \left\| W_{V1} \right\|^2 \right] \left\| \tilde{m}_1 \tilde{m}_2 \right\|^2}{1 + \left\| \hat{\Psi}_{V1} \right\|^2} - \frac{\alpha_m \beta_3}{2} \frac{\left\| \hat{\Psi}_{m1} \right\|^2}{1 + \left\| \hat{\Psi}_{m1} \right\|^2} \left\| \tilde{W}_{m1} \right\|^2 \\
&+ \alpha_m \frac{\beta_3 L_{\Psi_{m1}} \left\| W_{m1} \right\|^2 \left\| \tilde{V}_1 \right\|^2}{1 + \left\| \hat{\Psi}_{m1} \right\|^2} - \frac{\alpha_u \beta_4}{4} \frac{\left\| \hat{\phi}_{u1} \right\|^2}{1 + \left\| \hat{\phi}_{u1} \right\|^2} \left\| \tilde{W}_{u1} \right\|^2 \\
&+ \alpha_u \beta_4 \frac{\left\| R_1^{-1} g_1^T \left( x_1 \right) \right\|^2 \left\| \tilde{V}_1 \right\|^2}{1 + \left\| \hat{\phi}_{u1} \right\|^2} + \beta_4 \varepsilon_{Nu1} + \beta_3 \varepsilon_{NFPK1} + \beta_2 \varepsilon_{VHJI1} \\
&- \frac{\gamma_2 \beta_5}{2} \left\| x_2 \right\|^2 - \frac{\gamma_2 \beta_5}{2} \left\| x_2 \right\|^2 - \beta_5 \operatorname{tr} \left\{ x_2^T g_2 \left( x_2 \right) \tilde{u}_2 \right\} - \frac{2\beta_5}{\gamma_2} \left\| g_2 \left( x_2 \right) \tilde{u}_2 \right\|^2 \\
&+ \frac{2\beta_5}{\gamma_2} \left\| g_2 \left( x_2 \right) \tilde{u}_2 \right\|^2 - \frac{\alpha_h \beta_6}{4} \frac{\left\| \hat{\Psi}_{V2} \right\|^2}{1 + \left\| \hat{\Psi}_{V2} \right\|^2} \left\| \tilde{W}_{V2} \right\|^2 \\
&+ \alpha_h \frac{\beta_6 \left[ L_\Phi + L_{\Psi V2} \left\| W_{V2} \right\|^2 \right] \left\| \tilde{m}_1 \tilde{m}_2 \right\|^2}{1 + \left\| \hat{\Psi}_{V2} \right\|^2} - \frac{\alpha_m \beta_7}{2} \frac{\left\| \hat{\Psi}_{m2} \right\|^2}{1 + \left\| \hat{\Psi}_{m2} \right\|^2} \left\| \tilde{W}_{m2} \right\|^2 \\
&+ \alpha_m \frac{\beta_7 L_{\Psi_{m2}} \left\| W_{m2} \right\|^2 \left\| \tilde{V}_2 \right\|^2}{1 + \left\| \hat{\Psi}_{m2} \right\|^2} - \frac{\alpha_u \beta_8}{4} \frac{\left\| \hat{\phi}_{u2} \right\|^2}{1 + \left\| \hat{\phi}_{u2} \right\|^2} \left\| \tilde{W}_{u2} \right\|^2 \\
&+ \alpha_u \beta_8 \frac{\left\| R_2^{-1} g_2^T \left( x_2 \right) \right\|^2 \left\| \tilde{V}_2 \right\|^2}{1 + \left\| \hat{\phi}_{u2} \right\|^2} + \beta_8 \varepsilon_{Nu2} + \beta_7 \varepsilon_{NFPK2} + \beta_6 \varepsilon_{VHJI2}
\end{aligned}
$$

$$
\begin{aligned}
\leq &- \frac{\gamma_1 \beta_1}{2} \left\| x_1 \right\|^2 - \beta_1 \left[ \sqrt{\frac{\gamma_1}{2}} \| x_1 \| + \sqrt{\frac{2}{\gamma_1}} \| g_1(x_1) \tilde{u}_1 \right]^2 + \frac{2 g_{M1}^2 \beta_1}{\gamma_1} \| \tilde{u}_1 \|^2 \\
&- \frac{\alpha_h \beta_2}{4} \frac{\left\| \hat{\Psi}_{V1} \right\|^2}{1 + \left\| \hat{\Psi}_{V1} \right\|^2} \left\| \tilde{W}_{V1} \right\|^2 + \alpha_h \frac{\beta_2 \left[ L_\Phi + L_{\Psi V1} \left\| W_{V1} \right\|^2 \right] \left\| \tilde{m}_1 \tilde{m}_2 \right\|^2}{1 + \left\| \hat{\Psi}_{V1} \right\|^2} \\
&- \frac{\alpha_m \beta_3}{2} \frac{\left\| \hat{\Psi}_{m1} \right\|^2}{1 + \left\| \hat{\Psi}_{m1} \right\|^2} \left\| \tilde{W}_{m1} \right\|^2 + \alpha_m \frac{\beta_3 L_{\Psi_{m1}} \left\| W_{m1} \right\|^2 \left\| \tilde{V}_1 \right\|^2}{1 + \left\| \hat{\Psi}_{m1} \right\|^2} + \beta_2 \varepsilon_{VHJI1} \\
&- \frac{\alpha_u \beta_4}{4} \frac{\left\| \hat{\phi}_{u1} \right\|^2}{1 + \left\| \hat{\phi}_{u1} \right\|^2} \left\| \tilde{W}_{u1} \right\|^2 + \alpha_u \beta_4 \frac{\left\| R_1^{-1} g_1^T \left( x_1 \right) \right\|^2 \left\| \tilde{V}_1 \right\|^2}{1 + \left\| \hat{\phi}_{u1} \right\|^2} + \beta_4 \varepsilon_{Nu1} + \beta_3 \varepsilon_{NFPK1}
\end{aligned}
$$

$$
\begin{aligned}
&- \frac{\gamma_2\beta_5}{2}\|x_2\|^2 - \beta_5\left[\sqrt{\frac{\gamma_2}{2}}\|x_2\| + \sqrt{\frac{2}{\gamma_2}}\|g_2(x_2)\tilde{u}_2\right]^2 + \frac{2g_{M2}^2\beta_5}{\gamma_2}\|\tilde{u}_2\|^2 \\
&- \frac{\alpha_h\beta_6}{4}\frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\left\|\tilde{W}_{V2}\right\|^2 + \alpha_h\frac{\beta_6\left[L_\Phi + L_{\Psi V2}\|W_{V2}\|^2\right]\|\tilde{m}_1\tilde{m}_2\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2} \\
&- \frac{\alpha_m\beta_7}{2}\frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\left\|\tilde{W}_{m2}\right\|^2 + \alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2\left\|\tilde{V}_2\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} + \beta_6\varepsilon_{VHJI2} \\
&- \frac{\alpha_u\beta_8}{8}\frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\left\|\tilde{W}_{u2}\right\|^2 + \alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2\left\|\tilde{V}_2\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} + \beta_8\varepsilon_{Nu2} + \beta_7\varepsilon_{NFPK2}
\end{aligned}
$$

$$
\begin{aligned}
\leq &- \frac{\gamma_1}{2}\beta_1\|x_1\|^2 + \frac{2g_{M1}^2\beta_1}{\gamma_1}\|\tilde{u}_1\|^2 - \frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}\right\|^2 \\
&+ \alpha_h\frac{\beta_2\left[L_\Phi + L_{\Psi V1}\|W_{V1}\|^2\right]\|\tilde{m}_1\tilde{m}_2\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} - \frac{\alpha_m\beta_3}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}\right\|^2 \\
&- \frac{\alpha_u\beta_4}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}\right\|^2 + \beta_4\varepsilon_{Nu1} + \beta_3\varepsilon_{NFPK1} + \beta_2\varepsilon_{VHJI1} \\
&+ \left[\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} + \alpha_{u1}\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\left\|\tilde{V}_1\right\|^2 \\
&- \frac{\gamma_2}{2}\beta_5\|x_2\|^2 + \frac{2g_{M2}^2\beta_5}{\gamma_2}\|\tilde{u}_2\|^2 - \frac{\alpha_h\beta_6}{4}\frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\left\|\tilde{W}_{V2}\right\|^2 \\
&+ \alpha_h\frac{\beta_6\left[L_\Phi + L_{\Psi V2}\|W_{V2}\|^2\right]\|\tilde{m}_1\tilde{m}_2\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2} - \frac{\alpha_m\beta_7}{2}\frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\left\|\tilde{W}_{m2}\right\|^2 \\
&- \frac{\alpha_u\beta_8}{4}\frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\left\|\tilde{W}_{u2}\right\|^2 + \beta_8\varepsilon_{Nu2} + \beta_7\varepsilon_{NFPK2} + \beta_6\varepsilon_{VHJI2}
\end{aligned}
$$

$$+\left[\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\left\|W_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T\left(x_2\right)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\left\|\tilde{V}_2\right\|^2 \tag{3.75}$$

where $g_{M1}^2$ is the upper bound of $g_1^2(x_1)$, $g_{M2}^2$ is the upper bound of $g_2^2(x_2)$

Next, substituting (3.45) into (3.75), (3.75) can be represented as

$$\begin{aligned}
&\dot{L}_{sys}(t)\\
&\leq-\frac{\gamma_1}{2}\beta_1\|x_1\|^2+\frac{2g_{M1}^2\beta_1}{\gamma_1}\|\tilde{u}_1\|^2-\frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}\right\|^2\\
&\quad+\alpha_h\frac{\beta_2\left[L_\Phi+L_{\Psi V1}\left\|W_{V1}\right\|^2\right]\left\|\tilde{m}_1\tilde{m}_2\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}-\frac{\alpha_m\beta_3}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}\right\|^2\\
&\quad-\frac{\alpha_u\beta_4}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}\right\|^2+\beta_4\varepsilon_{Nu1}+\beta_3\varepsilon_{NFPK1}+\beta_2\varepsilon_{VHJI1}\\
&\quad+\begin{bmatrix}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\\+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{bmatrix}\begin{bmatrix}\|\tilde{W}_{V1}(t)\|\|\hat{\phi}_{V1}\|\\+L_{\phi v1}\|W_{V1}\|\|\tilde{m}_1\tilde{m}_2\|+\|\varepsilon_{HJI1}\|\end{bmatrix}^2\\
&\quad-\frac{\gamma_2}{2}\beta_5\|x_2\|^2+\frac{2g_{M2}^2\beta_5}{\gamma_2}\|\tilde{u}_2\|^2-\frac{\alpha_h\beta_6}{4}\frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\left\|\tilde{W}_{V2}\right\|^2\\
&\quad+\alpha_h\frac{\beta_6\left[L_\Phi+L_{\Psi V2}\left\|W_{V2}\right\|^2\right]\left\|\tilde{m}_1\tilde{m}_2\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}-\frac{\alpha_m\beta_7}{2}\frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\left\|\tilde{W}_{m2}\right\|^2\\
&\quad-\frac{\alpha_u\beta_8}{4}\frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\left\|\tilde{W}_{u2}\right\|^2+\beta_8\varepsilon_{Nu2}+\beta_7\varepsilon_{NFPK2}+\beta_6\varepsilon_{VHJI2}\\
&\quad+\begin{bmatrix}\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\\+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\end{bmatrix}\begin{bmatrix}\|\tilde{W}_{V2}(t)\|\|\hat{\phi}_{V2}\|+L_{\phi v2}\|W_{V2}\|\|\tilde{m}_1\tilde{m}_2\|\\+\|\varepsilon_{HJI2}\|\end{bmatrix}^2
\end{aligned}$$

$$
\begin{aligned}
\leq & -\frac{\gamma_1}{2}\beta_1\|x_1\|^2+\frac{2g_{M1}^2\beta_1}{\gamma_1}\|\tilde{u}_1\|^2-\frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}\right\|^2 \\
& -\frac{\alpha_m\beta_3}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}\right\|^2-\frac{\alpha_u\beta_4}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}\right\|^2+\beta_4\varepsilon_{Nu1}+\beta_3\varepsilon_{NFPK1} \\
& +\beta_2\varepsilon_{VHJI1}+3\left[\alpha_m\frac{\beta_3L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\|\tilde{W}_{V1}(t)\|^2\|\hat{\phi}_{V1}\|^2 \\
& +\left[\begin{array}{c}3\left[\alpha_m\frac{\beta_3L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]L_{\phi v1}^2\|W_{V1}\|^2 \\ +\alpha_h\frac{\beta_2\left[L_\Phi+L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\end{array}\right]\|\tilde{m}_1\tilde{m}_2\|^2 \\
& +3\left[\alpha_m\frac{\beta_3L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\|\varepsilon_{HJI1}\|^2 \\
& -\frac{\gamma_2}{2}\beta_5\|x_2\|^2+\frac{2g_{M2}^2\beta_5}{\gamma_2}\|\tilde{u}_2\|^2-\frac{\alpha_h\beta_6}{4}\frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\left\|\tilde{W}_{V2}\right\|^2+\beta_6\varepsilon_{VHJI2} \\
& -\frac{\alpha_m\beta_7}{2}\frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\left\|\tilde{W}_{m2}\right\|^2-\frac{\alpha_u\beta_8}{4}\frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\left\|\tilde{W}_{u2}\right\|^2+\beta_8\varepsilon_{Nu2}+\beta_7\varepsilon_{NFPK2} \\
& +3\left[\alpha_m\frac{\beta_7L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\|\tilde{W}_{V2}(t)\|^2\|\hat{\phi}_{V2}\|^2 \\
& +\left[\begin{array}{c}3\left[\alpha_m\frac{\beta_7L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]L_{\phi v2}^2\|W_{V2}\|^2 \\ +\alpha_h\frac{\beta_6\left[L_\Phi+L_{\Psi V2}\|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\end{array}\right]\|\tilde{m}_1\tilde{m}_2\|^2 \\
& +3\left[\alpha_m\frac{\beta_7L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\|\varepsilon_{HJI2}\|^2
\end{aligned}
\tag{3.76}
$$

Furthermore, substituting (3.58) into (3.76), (3.76) can be represented as

$$
\begin{aligned}
&\dot{L}_{sys}(t)\\
&\leq -\frac{\gamma_1}{2}\beta_1\|x_1\|^2 + \frac{2g_{M1}^2\beta_1}{\gamma_1}\|\tilde{u}_1\|^2 - \frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}\right\|^2\\
&+\begin{bmatrix} 3\begin{bmatrix}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\\ +\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{bmatrix}L_{\phi v1}^2\|W_{V1}\|^2\\ +\alpha_h\frac{\beta_2\left[L_\Phi + L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\end{bmatrix}\|\tilde{m}_2\|^2\begin{bmatrix}\|\tilde{W}_{m1}(t)\|\|\phi_{m1}\|\\ +\|\varepsilon_{FPK1}\|\end{bmatrix}^2\\
&-\frac{\alpha_m\beta_3}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}\right\|^2 - \frac{\alpha_u\beta_4}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}\right\|^2 + \beta_4\varepsilon_{Nu1} + \beta_3\varepsilon_{NFPK1}\\
&+3\left[\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} + \alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\|\tilde{W}_{V1}(t)\|^2\|\hat{\phi}_{V1}\|^2 + \beta_2\varepsilon_{VHJI1}\\
&+3\left[\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} + \alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\|\varepsilon_{HJI1}\|^2\\
&-\frac{\gamma_2}{2}\beta_5\|x_2\|^2 + \frac{2g_{M2}^2\beta_5}{\gamma_2}\|\tilde{u}_2\|^2 - \frac{\alpha_h\beta_6}{4}\frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\left\|\tilde{W}_{V2}\right\|^2\\
&+\begin{bmatrix} 3\begin{bmatrix}\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\\ +\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\end{bmatrix}L_{\phi v2}^2\|W_{V2}\|^2\\ +\alpha_h\frac{\beta_6\left[L_\Phi + L_{\Psi V2}\|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\end{bmatrix}\|\tilde{m}_1\|^2\left[\|\tilde{W}_{m2}(t)\|\|\phi_{m2}\| + \|\varepsilon_{FPK2}\|\right]^2\\
&-\frac{\alpha_m\beta_7}{2}\frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\left\|\tilde{W}_{m2}\right\|^2 - \frac{\alpha_u\beta_8}{4}\frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\left\|\tilde{W}_{u2}\right\|^2 + \beta_8\varepsilon_{Nu2} + \beta_7\varepsilon_{NFPK2}
\end{aligned}
$$

$$+3\left[\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\|\tilde{W}_{V2}(t)\|^2\|\hat{\phi}_{V2}\|^{+}\beta_6\varepsilon_{VHJI2}$$

$$+3\left[\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\|\varepsilon_{HJI2}\|^2$$

$$\leq-\frac{\gamma_1}{2}\beta_1\|x_1\|^2+\frac{2g_{M1}^2\beta_1}{\gamma_1}\|\tilde{u}_1\|^2-\frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}\right\|^2$$

$$+2\left[\begin{array}{l}3\left[\begin{array}{c}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\\+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{array}\right]L_{\phi v1}^2\|W_{V1}\|^2\\+\alpha_h\frac{\beta_2\left[L_\Phi+L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\end{array}\right]\|\tilde{m}_2\|^2\|\tilde{W}_{m1}(t)\|^2\|\phi_{m1}\|^2$$

$$+2\left[\begin{array}{l}3\left[\begin{array}{c}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\\+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{array}\right]L_{\phi v1}^2\|W_{V1}\|^2\\+\alpha_h\frac{\beta_2\left[L_\Phi+L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\end{array}\right]\|\tilde{m}_2\|^2\|\varepsilon_{FPK}\|^2$$

$$-\frac{\alpha_m\beta_3}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}\right\|^2-\frac{\alpha_u\beta_4}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}\right\|^2+\beta_4\varepsilon_{Nu1}+\beta_3\varepsilon_{NFPK1}$$

$$+3\left[\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\|\tilde{W}_{V1}(t)\|^2\|\hat{\phi}_{V1}\|^2+\beta_2\varepsilon_{VHJI1}$$

$$+3\left[\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\|\varepsilon_{HJI1}\|^2$$

$$-\frac{\gamma_2}{2}\beta_5\|x_2\|^2+\frac{2g_{M2}^2\beta_5}{\gamma_2}\|\tilde{u}_2\|^2-\frac{\alpha_h\beta_6}{4}\frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\left\|\tilde{W}_{V2}\right\|^2$$

$$
\begin{aligned}
&+2\left[\begin{array}{c}3\left[\begin{array}{c}\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\\+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\end{array}\right]L_{\phi v2}^2\|W_{V2}\|^2\\+\alpha_h\frac{\beta_6\left[L_\Phi+L_{\Psi V2}\|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\end{array}\right]\|\tilde{m}_1\|^2\|\tilde{W}_{m2}(t)\|^2\|\phi_{m2}\|^2\\
&+2\left[\begin{array}{c}3\left[\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]L_{\phi v2}^2\|W_{V2}\|^2\\+\alpha_h\frac{\beta_6\left[L_\Phi+L_{\Psi V2}\|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\end{array}\right]\|\tilde{m}_1\|^2\|\varepsilon_{FPK}\|^2\\
&-\frac{\alpha_m\beta_7}{2}\frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\left\|\tilde{W}_{m2}\right\|^2-\frac{\alpha_u\beta_8}{4}\frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\left\|\tilde{W}_{u2}\right\|^2+\beta_8\varepsilon_{Nu2}+\beta_7\varepsilon_{NFPK2}\\
&+3\left[\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\|\tilde{W}_{V2}(t)\|^2\|\hat{\phi}_{V2}\|^2+\beta_6\varepsilon_{VHJI2}\\
&+3\left[\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\|\varepsilon_{HJI2}\|^2
\end{aligned}
\tag{3.77}
$$

Next, substitute (3.71) into (3.77), (3.77) can be represented as:

$$
\begin{aligned}
&\dot{L}_{sys}(t)\\
&\leq-\frac{\gamma_1}{2}\beta_1\|x_1\|^2+\frac{2g_{M1}^2\beta_1}{\gamma_1}\left[\begin{array}{c}\|\tilde{W}_u(t)\|\|\hat{\phi}_{u1}\|\\+L_{\phi u}\|W_u\|\|\tilde{m}_1\tilde{m}_2\|+\|\varepsilon_{u1}\|\end{array}\right]^2\\
&-\frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}\right\|^2+\beta_2\varepsilon_{VHJI1}
\end{aligned}
$$

$$+2\left[\begin{array}{c}3\left[\begin{array}{c}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\\+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{array}\right]L_{\phi v1}^2\|W_{V1}\|^2\\+\alpha_h\dfrac{\beta_2\left[L_\Phi+L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\end{array}\right]\|\tilde{m}_2\|^2\|\tilde{W}_{m1}(t)\|^2\|\phi_{m1}\|^2$$

$$+2\left[\begin{array}{c}3\left[\begin{array}{c}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\\+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{array}\right]L_{\phi v1}^2\|W_{V1}\|^2\\+\alpha_h\dfrac{\beta_2\left[L_\Phi+L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\end{array}\right]\|\tilde{m}_2\|^2\|\varepsilon_{FPK}\|^2$$

$$-\frac{\alpha_m\beta_3}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\left\|\tilde{W}_{m1}\right\|^2-\frac{\alpha_u\beta_4}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\left\|\tilde{W}_{u1}\right\|^2+\beta_4\varepsilon_{Nu1}+\beta_3\varepsilon_{NFPK1}$$

$$+3\left[\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\|\tilde{W}_{V1}(t)\|^2\|\hat{\phi}_{V1}\|^2$$

$$+3\left[\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}+\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\|\varepsilon_{HJI1}\|^2$$

$$-\frac{\gamma_2}{2}\beta_5\|x_2\|^2+\frac{2g_{M2}^2\beta_5}{\gamma_2}\left[\|\tilde{W}_{u2}(t)\|\|\hat{\phi}_{u2}\|+L_{\phi u2}\|W_{u2}\|\|m\tilde{1}m\tilde{2}\|+\|\varepsilon_{u2}\|\right]^2$$

$$-\frac{\alpha_h\beta_6}{4}\frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\left\|\tilde{W}_{V2}\right\|^2$$

$$+2\left[\begin{array}{c}3\left[\begin{array}{c}\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\\+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\end{array}\right]L_{\phi v2}^2\|W_{V2}\|^2\\+\alpha_h\dfrac{\beta_6\left[L_\Phi+L_{\Psi V2}\|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\end{array}\right]\|\tilde{m}_1\|^2\|\tilde{W}_{m2}(t)\|^2\|\phi_{m2}\|^2$$

$$
\begin{aligned}
&+2\left[\begin{array}{l}3\left[\begin{array}{l}\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\\ +\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\end{array}\right]L_{\phi v2}^2\|W_{V2}\|^2\\ +\alpha_h\frac{\beta_6\left[L_\Phi+L_{\Psi V2}\|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2}\end{array}\right]\|\tilde{m}_1\|^2\|\varepsilon_{FPK}\|^2\\
&-\frac{\alpha_m\beta_7}{2}\frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}\left\|\tilde{W}_{m2}\right\|^2-\frac{\alpha_u\beta_8}{4}\frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\left\|\tilde{W}_{u2}\right\|^2+\beta_8\varepsilon_{Nu2}+\beta_7\varepsilon_{NFPK2}\\
&+3\left[\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\|\tilde{W}_{V2}(t)\|^2\|\hat{\phi}_{V2}\|^2+\beta_6\varepsilon_{VHJI2}\\
&+3\left[\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2}+\alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\|\varepsilon_{HJI2}\|^2
\end{aligned}
$$

$$
\begin{aligned}
&\le-\frac{\gamma_1}{2}\beta_1\|x_1\|^2+\frac{6g_{M1}^2\beta_1}{\gamma_1}\|\hat{\phi}_{u1}\|^2\|\tilde{W}_u(t)\|^2-\frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2}\left\|\tilde{W}_{V1}\right\|^2\\
&+2\left[\begin{array}{l}3\left[\begin{array}{l}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\\ +\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{array}\right]L_{\phi v1}^2\|W_{V1}\|^2\\ +\alpha_h\frac{\beta_2\left[L_\Phi+L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2}+\frac{6g_{M1}^2\beta_1}{\gamma}L_{\phi u}^2\|W_u\|^2\end{array}\right]\|\tilde{m}_2\|^2\|\tilde{W}_{m1}(t)\|^2\|\phi_{m1}\|^2\\
&+2\left[\begin{array}{l}3\left[\begin{array}{l}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2}\\ +\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{array}\right]L_{\phi v1}^2\|W_{V1}\|^2\\ +\alpha_h\frac{\beta_2\left[L_\Phi+L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2}+\frac{6g_{M1}^2\beta_1}{\gamma}L_{\phi u}^2\|W_u\|^2\end{array}\right]\|\tilde{m}_2\|^2\|\varepsilon_{FPK}\|^2\\
&+\frac{6g_{M1}^2\beta_1}{\gamma}\|\varepsilon_{u1}\|^2+\beta_3\varepsilon_{NFPK1}+\beta_2\varepsilon_{VHJI1}
\end{aligned}
$$

$$
\begin{aligned}
&- \frac{\alpha_m \beta_3}{2} \frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \left\|\tilde{W}_{m1}\right\|^2 - \frac{\alpha_u \beta_4}{4} \frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} \left\|\tilde{W}_{u1}\right\|^2 + \beta_4 \varepsilon_{Nu1} \\
&+ 3\left[\alpha_m \frac{\beta_3 L_{\Psi_{m1}} \|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} + \alpha_u \beta_4 \frac{\left\|R_1^{-1} g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right] \|\tilde{W}_{V1}(t)\|^2 \|\hat{\phi}_{V1}\|^2 \\
&+ 3\left[\alpha_m \frac{\beta_3 L_{\Psi_{m1}} \|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} + \alpha_u \beta_4 \frac{\left\|R_1^{-1} g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right] \|\varepsilon_{HJI1}\|^2 \\
&- \frac{\gamma_2}{2}\beta_5 \|x_2\|^2 + \frac{6 g_{M2}^2 \beta_5}{\gamma_2} \|\hat{\phi}_{u2}\|^2 \|\tilde{W}_{u2}(t)\|^2 - \frac{\alpha_h \beta_6}{4} \frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2} \left\|\tilde{W}_{V2}\right\|^2 \\
&+ 2\left[\begin{array}{l} 3\left[\begin{array}{l} \alpha_m \frac{\beta_7 L_{\Psi_{m2}} \|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} \\ +\alpha_u \beta_8 \frac{\left\|R_2^{-1} g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} \end{array}\right] L_{\phi v2}^2 \|W_{V2}\|^2 \\ + \alpha_h \frac{\beta_6 \left[L_\Phi + L_{\Psi V2} \|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2} + \frac{6 g_{M2}^2 \beta_5}{\gamma} L_{\phi u2}^2 \|W_{u2}\|^2 \end{array}\right] \|\tilde{m}_1\|^2 \|\tilde{W}_{m2}(t)\|^2 \|\phi_{m2}\|^2 \\
&+ 2\left[\begin{array}{l} \left[\begin{array}{l} \alpha_m \frac{\beta_7 L_{\Psi_{m2}} \|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} \\ +\alpha_u \beta_8 \frac{\left\|R_2^{-1} g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} \end{array}\right] L_{\phi v2}^2 \|W_{V2}\|^2 \\ + \alpha_h \frac{\beta_6 \left[L_\Phi + L_{\Psi V2} \|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2} + \frac{6 g_{M2}^2 \beta_5}{\gamma} L_{\phi u2}^2 \|W_{u2}\|^2 \end{array}\right] \|\tilde{m}_1\|^2 \|\varepsilon_{FPK}\|^2 \\
&+ \frac{6 g_{M2}^2 \beta_5}{\gamma} \|\varepsilon_{u2}\|^2 + \beta_8 \varepsilon_{Nu2} + \beta_7 \varepsilon_{NFPK2} + \beta_6 \varepsilon_{VHJI2} \\
&- \frac{\alpha_m \beta_7}{2} \frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} \left\|\tilde{W}_{m2}\right\|^2 - \frac{\alpha_u \beta_8}{4} \frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} \left\|\tilde{W}_{u2}\right\|^2 \\
&+ 3\left[\alpha_m \frac{\beta_7 L_{\Psi_{m2}} \|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} + \alpha_u \beta_8 \frac{\left\|R_2^{-1} g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right] \|\tilde{W}_{V2}(t)\|^2 \|\hat{\phi}_{V2}\|^2 \\
&+ 3\left[\alpha_m \frac{\beta_7 L_{\Psi_{m2}} \|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} + \alpha_u \beta_8 \frac{\left\|R_2^{-1} g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right] \|\varepsilon_{HJI2}\|^2
\end{aligned}
\tag{3.78}
$$

Combine the terms in (3.78) yields:

$$\dot{L}_{sys}(t) \leq -\frac{\gamma_1}{2}\beta_1\|x_1\|^2 - \left[\frac{\alpha_u\beta_4}{4}\frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} - \frac{6g_{M1}^2\beta_1}{\gamma_1}\|\hat{\phi}_{u1}\|^2\right]\|\tilde{W}_u(t)\|^2$$

$$- \left[\frac{\alpha_h\beta_2}{4}\frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} - 3\begin{bmatrix}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \\ +\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{bmatrix}\|\hat{\phi}_{V1}\|^2\right]\|\tilde{W}_{V1}(t)\|^2$$

$$- \begin{bmatrix}\frac{\alpha_m\beta_3}{2}\frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \\ -2\begin{bmatrix}3\begin{bmatrix}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \\ +\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{bmatrix}L_{\phi v1}^2\|W_{V1}\|^2 \\ +\alpha_h\frac{\beta_2\left[L_\Phi + L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \\ +\frac{6g_{M1}^2\beta_1}{\gamma_1}L_{\phi u}^2\|W_{u1}\|^2\end{bmatrix}\|\tilde{m}_2\|^2\|\phi_{m1}\|^2\end{bmatrix}\left\|\tilde{W}_{m1}\right\|^2$$

$$+ 2\begin{bmatrix}3\begin{bmatrix}\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \\ +\alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{bmatrix}L_{\phi v1}^2\|W_{V1}\|^2 \\ +\alpha_h\frac{\beta_2\left[L_\Phi + L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \\ +\frac{6g_{M1}^2\beta_1}{\gamma_1}L_{\phi u}^2\|W_u\|^2\end{bmatrix}\|\tilde{m}_2\|^2\|\varepsilon_{FPK}\|^2 + \frac{6g_{M1}^2\beta_1}{\gamma_1}\|\varepsilon_{u1}\|^2$$

$$+ \left[\alpha_m\frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} + \alpha_u\beta_4\frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\right]\|\varepsilon_{HJI1}\|^2 + \beta_4\varepsilon_{Nu1} + \beta_3\varepsilon_{NFPK1}$$

$$- \frac{\gamma_2}{2}\beta_5\|x_2\|^2 - \left[\frac{\alpha_u\beta_8}{4}\frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} - \frac{6g_{M2}^2\beta_5}{\gamma_2}\|\hat{\phi}_{u2}\|^2\right]\|\tilde{W}_{u2}(t)\|^2 + \beta_2\varepsilon_{VHJI1}$$

$$- \begin{bmatrix}\frac{\alpha_h\beta_6}{4}\frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2} \\ -3\left[\alpha_m\frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} + \alpha_u\beta_8\frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\right]\|\hat{\phi}_{V2}\|^2\end{bmatrix}\|\tilde{W}_{V2}(t)\|^2$$

$$
\begin{aligned}
&- \left[ \begin{array}{c} \frac{\alpha_m \beta_7}{2} \frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} \\ -2 \left[ \begin{array}{l} 3 \left[ \begin{array}{l} \alpha_m \frac{\beta_7 L_{\Psi_{m2}} \|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} \\ +\alpha_u \beta_8 \frac{\left\|R_2^{-1} g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} \end{array} \right] L_{\phi v2}^2 \|W_{V2}\| \\ + \alpha_h \frac{\beta_6 \left[L_\Phi + L_{\Psi V2} \|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2} \\ + \frac{6 g_{M2}^2 \beta_5}{\gamma_2} L_{\phi u2}^2 \|W_{u2}\|^2 \end{array} \right] \|\tilde{m}_1\|^2 \|\phi_{m2}\|^2 \end{array} \right] \left\|\tilde{W}_{m2}\right\|^2 \\
&+ 2 \left[ \begin{array}{l} 3 \left[ \begin{array}{l} \alpha_m \frac{\beta_7 L_{\Psi_{m2}} \|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} \\ +\alpha_u \beta_8 \frac{\left\|R_2^{-1} g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} \end{array} \right] L_{\phi v2}^2 \|W_{V2}\| \\ + \alpha_h \frac{\beta_6 \left[L_\Phi + L_{\Psi V2} \|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2} \\ + \frac{6 g_{M2}^2 \beta_5}{\gamma_2} L_{\phi u2}^2 \|W_{u2}\|^2 \end{array} \right] \|\tilde{m}_1\|^2 \|\varepsilon_{FPK2}\|^2 + \frac{6 g_{M2}^2 \beta_5}{\gamma_2} \|\varepsilon_{u2}\|^2 \\
&+ \left[ \alpha_m \frac{\beta_7 L_{\Psi_{m2}} \|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} + \alpha_u \beta_8 \frac{\left\|R_2^{-1} g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} \right] \|\varepsilon_{HJI2}\|^2 \\
&+ \beta_8 \varepsilon_{Nu2} + \beta_7 \varepsilon_{NFPK2} + \beta_6 \varepsilon_{VHJI2}
\end{aligned}
$$

$$
\begin{aligned}
&\leq -\frac{\gamma_1 \beta_1}{2} - \frac{\gamma_2 \beta_5}{2} - \kappa_{u1} \|\tilde{W}_{u1}\|^2 - \kappa_{m1} \|\tilde{W}_{m1}\|^2 - \kappa_{V1} \|\tilde{W}_{V1}\|^2 \\
&- \kappa_{u2} \|\tilde{W}_{u2}\|^2 - \kappa_{m2} \|\tilde{W}_{m2}\|^2 - \kappa_{V2} \|\tilde{W}_{V2}\|^2 + \varepsilon_{CLS1} + \varepsilon_{CLS2}
\end{aligned} \tag{3.79}
$$

with $\kappa$ and $\varepsilon$ parameters defined as

$$
\kappa_{u1} = \left[ \frac{\alpha_u \beta_4}{4} \frac{\left\|\hat{\phi}_{u1}\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} - \frac{6 g_{M1}^2 \beta_1}{\gamma_1} \|\hat{\phi}_{u1}\|^2 \right]
$$

$$\kappa_{m1} = \left[ \begin{array}{c} \frac{\alpha_m \beta_3}{2} \frac{\left\|\hat{\Psi}_{m1}\right\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \\ -2 \left[ \begin{array}{l} 3 \left[ \begin{array}{c} \alpha_m \frac{\beta_3 L_{\Psi_{m1}} \|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \\ +\alpha_u \beta_4 \frac{\left\|R_1^{-1} g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} \end{array} \right] L_{\phi v1}^2 \|W_{V1}\|^2 \\ + \alpha_h \frac{\beta_2 \left[L_\Phi + L_{\Psi V1} \|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \\ + \frac{6 g_{M1}^2 \beta_1}{\gamma_1} L_{\phi u}^2 \|W_{u1}\|^2 \end{array} \right] \|\tilde{m}_2\|^2 \|\phi_{m1}\|^2 \end{array} \right]$$

$$\kappa_{V1} = \left[ \frac{\alpha_h \beta_2}{4} \frac{\left\|\hat{\Psi}_{V1}\right\|^2}{1+\left\|\hat{\Psi}_{V1}\right\|^2} - 3 \left[ \begin{array}{c} \alpha_m \frac{\beta_3 L_{\Psi_{m1}} \|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \\ +\alpha_u \beta_4 \frac{\left\|R_1^{-1} g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2} \end{array} \right] \|\hat{\phi}_{V1}\|^2 \right]$$

$$\kappa_{u2} = \left[ \frac{\alpha_u \beta_8}{4} \frac{\left\|\hat{\phi}_{u2}\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} - \frac{6 g_{M2}^2 \beta_5}{\gamma_2} \|\hat{\phi}_{u2}\|^2 \right]$$

$$\kappa_{m2} = \left[ \begin{array}{l} \frac{\alpha_m \beta_7}{2} \frac{\left\|\hat{\Psi}_{m2}\right\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} - 2 \left[ \begin{array}{l} 3 \left[ \begin{array}{c} \alpha_m \frac{\beta_7 L_{\Psi_{m2}} \|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} \\ +\alpha_u \beta_8 \frac{\left\|R_2^{-1} g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} \end{array} \right] L_{\phi v2}^2 \|W_{V2}\| \\ + \alpha_h \frac{\beta_6 \left[L_\Phi + L_{\Psi V2} \|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2} \\ + \frac{6 g_{M2}^2 \beta_5}{\gamma_2} L_{\phi u2}^2 \|W_{u2}\|^2 \end{array} \right] \|\tilde{m}_1\|^2 \|\phi_{m2}\|^2 \end{array} \right]$$

$$\kappa_{V2} = \left[ \frac{\alpha_h \beta_6}{4} \frac{\left\|\hat{\Psi}_{V2}\right\|^2}{1+\left\|\hat{\Psi}_{V2}\right\|^2} - 3 \left[ \alpha_m \frac{\beta_7 L_{\Psi_{m2}} \|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} + \alpha_u \beta_8 \frac{\left\|R_2^{-1} g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2} \right] \|\hat{\phi}_{V2}\|^2 \right]$$

$$\varepsilon_{CLS1} = 2\left[\begin{array}{l} 3\left[\begin{array}{l}\alpha_m \frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \\ +\alpha_u\beta_4 \frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{array}\right] L_{\phi v1}^2\|W_{V1}\| \\ +\alpha_h \dfrac{\beta_2\left[L_\Phi + L_{\Psi V1}\|W_{V1}\|^2\right]}{1+\left\|\hat{\Psi}_{V1}\right\|^2} \\ +\dfrac{6g_{M1}^2\beta_1}{\gamma_1}L_{\phi u}^2\|W_u\|^2 \end{array}\right]\|\tilde{m}_2\|^2\|\varepsilon_{FPK}\|^2 + \frac{6g_{M1}^2\beta_1}{\gamma_1}\|\varepsilon_{u1}\|^2$$
$$+\left[\begin{array}{l}\alpha_m \frac{\beta_3 L_{\Psi_{m1}}\|W_{m1}\|^2}{1+\left\|\hat{\Psi}_{m1}\right\|^2} \\ +\alpha_u\beta_4 \frac{\left\|R_1^{-1}g_1^T(x_1)\right\|^2}{1+\left\|\hat{\phi}_{u1}\right\|^2}\end{array}\right]\|\varepsilon_{HJI1}\|^2 + \beta_4\varepsilon_{Nu1} + \beta_3\varepsilon_{NFPK1} + \beta_2\varepsilon_{VHJI1}$$

$$\varepsilon_{CLS2} = 2\left[\begin{array}{l} 3\left[\begin{array}{l}\alpha_m \frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} \\ +\alpha_u\beta_8 \frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\end{array}\right] L_{\phi v2}^2\|W_{V2}\| \\ +\alpha_h \dfrac{\beta_6\left[L_\Phi + L_{\Psi V2}\|W_{V2}\|^2\right]}{1+\left\|\hat{\Psi}_{V2}\right\|^2} \\ +\dfrac{6g_{M2}^2\beta_5}{\gamma_2}L_{\phi u2}^2\|W_{u2}\|^2 \end{array}\right]\|\tilde{m}_1\|^2\|\varepsilon_{FPK2}\|^2 + \frac{6g_{M2}^2\beta_5}{\gamma_2}\|\varepsilon_{u2}\|^2$$
$$+\left[\begin{array}{l}\alpha_m \frac{\beta_7 L_{\Psi_{m2}}\|W_{m2}\|^2}{1+\left\|\hat{\Psi}_{m2}\right\|^2} \\ +\alpha_u\beta_8 \frac{\left\|R_2^{-1}g_2^T(x_2)\right\|^2}{1+\left\|\hat{\phi}_{u2}\right\|^2}\end{array}\right]\|\varepsilon_{HJI2}\|^2 + \beta_8\varepsilon_{Nu2} + \beta_7\varepsilon_{NFPK2} + \beta_6\varepsilon_{VHJI2}$$

Note that the coefficient functions $\kappa_{u1}$, $\kappa_{m1}$, $\kappa_{V1}$, $\kappa_{u2}$, $\kappa_{m2}$, and $\kappa_{V2}$ are all positive definite, and the terms $\varepsilon_{CLS1}$ and $\varepsilon_{CLS2}$ go to zero if the reconstruction errors $\varepsilon_{HJI1}$, $\varepsilon_{FPK1}$, $\varepsilon_{u1}$, $\varepsilon_{HJI2}$, $\varepsilon_{FPK2}$, $\varepsilon_{u2}$ go to zero. The meaning of reconstruction error goes to zero means that the neural network structure and activation functions are perfectly selected. In that case, the first derivative of the Lyapunov function is negative definite which means the closed loop system is asymptotically stable. In the case where the reconstruction error is not zero, the closed loop system is Uniformly Ultimately Bounded (UUB). □

## 3.4 Numerical Simulations

In the numerical simulations, a pursuit-evasion game with very large scale pursuers and evaders are constructed. The evaders aim to get to a target while dodging the pursuers. The pursuers, on the other hand, focus on catching the evaders by mass capture. In the designed game, the pursuers and the evaders have heterogeneous stochastic nonlinear dynamics. Note that during the game, the evaders and pursuers are considered "blind", i.e., they will not have information from any other agents. Two tests are conducted to demonstrate the effectiveness of the proposed mean-field type of ACMO algorithm. Firstly, the pursuers use the developed algorithm while the evaders follow the traditional optimal control design. Secondly, all of the pursuers and evaders play the computed mean-field type of optimal control algorithm.

We employed 1000 pursuers as well as the same amount of evaders in total. The initial positions of all agents are randomly selected with the pursuers' initial positions following a normal distribution (i.e., $\mathcal{N}(\mu = [0, 3], \sigma = 0.1 \times I_2)$ and the evaders follow $\mathcal{N}(\mu = [3, 3], \sigma = 0.5 \times I_2)$. All agents' initial velocities are set to zero. Each pursuer agent's system dynamics are defined as:

Pursuer group $\mathcal{G}_1$ agents:

$$dx_{j,1}(t) = \begin{bmatrix} f_1\left(x_{j,1}\right) + l_1(x_{j,1})\bar{x}_2 \\ +g_1\left(x_{j,1}\right) u_{j,1}(t) \end{bmatrix} dt + D_1 dw_{j,1}(t), \quad 1 \leq j \leq N_1$$

where

$$f_1\left(x_{j,1}\right) = \begin{bmatrix} -x_{j,1,1} + 0.5x_{j,1,2}^2 \\ -0.3x_{j,1,2}^2 \end{bmatrix}, g_1\left(x_{j,1}\right) = \begin{bmatrix} x_{j,1,1} \\ 0 \end{bmatrix}$$

$$l_1(x_{j,1}) = \begin{bmatrix} 0.1 \\ 0 \end{bmatrix}, \qquad D_1 = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$$

with $x_{j,1} = [x_{j,1,1} \quad x_{j,1,2}]^T$. Similarly, the system dynamics for the evaders are given as:

$$dx_{i,2}(t) = \begin{bmatrix} f_2\left(x_{i,2}\right) + l_2(x_{i,2})\bar{x}_1 \\ +g_2\left(x_{i,2}\right) u_{i,2}(t) \end{bmatrix} dt + D_2 dw_{i,2}(t), \quad 1 \leq i \leq N_2$$

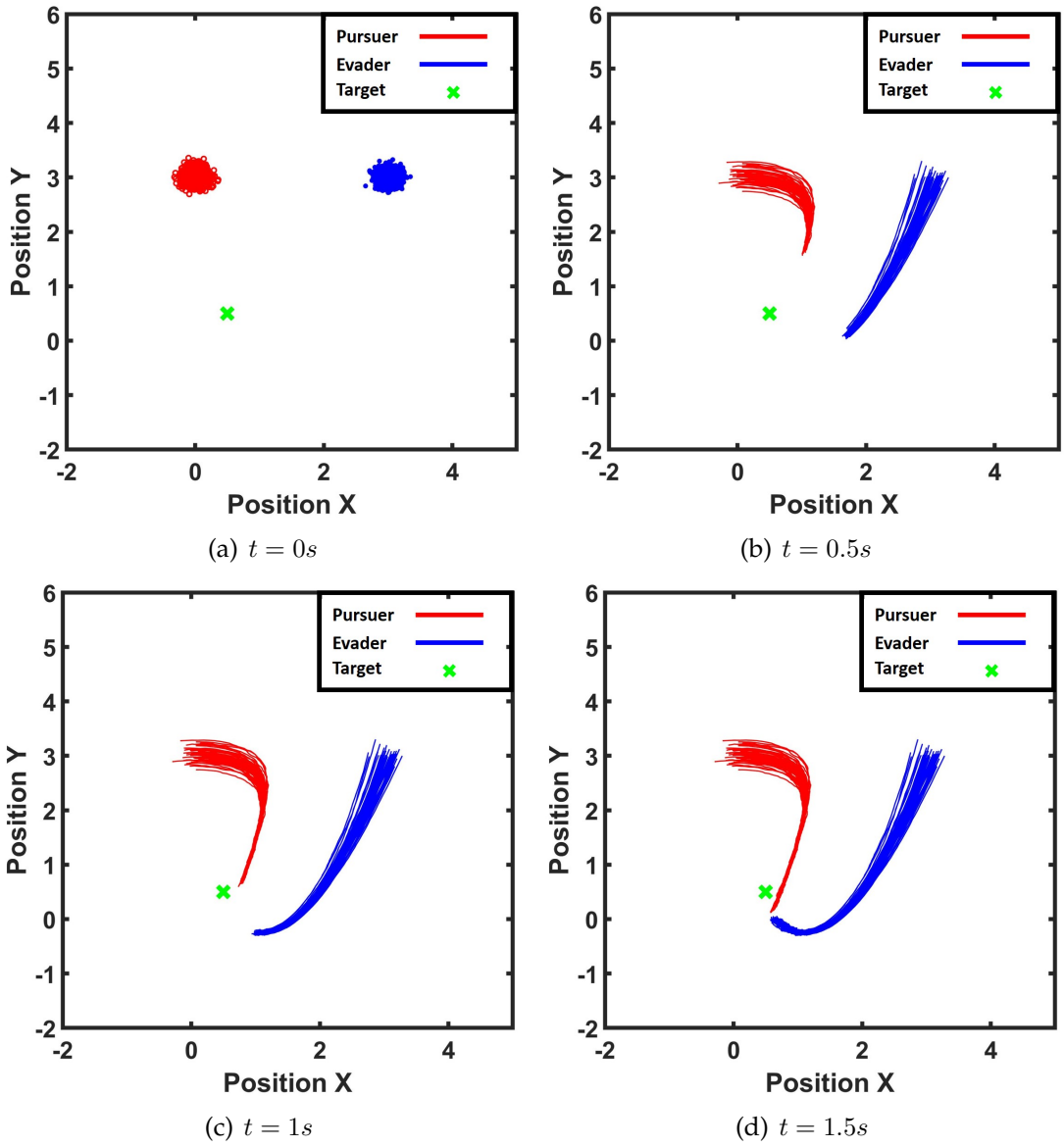
where

$$f_2\left(x_{i,2}\right) = \begin{bmatrix} -x_{i,2,1}^2 + x_{i,2,2} \\ -0.2x_{i,2,2}^2 \end{bmatrix}, \quad g_1\left(x_{j,1}\right) = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$

$$l_2(x_{i,2}) = \begin{bmatrix} 0 \\ 0.2 \end{bmatrix}, \qquad D_2 = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$$

Furthermore, we consider that the target for the evaders being the point $[0.5 \quad 0.5]^T$. In the cost functions (3.7) and (3.8) $R_1 = R_2 = 1$, $Q_1 = Q_2 = I_2$, $I_2$ represents the two-dimensional identity matrix. The learning rate of the neural networks are given as $\alpha_{h,0} = \alpha_{h,i} = 2 \times 10^{-6}$, $\alpha_{u,0} = \alpha_{u,i} = 2 \times 10^{-4}$, and $\alpha_{m,i} = 1 \times 10^{-3}$. The total simulation time is 70 seconds.

From a pursuer's perspective, it needs to estimate the optimal cost function, PDF of the pursuer group, the evaders' worst actions, and its own optimal control. Therefore, each pursuer maintains four neural networks whose activation functions are selected form the expansion terms of the polynomial $\sum_{\beta=1}^{M}(\sum_{j=1}^{n} z_j)\beta$ where $n$ represents the input number of the neural network, and $M$ stands for the estimation error. For the critic and actor neural networks, we selected $M = 4$. For the mass neural network, the constant $M$ is set to $5$.

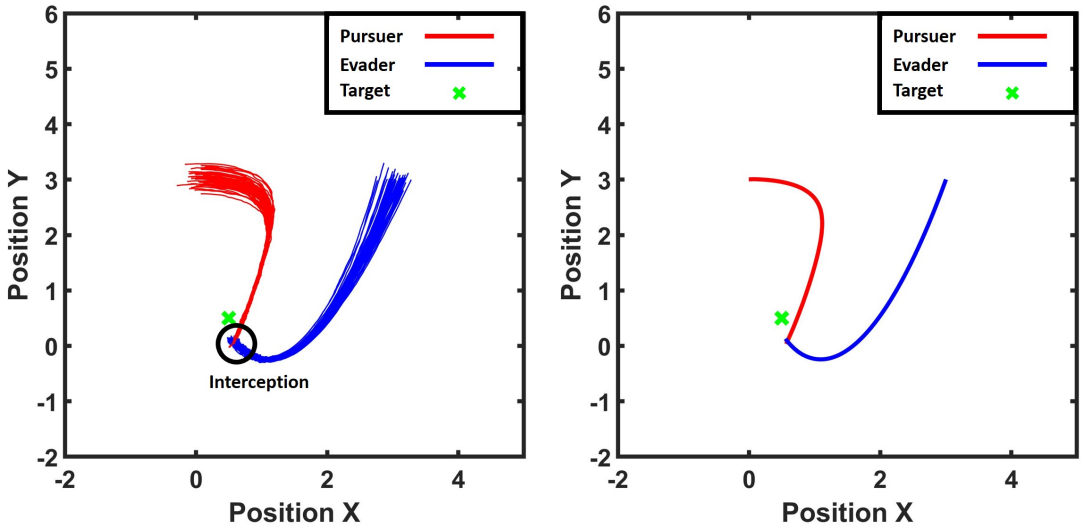



(a) $t = 0s$ (b) $t = 0.5s$

(c) $t = 1s$ (d) $t = 1.5s$

Figure 3.3: The trajectory of agents with respect to time. For this plot's visibility, only 50 pursuers and 50 evaders are plotted (total 1000).

The evaders' position is plotted as the blue star with the pursuers' positions plotted as the red circles. The trajectory of the pursuers and evaders are plotted as red and blue curves, respectively. The green cross marks the target position for the evaders.




(a) 50 agents' trajectory at the time of capture. (b) The average trajectory of two groups at the time of capture.

Figure 3.4: The plot of the agents' positions and trajectories at the time of capture. Pursuers' and evaders' trajectories are marked with red and blue curves, respectively. The left image shows the time evolution of the trajectories for 50 pursuers and 20 evaders. The right plot demonstrates the average trajectory of both pursuers' team and evaders' team.

Firstly, let the pursers follow the developed mean field type of ACMO algorithm, but the evaders play regular mean field type of optimal control method introduced in [119]. This game lasts 1.7 seconds when the pursuers catch the evaders. Note that due to the stochastic term in system dynamics, we use a loose version instead of the exact "mass capture" criteria, i.e., the evader group is said to be captured if the evader mass and the pursuer mass are bounded in a constant distance: $\|\mathbb{E}\{m_1^*\} - \mathbb{E}\{m_2^*\}\| \leq \epsilon_c$. The time evolution of all agents' overall positions is plotted in Fig. 3.3 which includes all agents' positions at initial status, 0.5s, 1s, and 1.5s. For a better visibility, only 50 pursuers and 20 evaders instead of 2000 agents are plotted. Since the evaders compute the optimal trajectory without considering the pursuers, they are captured after 1.7 seconds. The final captured trajectories

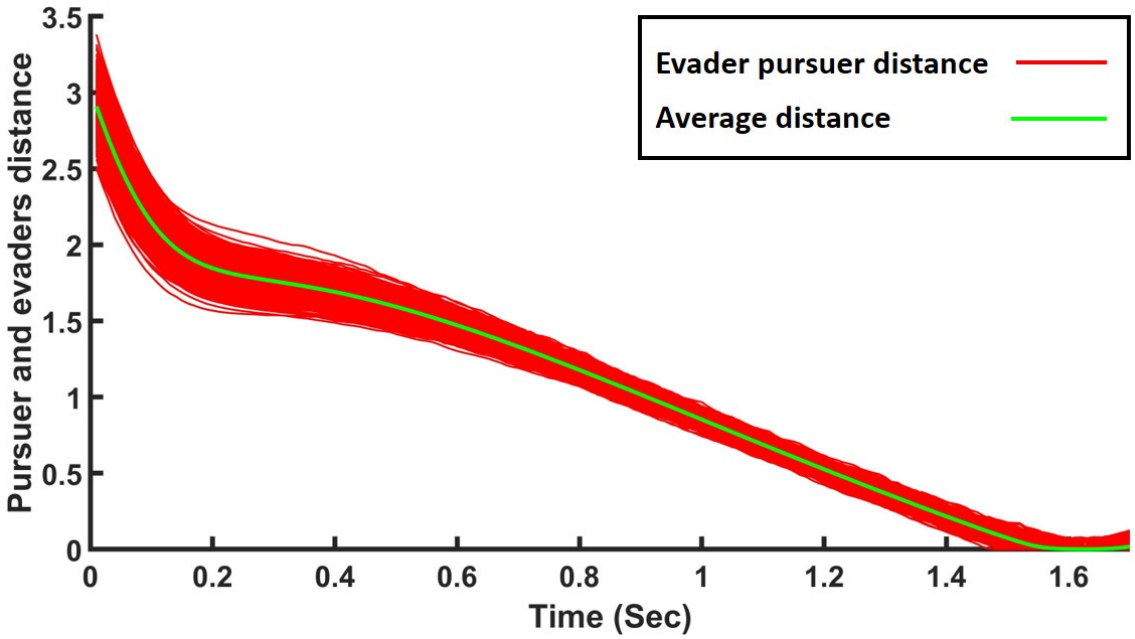
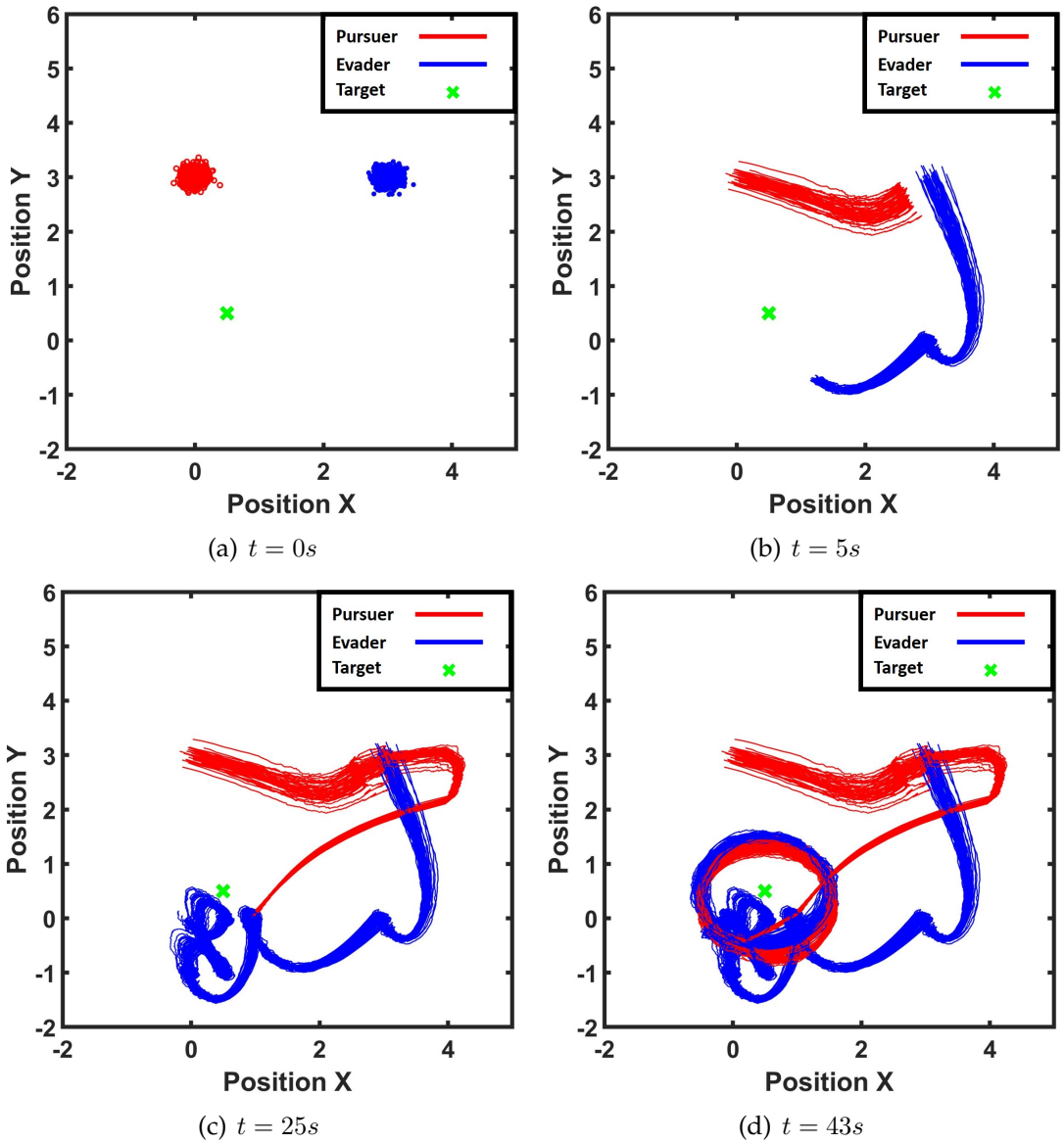



Figure 3.5: The distance between the pursuers and evaders. The red curves show the distances between all agents and the green curve shows the average distance.

and the mean positions for both groups are plotted in Figs. 3.4(a) and 3.4(b). We can see that the evader group is seized without reaching the target. Moreover, we plotted the distances between the pursuers and evaders in Fig. 3.5. Note that in this plot, we only plotted the distance $d_{ij} = \|x_{j,1} - x_{i,2}\|$ with $i = j$. Fig. 3.5 also confirms that the pursuers' group successfully intercepts the evaders' group. To further verify the interception performance, an additional experiment, where the evaders have a moving target, is designed. In this simulation, we set the target for all evaders as a time-varying function, i.e.,

$$x_d(t) = \begin{bmatrix} t \\ 2\sin(8t) + 2 \end{bmatrix}$$

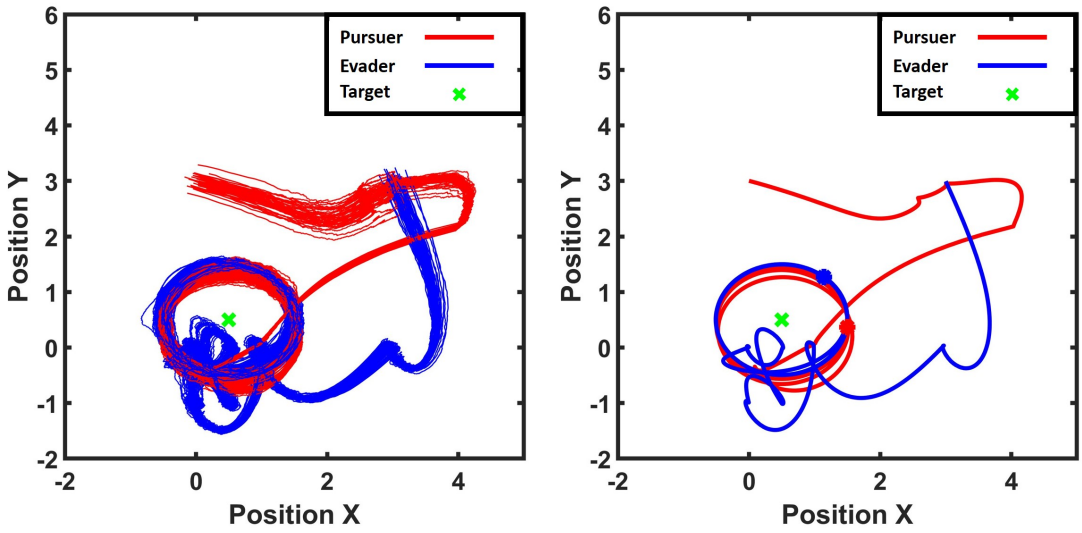
The resulting average trajectories of both evader group and pursuer group is plotted in Fig. 3.8, where the pursuers successfully capture the evaders at $2.8s$.




(a) $t = 0s$ (b) $t = 5s$

(c) $t = 25s$ (d) $t = 43s$

Figure 3.6: The trajectory of agents with respect to time. For the visibility of this plot, only 50 pursuers as well as 50 evaders are plotted (total 1000). The evaders' positions are plotted as the blue star while the pursuers' positions are plotted as the red circles. The trajectories of the pursuers and evaders are plotted as red and blue curves respectively. The green cross marks the target position for the evaders.




(a) 50 agents' trajectory at the end of simulation.

(b) The average trajectory of two groups at the end of simulation.

Figure 3.7: The plot of the agents' positions and trajectories at the end of the simulation. Pursuers' and evaders' trajectories are marked with red and blue curves, respectively. The left image shows the time evolution of the trajectories for 50 pursuers and 20 evaders. The right plot demonstrates the average trajectory of both pursuers' team and evaders' team.

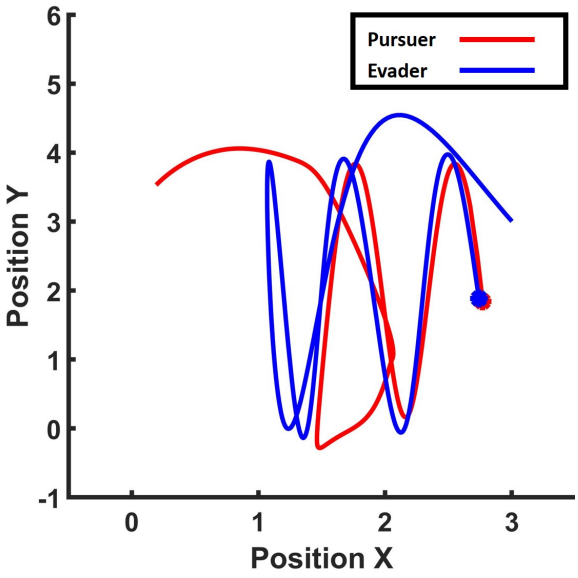
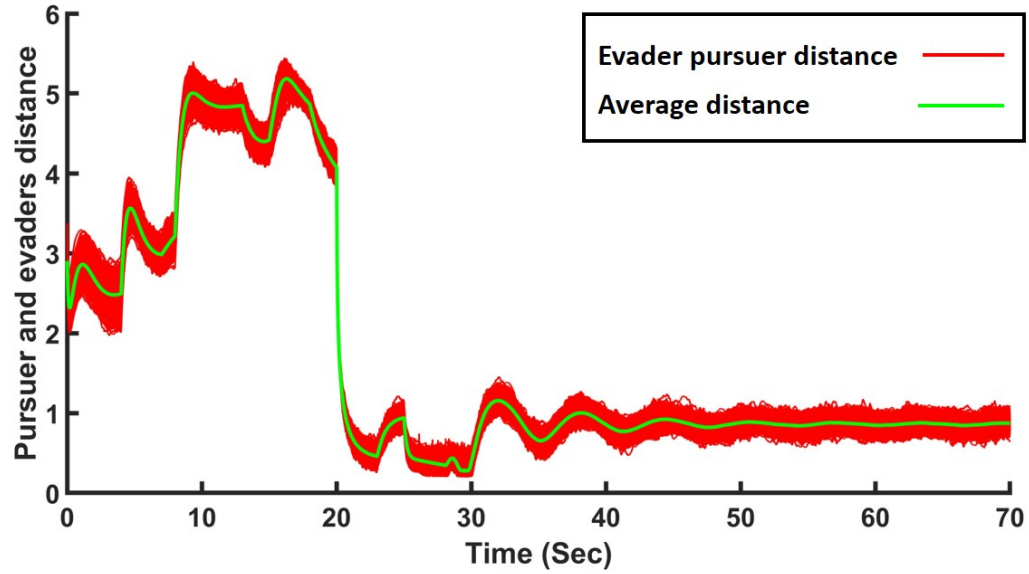



Figure 3.8: Average trajectory of pursuers and evaders when the evader follows a sine trajectory.

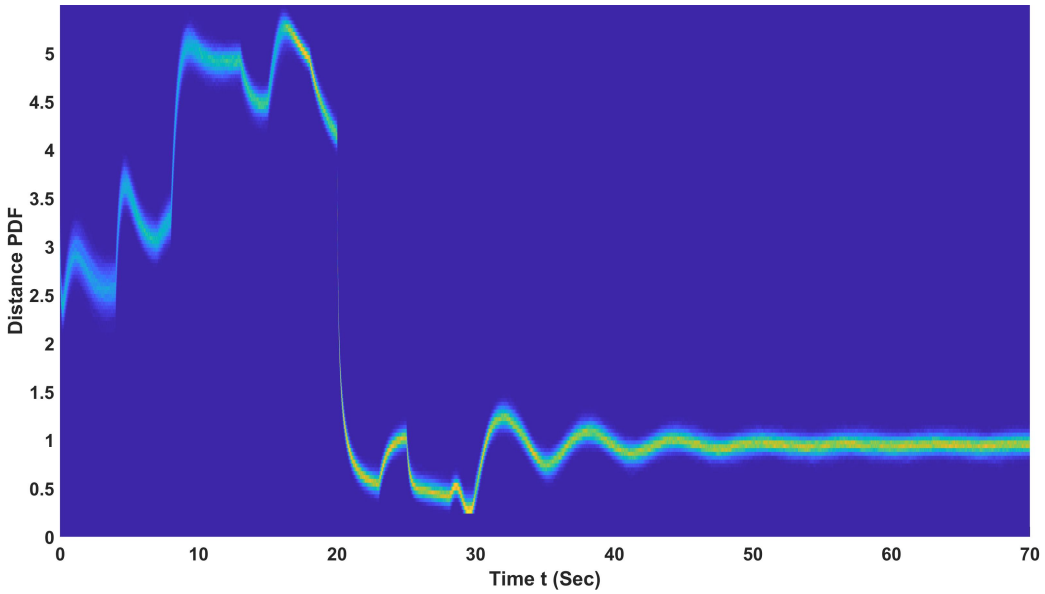
Next, the proposed ACMO algorithm is used by not only the pursuers but also the evaders. Similar to the above analysis, we first plot the time evolution of all agents in Fig. 3.6. With the same initial positions' distribution, the second experiment shows a great difference. Compared with Fig. 3.3 in the first experiment, the evaders are aware of the pursuers and start to escape from the formal optimal path. At 5s, the evader group makes a detour to avoid being captured by the pursuers directly. Meanwhile, the pursuers estimate the evaders' worst actions instead of simply following the evaders. At time 25s, the pursuers move directly toward the evader groups' possible locations. Finally, at 43s, the evaders and the pursuers reach an equilibrium near the target. The overall trajectories for all agents at the end of the simulation, i.e., the 70s, are plotted in Fig. 3.7(a). Furthermore, the average trajectories are demonstrated in Fig. 3.7(b). Fig. 3.7(b) confirms that all agents form an equilibrium from 43s to 70s. This equilibrium is known as the Nash Equilibrium, which is shown in the following plots.

To demonstrate more details, we also plot the distances between the pursuers and evaders in Fig. 3.9(a). Similar to Fig. 3.5, the individual distances are plotted using the light red curve, and the average distance is shown in the green curve. The results reveal that the pursuers cannot capture the evaders, and the evaders can neither reach the target position. An equilibrium is reached after the 60s when the pursers and evaders maintain the same distance. Meanwhile, the PDF of the distances is also given in Fig. 3.9(b) which confirms the equilibrium in another aspect. It is worth noting that the stochastic terms cause the variance of the distance after 60 seconds, i.e., $D_1 dw_{j,1}$ and $D_2 dw_{i,2}$. The variance goes to zero when the coefficient matrices $D_1$ and $D_2$ go to zero.

After the controller's effectiveness is proved, we will demonstrate the optimal-




(a) The distances between pursuers and evaders.




(b) The PDF plot of the distances

Figure 3.9: The distances between the pursuers and evaders. In the upper image, the red curves show the distances between all agents, and the green curve shows the average distance. In the lower figure, the tracking error with higher probability shows yellow color.

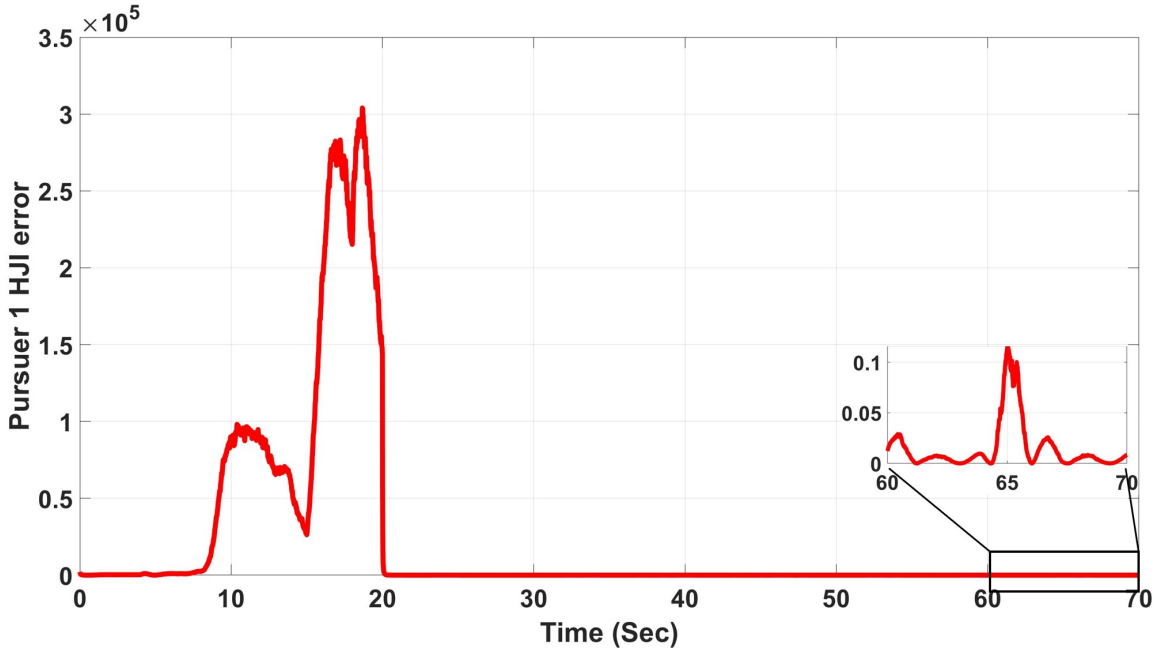



Figure 3.10: The error plot of pursuer 1's HJI equation (critic NN). The results at 60-70s are enlarged.

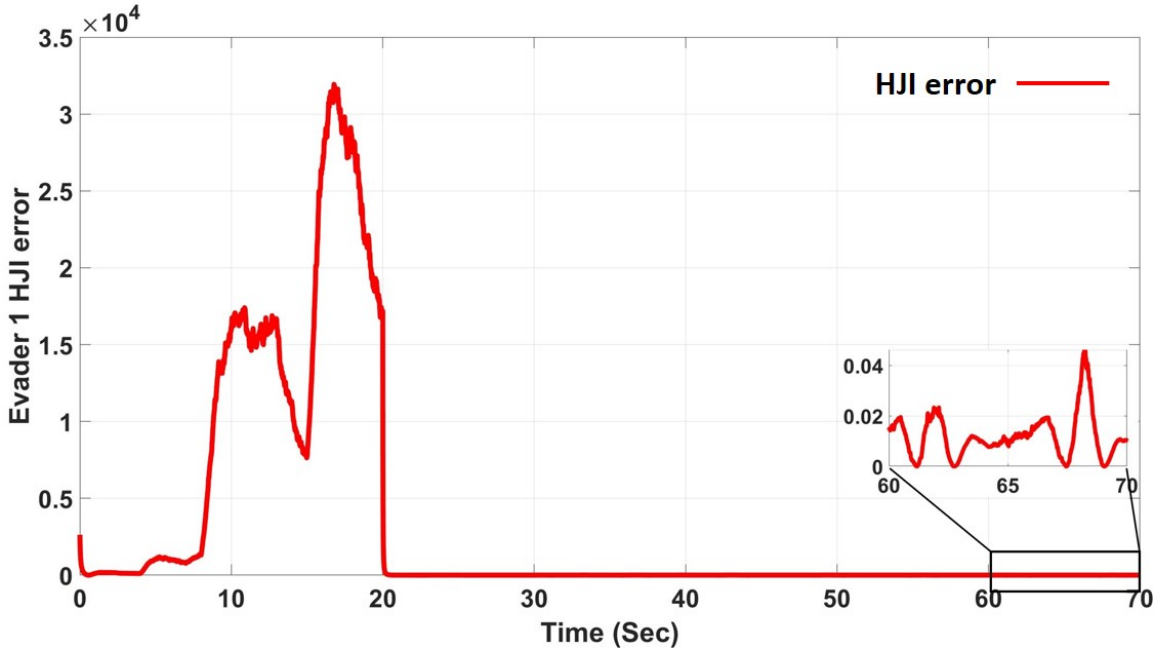



Figure 3.11: The error plot of evader 1's HJI equation (critic NN). The results at 60-70s are enlarged.

ity and the NNs' learning performance through the HJI equation error, i.e., critic NN's estimation error. To investigate more details, we show a single pursuer's and an individual evader's HJI equation error in Figs. 3.10 and 3.11. We can obviously see that both pursuer's and evader's HJI equation error is bounded near zero after the 60s. The HJI equation convergence confirms that the MFG equations are successfully estimated, and the Nash Equilibrium, which is considered as the optimal strategy for large scale multi-player pursuit-evasion game in this paper, is reached. Provided by the trajectories, distances, and the HJI equation's estimation error, the results demonstrate that the optimal strategy is reached for the very large scale multi-player pursuit-evasion game.

Finally, the developed ACMO algorithm's performance is compared with a traditional reinforcement learning method for multi-agent stochastic differential games [62]. In the comparison simulation, multiple pursuers and one evader are considered for simplicity. Therefore, the mass of the evaders' team is replaced by the position of the single evader. The dynamics of pursuers and the evader are selected as:

$$\text{Pursuers' dynamics } dx_{j,1} = \left( \begin{bmatrix} -3 & 2 \\ 1 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \right) dt + D_1 dw_{j,1}$$

$$\text{Evader's dynamics } dx_2 = \left( \begin{bmatrix} -3 & 5 \\ -1 & 1 \end{bmatrix} x + \begin{bmatrix} 2 \\ 1 \end{bmatrix} u \right) dt + D_2 dw_2$$

The evader follows the traditional optimal control strategy [52] to track a given target, i.e., $[0, 0]$ while the pursuers' controls are calculated using the ACMO algorithm and traditional reinforcement learning (RL) algorithm. The results are plotted in Figs. 3.12, 3.13. In Fig. 3.12, the average distance between pursuers and

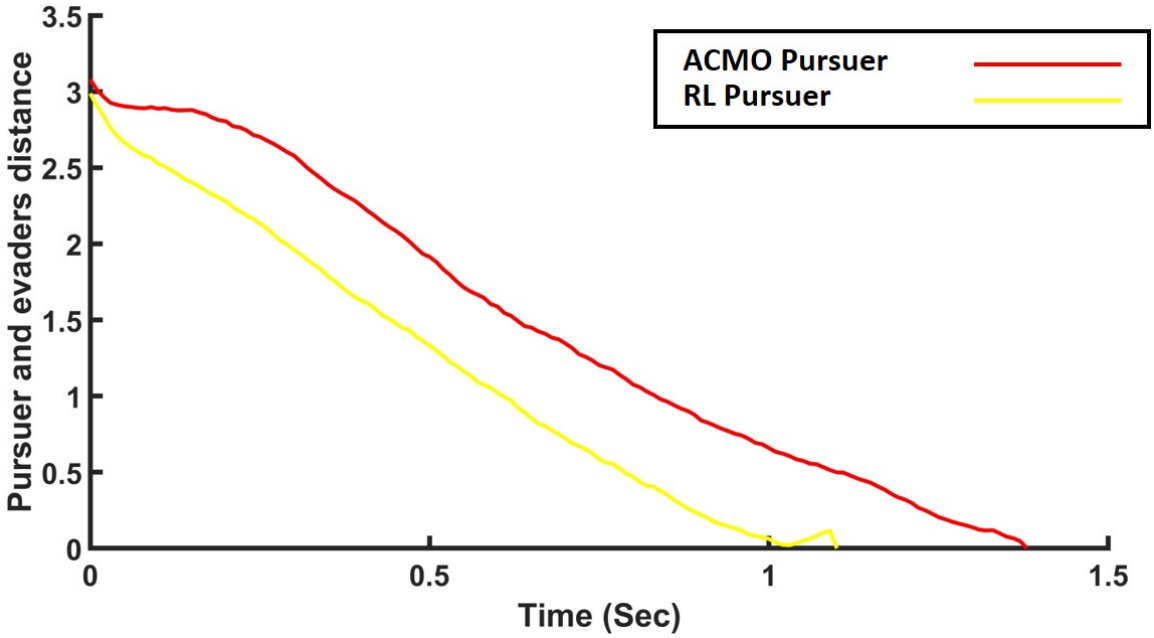



Figure 3.12: Average pursuer and evader distance. The red and yellow curves represent the average distance between pursers and the evader using the developed ACMO algorithm and the reinforcement learning (RL) algorithm [2] respectively.

the evader is plotted for both algorithms. The RL pursuers can capture the evader faster than the ACMO pursuers as illustrated in Fig. 3.12. The RL method developed in [62] is known as a centralized solution and guarantee the optima while the ACMO algorithm requires a longer time to tune the neural networks and approximate the $\epsilon_N$ optimal control, which is worse than the actual optimal control. However, despite the RL [62] method can capture the evader faster, the cost of communication between all agents cannot be ignored.

Next, the comprehensive cost, which contains the communication cost, is calculated to provide a complete comparison of the performance. Specifically, the comprehensive cost is defined as the summation of the running cost from the cost

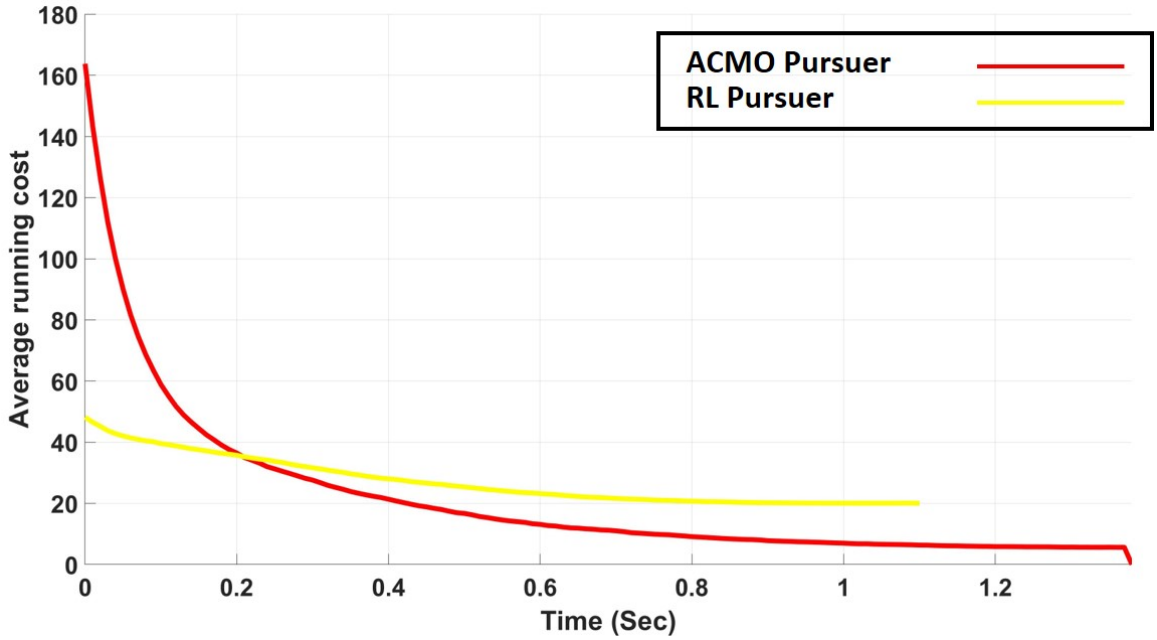



Figure 3.13: Average pursuer and evader running cost. The red and yellow curves represent the average running cost between pursers and the evader using the developed ACMO algorithm and the reinforcement learning (RL) algorithm [2] respectively

function and the communication cost, i.e.,

$$C_{j,1} = \|x_{j,1} - \bar{x}_1\|^2_{Q_1} + \|x_{j,1} - x_2\|^2_{Q_2} + \|u_{j,1}\|^2_{R1} + \alpha N_1$$

where $\alpha$ is the communication cost weight, $N_1$ is the agent number of the pursuers, and the term $\alpha N_1$ denotes the communication cost. The costs of both algorithms are plotted in 3.13.

It is evident in Fig. 3.13 that the initial cost of the RL pursuers are lower because they're executing the centralized optimal solution. The initial cost of the ACMO pursuers is higher because of the NNs' convergence speed. The neural networks (NNs) of the ACMO algorithm are more complicated than the RL algorithm due to the mass NN. And thus, the ACMO algorithm's convergence speed is slower than the RL algorithm, which results in non-optimal controls. However,

after 0.2s, the cost of the RL pursuers are higher than the ACMO pursuers. The main reason is that the approximated optimal control using ACMO is becoming closer to the actual optima. In contrast, the RL pursuers, which are already at optima, is penalized for high communication demands. In summary, the traditional reinforcement algorithms [102, 62] can find the optimal solution faster and more accurate. However, the developed ACMO algorithm is a decentralized algorithm that is especially suitable for tasks that have large scale number of agents, but no communications are allowed.

## 3.5 Conclusions

In this paper, a novel actor-critic-mass-opponent (ACMO) method has been developed to solve the pursuit-evasion game with large-scale pursuers and evaders in a decentralized fashion. The developed decentralized algorithm can effectively tackle the notorious *"Curse of Dimensionality"* challenge and unrealistic assumption of the real-time reliable communication network for extremely massive agents in the traditional distributed pursuers' as well as the evaders' control designs. The mean field game (MFG) is utilized such that the pursuer and evader groups are formulated into two stochastic virtual mass players with a time-varying probability density function (PDF) computed by solving the coupled HJI-FPK equation systems. To solve the coupled equations, four neural networks are employed, i.e., the actor neural network (NN) to learn optimal control, the critic NN to estimate the optimal cost function, the mass NN for estimating the own group's PDF, and the opponent NN for approximating the influence from the opponent group. Moreover, we analyze the closed-loop system's stability and convergence of the proposed NNs systematically through Lyapunov techniques. The simulation results

also demonstrated that the developed mean-field type of ACMO algorithm could effectively approximate the optimal solution.

CHAPTER 4

# LARGE-SCALE MULTI-AGENT REINFORCEMENT LEARNING WITH APPLICATIONS IN ELECTRICAL VEHICLE CHARGING [116]

## 4.1 Introduction

Electric vehicles (EVs) have gained increasing attention since EVs can reduce gas emission and increase the practical usage of renewable energy [63]. On the other hand, EVs are also capable of being potentially transformed into autonomous vehicles, which are one of the essential applications for future smart cities [82]. As the most common EVs applications, the electric buses have some distinct characteristics from other EVs. Firstly, most electric buses in the same city are owned by a handful of companies, which means the battery designs and models can be similar. Secondly, unlike the residential EVs, the electric buses have a stable routine, indicating a collective charging schedule. The power usage pattern often coincides with the electricity usage pattern for the residential load. With the more and more electrical buses employed in the city, the concurrent charging behavior during the evening, along with the surge of residential load, would inevitably cause severe supply imbalance to the power grid (see fig. 4.1). Therefore, an effective charge schedule system for a large scale of electric buses is urgently needed.

The time-of-use (TOU) price [104, 42], which has been successfully applied by many utility companies to adjust the users' energy consumption pattern [89], is introduced to coordinate the electric buses' charging schedule along with the residential load. In the TOU price pattern, the utility company raise the electricity price when the demand is high and reduce the rate while the need is low. Under such a scheme, the charging stations are encouraged to purchase electricity when

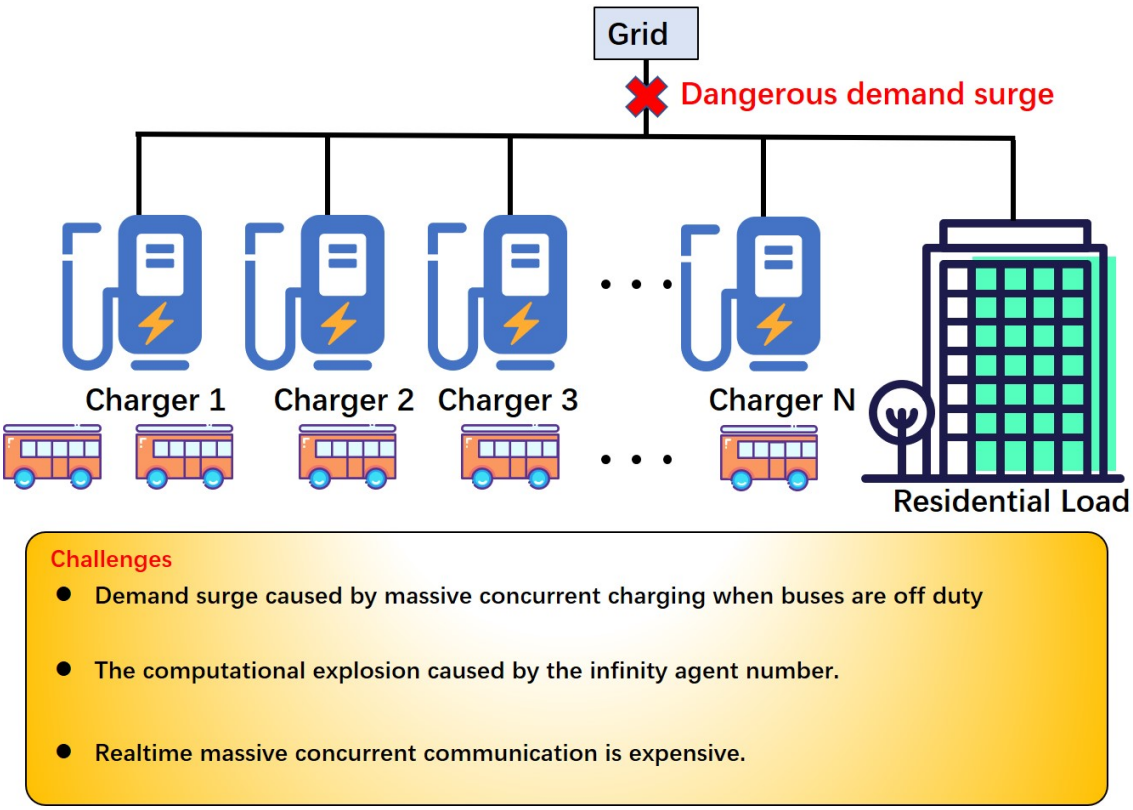



Figure 4.1: The problem of large scale electric buses charging problem.

the residential load demand is not significant. As such, the total power consumption becomes smooth. Due to the similarity to economic problems, game theory has been used broadly to solve the optimal TOU strategy, such as dynamic price adjustment [105, 46]. By integrating electric buses charging into the TOU games effectively, the charging schedule can automatically adapt to the dynamically changing residential power consumption. When considering the electric buses' total charging consumption and the residential load demand in the energy market, each bus needs to find the optimal charging strategy such that its electricity price can be minimized. This price competition exists not only between the buses and the residential load but also inside the electric buses group. For example, at midnight, the residential load demand is low, which results in lower TOU price. However,

if all electric buses are charging simultaneously, the concurrent power demand would quickly inflate the TOU price. Hence, it is necessary to design a charging control based on game theory to coordinate the charging schedule. In previous researches, the charging problem is formulated into a large scale **multi-agent system (MAS)** optimization problem. For instance, [44, 56] proposed centralized charging controller to compute optimal charging schedule for all agents at a centralized dispatch center. [75] studied the problem under a distributed system. These approaches mainly suffer from two stringent requirements, i.e., 1) The increment of vehicles' number drastically increases the computational complexity while solving the decentralized optimal charging control, which is also known as the notorious *"Curse of Dimensionality"*; 2) The limited communication resource cannot ensure the reliable data exchange among a large number of electric buses during charging [19, 112, 3]. To overcome these difficulties, the Mean Field Game (MFG) theory [34, 48] can be adopted and further modified to apply to address the very large scale of electric buses decentralized optimal charging problem.

Compared to traditional multi-agent systems optimization algorithms such as [88, 26, 101] where the states of all the agents are required in the objective function, the MFG theory tackles the *"Curse of Dimensionality"* by replacing high-dimension augmented states from all the agents with a fix-dimension Probability Density Function (PDF). Specifically, while applying the mean field game theory, each electric bus will encode the state-of-charge (SOC) and the charging rate of all other electric buses into a form of probability density function (PDF) to avoid the dimension explosion even when the total number of electric buses is enormous. More importantly, the new form of PDF does not need to be acquired through data communication. It can be obtained by solving a new type of partial differential equation (PDE) named the Fokker-Planck-Kolmogorov (FPK) equation [34]

with local information only. In the large scale of the electric bus charging process, we assume that all buses have homogeneous charging dynamics because they are often purchased in bulk (e.g., New York City [1]). Thus, every individual electric bus can estimate the group charging rate by substituting the SOC of all other electric buses, representing in the form of PDF, into corresponding policy. As a result, the estimated group charging rates can be utilized to update the time-varying PDF via the FPK equation. Since all electric buses' SOC distribution are calculated locally (except the initial distribution [34]), the communication burden can be released significantly. In our previous work [120], the general mean field game theory was applied to form a decentralized online algorithm to optimize the tracking performance of large scale multi-agent systems. Similar approaches that applied the mean field games into the electric vehicle charging problem can be found in [18, 121]. In [121], the authors used the mean field game theory to reduce the charging time but did not consider the TOU price. In [18], a similar technique is discussed, but the solution is off-line, which is not practical in realistic applications. None of these works develop a specific algorithm that can online solve the generalized mean field game-based charging problem for large-scale electric buses. The mean field game features two coupled PDEs that need to be solved simultaneously, i.e., the backward HJB equation and the forward FPK equation. Due to those two coupled PDEs' nonlinearity, it is nearly impossible to obtain the exact analytical solution [119, 35]. Therefore, we proposed the Actor-Critic-Mass (ACM) algorithm along with adaptive learning methods [51, 52] to approximate the optimal charging control by learning the solution of two coupled PDEs online.

In this chapter, the very large scale electric bus charging problem has been studied considering the TOU price. Specifically, an algorithm named the Actor-Critic-Mass (ACM) has been proposed to employ three neural networks to approximate

the solution to the HJB and FPK equations numerically.

The contribution of this chapter can be summarized as:

1. The very large scale of the electric bus charging problem has been formulated as a mean field game to obtain an optimal solution considering the time-of-use (TOU) problem in **decentralized** fashion.
2. A novel online Actor-Critic-Mass (ACM) algorithm for a very large scale of electric bus charging control has been proposed to break the notorious *"Curse of Dimensionality"* and release the communication burden through integrating the Mean Field Games (MFG) theory with adaptive learning techniques.

The structure of this chapter is given as follows. Section II provides the background as well as a very large scale of electric vehicle charging control problem formulation. In Section III, the Mean Field Game formulation and the mean field equilibrium are given, and further, the Actor-Critic-Mass algorithm and adaptive learning are developed. Then, the numerical simulation is shown in Section IV to demonstrate the effectiveness of the proposed design.

## 4.2 Background and Problem Formulation

### 4.2.1 Single Electric Bus Charging Control Model

Consider a finite set $\mathcal{V} = \{1, 2, \cdots, N\}$ that represents the set of all electric buses in the city. With the assumption that the daily routine of those electric buses are

predefined and their operating power are similar since all those buses are manufactured under the same standard, the power consumption per unit time in an individual electric bus can be defined as a time-dependent deterministic function $c(t)$ that is homogeneous. Moreover, the electric buses need to be charged at the charge stations by considering the time-of-use (TOU) price. Let $s_i(t) \in [0, i]$ denotes the state of charge (SOC) at time $t$, $s_i = 0$ represents the battery being completely empty and $s_i = 1$ indicates fully charged. Next, each charger's power delivery rate, which is its output power, is represented as a function with respect to the SOC, i.e., $\alpha(s_i(t))$. The actual power acceptance rate of the bus, which denotes the received power, is $\eta\alpha(s_i(t))$ with $\eta \in (0, 1)$ being a discount factor. Then, the following stochastic differential equation for SOC can be obtained as

$$\dot{s}_i(t) = \eta\alpha_i(s_i(t)) - c(t) \tag{4.1}$$

where $i$ is the index for an individual electric bus. Because all electric buses have predefined fixed routes, we admit the electric buses' power consumption $c(t)$ periodic. Thus, similar to Couillet et al. [18], we study the charging schedule in a statistical sense instead of a single period since the power consumption can be somewhat different at every period and the charging process $\alpha_i$ in a single period is discrete if the period is long (i.e., three days in [18]). In terms of game theory, the charging process is modeled as a repeated game, and $c(t)$, $\alpha_i$ are averaged over infinite trials in a fixed period, i.e., $t \in [0, T]$. Note that such a definition is empirical and has statistical significance. In practical applications, $c(t)$ can be obtained through a relatively large dataset, and $\alpha_i$ provides necessary guidelines on when to charge. Suggested by [18], we shall now call $\alpha_i$ the provisioning rate, which denotes the purchase of electricity in unit time in the statistical sense.

Next, the remaining power level in a battery can be defined in terms of SOC as,

$$x_i(t) = 1 - s_i(t), \quad x_i(t) \in [0, 1], i \in (0, N] \tag{4.2}$$

Substituting (4.2) into the SOC dynamics (4.1), one obtains the evolution for the remaining power level as,

$$\begin{aligned} \dot{x}_i(t) &= -\dot{s}_i(t) \\ &= -\eta\alpha_i(s_i(t)) + c(t) \\ &= -\eta\alpha_i(1 - x_i(t)) + c(t) \\ &= -\eta\beta_i(t) + c(t) \end{aligned} \tag{4.3}$$

with $i$ being the index of individual electric bus.

To model the degradation of batteries that can affect the provisioning rate, a random Brownian noise independent of the SOC is included in (4.3) and yields

$$dx_i(t) = [-\eta\beta_i(t) + c(t)]dt + \sigma dW_i \tag{4.4}$$

where $W_i$ denotes the independent Brownian noise [18], and $\sigma$ is a coefficient constant for the random noise. The function $\beta_i(t)$ is directly associated with the battery's provisioning rate, and thus we will refer it as provisioning rate in this chapter for simplicity. Each electric bus aims to let $x_i(t)$ goes to zero. Without proper coordination, all electric buses will inevitably raise the power acceptance rate $\alpha_i(t)$ simultaneously that will lead to the increment of TOU. Therefore, a game-theoretic formulation is needed.

### 4.2.2 Game Formulation with Time-Of-Use and Prices

Let $\mathcal{X} = x_1, \cdots, x_N$ be the set of all electric buses remaining power level, $\Omega = \beta_1, \cdots, \beta_N$ denote the set of the provisioning rates of electric buses. Then, the cost to each electric bus includes three parts, i.e.,

1. The time-of-use (TOU) price is the cost of buying electricity at different times. The utility companies use this dynamic pricing model to adjust customers' power consumption. In this chapter, we model the TOU price in terms of the provisioning rate of electric buses $\Omega$, and the power consumed by residential areas $z(t)$, i.e., $p_i(\Omega, z(t))$.
2. The cost of current charge rate, i.e., $R\beta_i^2(t)$. This cost is defined to penalize the high charge power which can cause battery degradation.
3. The cost of current SOC, i.e., $Qx_i^2(t)$. This cost encourages the bus to charge their batteries.
4. The cost of SOC at terminal time, i.e., $\gamma(x_i(T))$. This terminal cost penalizes the charging strategy if the battery is still low at the terminal time.

Combining those factors yields the following cost function:

$$V_i(x_i, \Omega, \beta_i) = \mathbb{E}\left\{\int_0^T \left[p_i(\Omega, z) + R\beta_i^2 + Qx_i^2\right] dt + \gamma(x_i(T))\right\} \tag{4.5}$$

where $Q > 0, R > 0$ are constant weighting coefficients.

Consider the above cost function, the optimization of the total $N$ players, i.e. electric buses, at the charging station can be formulated as a non-cooperative game. It is obvious that each electric bus tries to minimize the cost function given in (4.5) by computing a dynamic routine of the provisioning rate $\beta_i$. Upon the non-cooperative game formulation, the Nash Equilibrium (NE) is introduced as the

optimal strategy set for all players/electric buses. Let $\Omega_f$ denote the set of all feasible provisioning rates and defining a mapping $\mathbb{F}_i : (\mathcal{X}, \Omega) \rightarrow \beta_i$ to represent the choice of provisioning rate for player/electric bus $i$ while considering the information from all the players/electric buses. We can further denote $\Omega^{(i)} = \Omega \setminus \beta_i$ as the set of actions other than player $i$'s action.

**Definition 1.** *(Nash Equilibrium (NE))* Given any SOC set $\mathcal{X}$ and provisioning rate set $\Omega$ at any given time $t$, the Nash Equilibrium (NE) of the $N$ players non-cooperative game is the strategy set $\Omega* = \{\beta_1^*, \cdots, \beta_N^*\}$ that generated by $\mathbb{F}_i : (\mathcal{X}, \Omega)$, and satisfies the following condition for all the players

$$V_i(x_i, \Omega^{*(-i)}, \beta_i^*) \leq V_i(x_i, \Omega^{*(-i)}, \beta_i), \quad \forall \beta_i \in \Omega_f$$

Definition 1 states that the NE action set $\Omega^*$ are the optimal strategy where all players have no intent to change their individual policies in order to obtain higher individual costs. Recall differential games [93], the optimal control can be solved backwards through the Bellman Principle of Optimality [52].

In addition, the optimal cost function for each player can be represented as

$$V_i^*(x_i, \Omega^{*(-i)}, \beta_i^*) = \min_{\beta_i} \mathbb{E} \left\{ \begin{array}{c} p_i(\Omega, z) + R\beta_i^2 + Qx_i^2 \\ +V_i^*(x_i, \Omega^{*(-i)}, \beta_i^*)\Big|_{t=t+\Delta t} \end{array} \right\} \tag{4.6}$$

where $\Delta t \rightarrow 0$ is an infinitesimal time duration.

Next, the corresponding optimal control can be derived as

$$\beta_i^* = \underset{\beta_i}{\text{argmin}} \mathbb{E} \left\{ \begin{array}{c} p_i(\Omega, z) + R\beta_i^2 + Qx_i^2 \\ +V_i^*(x_i, \Omega^{*(-i)}, \beta_i^*)\Big|_{t=t+\Delta t} \end{array} \right\} \tag{4.7}$$

In the optimal control policy (4.7), all players' information is required which causes two major challenges when the number of players, i.e. $N$, goes to infinity.

The first one is the difficulty of sharing real-time data in a communication network with very large number of players. And the second one is the drastically increased dimension of (4.7) which will cause severe computational complexity explosion. To solve the potential difficulties, the Mean Field Games (MFG) theory is engaged.

## 4.3 The ACM Algorithm Based On MFG

### 4.3.1 Mean Field Games Formulation

Mean Field Game (MFG) theory [34] is an emerging technique that can effectively solve stochastic decision-making problems with a large population of players in a decentralized manner. To formulate the mean field game, the following assumptions are needed: 1) the electric buses group is sufficiently large, i.e., $N \to \infty$, and 2) The charging dynamics of electric buses 4.4 are homogeneous. The second assumption needs that the buses have a similar discount factor $\eta$ and consumption rate $c(t)$. Such an assumption is reasonable since all the electric buses are manufactured under a unique standard.

The challenges of a large scale of electric buses charging problem is to include control inputs from all the electric buses into the optimal cost function (4.5), which can lead to the computation explosion and communication congestion as the number of electric buses goes to large and even close to infinity. In the MFG, the states from all the players need to be encoded into a multi-variant PDF $m(x, t)$ whose dimension is identical to the state space for breaking the notorious *"Curse of Dimensionality"* and further solving computation explosion as well as communication congestion problem. Intuitively, the dimension is reduced by transforming

the multi-player game into a two players' game, i.e., the individual bus and the entire team [13]. More importantly, such PDF can be computed by a partial differential equation (PDE) named the Fokker-Planck-Kolmogorov (FPK) equation with local information only [34, 20].

Next, the TOU price, in the context of mean field games, can be derived as the function with respect to the PDF $m(x,t)$ by substituting the provisioning rate dynamics (4.3) as

$$\begin{aligned} p(m,z) &= \sum_{\beta_i \in \Omega_f} \beta_i + z(t) \\ &= \frac{-N\frac{d}{dt}\mathbb{E}_x\{m(x,t)\} + c(t)}{\eta} + z(t) \end{aligned} \tag{4.8}$$

Then, substituting the TOU price (4.8) into the coupling term in (4.5), one obtains the Mean Field coupling cost function as

$$V_i(x_i, m, \beta_i) = \mathbb{E}\left\{\int_0^T \left[p_i(m,z) + R\beta_i^2 + Qx_i^2\right]dt + \gamma(x_i(T))\right\} \tag{4.9}$$

Note that in (4.9), the provisioning rate information from all electric buses has been replaced by the PDF $m(x,t)$.

Recall optimal control theory [51], the optimal cost function can be obtained through the Hamilton-Jacobi-Bellman (HJB) equation, i.e.,

$$\begin{aligned} &-\frac{\partial}{\partial t}V_i^*(x_i, m, \beta_i^*) - \frac{\sigma^2}{2}\frac{\partial^2}{\partial t^2}V_i^*(x_i, m, \beta_i^*) + R\beta_i^2 + Qx_i^2 \\ &+ \left[-\eta\beta_i^*(t) + c(t) + 1\right]\frac{\partial}{\partial x_i}V_i^*(x_i, m, \beta_i) = p(m,z) \end{aligned} \tag{4.10}$$

Solving the optimal charging strategy, in the mean field form, is to approximates the solution to the HJB equation (4.6) and also gives the optimal provision-

ing rate explicitly [34], i.e.,

$$\beta_i^* = \frac{-\eta}{2R}\frac{\partial}{\partial x_i}V_i^*(x_i, m, \beta_i^*) \tag{4.11}$$

To solve the HJB equation (4.10) and optimal provisioning rate (4.11), the PDF $m(x,t)$ is needed. Recall the MFG [48], the PDF can be attained by solving the Fokker-Planck-Kolmogorov (FPK) equation based on the *"law of large numbers"*, i.e.,

$$\frac{\partial}{\partial t}m(x,t) = -\frac{\partial}{\partial x}\left[m(x,t)\frac{-\eta}{2R}\frac{\partial}{\partial x_i}V_i^*(x_i, m, \beta_i^*)\right] + \frac{\sigma^2}{2}\frac{\partial^2}{\partial x^2}m(x,t) \tag{4.12}$$

Note that in (4.12), the required information can be obtained locally. Under the assumption that the charging dynamics of electric buses are homogeneous, the time evolution of the PDF is able to be approximated through the local optimal provisioning rate. As such, the large scale of electric buses charging control is able to follow a decentralized manner which greatly relieved the communication burden.

At this point, it is clear that the optimal cost function $V_i^*(x_i, m, \beta_i^*)$ as well as the state PDF $m(x,t)$ has formed a solution pair that is subjecting to both HJB (4.10) and FPK (4.12) equations. Upon the approximation of HJB equation, the optimal provisioning rate yields a near Nash Equilibrium [13], i.e.,

**Definition 2.** *($\epsilon_N$ Nash Equilibrium)* Given any SOC set $\mathcal{X}$ and provisioning rate set $\Omega$ at any given time $t$, the Nash Equilibrium (NE) of the $N$ players non-cooperative game is a strategy set $\Omega* = \{\beta_1^*, \cdots, \beta_N^*\}$ that generated by $\mathbb{F}_i : (\mathcal{X}, \Omega)$, and satisfies the following condition for all agents

$$V_i(x_i, \Omega^{*(-i)}, \beta_i^*) \leq V_i(x_i, \Omega^{*(-i)}, \beta_i) + \epsilon_N, \quad \forall \beta_i \in \Omega_f$$

And the solution PDE pair has the following property,

**Theorem 11.** *(Solution of the coupled HJB-FPK PDEs)* [48] The solution of the coupled HJB-FPK PDEs exists and is unique. The generated optimal control strategy is an $\epsilon_N$ Nash Equilibrium such that $\lim_{N\to\infty} \epsilon_N = 0$.

The proof can be found in numerous mean field games literature such as [34, 48, 70].

Different than conventional distributed control for EVs charging that needs accurate information from all the EVs, the MFG-based method has shown that the performance of individual EV relies on the combination of local information and influence from the entire team of EVs. To describe the influence, the PDF of massive states from all the EVs $m(x,t)$ has been introduced. It is important to note that the PDF can be computed without knowing the other EVs' real-time information. To obtain the optimal provisioning rate, the coupled HJB-FPK PDEs need to be solved simultaneously at each EV. However, the HJB equation (4.10) needs to be solved backward-in-time [52] whereas the FPK equation (4.12) is solved forward-in-time. This makes the Mean Field Type of control difficult to solve analytically. Therefore, in this chapter, a novel reinforcement learning and approximate dynamic programming (ADP) technique has been developed to learn the coupled HJB-FPK PDEs solution online and simultaneously.

### 4.3.2 Actor-Critic-Mass (ACM) based Adaptive Learning Algorithm

In this section, the proposed Actor-Critic-Mass (ACM) adaptive learning algorithm is developed in detail. The structure of the ACM based adaptive learning algorithm is shown in Fig. 4.2. In the proposed structure, each electrical bus maintains three neural networks, i.e., the actor neural network to approximate the op-

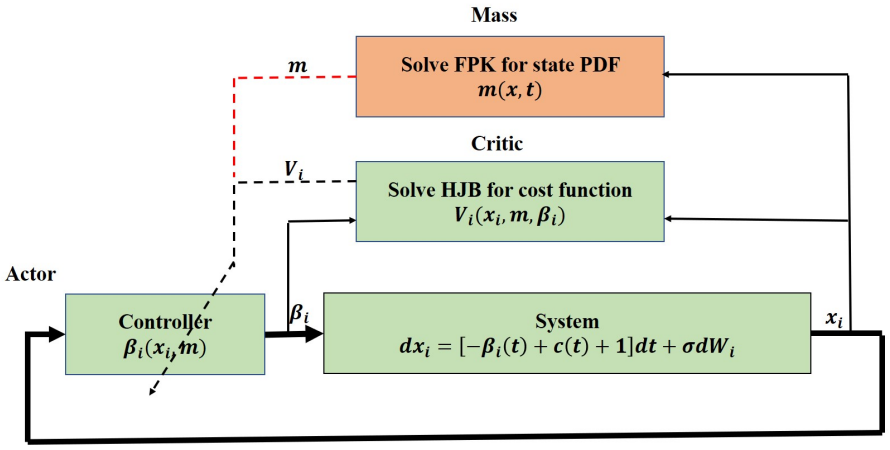



Figure 4.2: An illustration of the proposed ACM based adaptive learning structure. The actor, critic, and mass neural network estimated the solution to the optimal control (i.e., provisioning rate), optimal cost function, and the SOC PDF of all agents.

timal control (i.e., provisioning rate), the critic neural network to approximate the optimal cost function, and the mass neural network to approximate the SOC PDF.

With the mild assumptions that there exists constant neural network weights $W_{V,i}$, $W_{u,i}$, $W_{m,i}$ such that the optimal provisioning rate, optimal cost function, the SOC PDF for electric bus $i$ can be represented as

$$\begin{aligned} &\text{Critic:} \quad V_i^*(x_i, m) = W_{V,i}^T \phi_{V,i}(x_i, m) + \varepsilon_V \\ &\text{Actor:} \quad \beta_i^*(x_i, m) = W_{u,i}^T \phi_{u,i}(x_i, m) + \varepsilon_u \\ &\text{Mass:} \quad m(x_i, t) = \hat{W}_{m,i}^T \phi_{m,i}(x_i, t) + \varepsilon_m \end{aligned} \tag{4.13}$$

where the functions $\phi_{V,i}(x_i, m)$, $\phi_{u,i}(x_i, m)$, and $\phi_m(x_i, t)$ are bounded and continuous activation functions, $\varepsilon_V$, $\varepsilon_u$, and $\varepsilon_m$ are the reconstruction errors of the neural networks (NNs). The NN weights $W_{V,i}$, $W_{u,i}$, $W_{m,i}$ are unknown and expected to be solved. Let the the approximated weights be denoted as $\hat{W}_{V,i}$, $\hat{W}_{u,i}$, $\hat{W}_{m,i}$, the optimal provisioning rate, optimal cost function, and SOC PDF can be estimated

as

$$\begin{aligned} &\text{Critic:} \quad \hat{V}_i(x_i, m) = \hat{W}_{V,i}^T(t)\phi_{V,i}(x_i, \hat{m}_i) \\ &\text{Actor:} \quad \hat{\beta}_i(x_i, m) = \hat{W}_{u,i}^T(t)\phi_{u,i}(x_i, \hat{m}_i) \\ &\text{Mass:} \quad \hat{m}_i(x_i, t) = \hat{W}_{m,i}^T(t)\phi_{m,i}(x_i, t) \end{aligned} \tag{4.14}$$

Substituting the (4.14) into the mean field equations, i.e. (4.10) and (4.12), they will not hold. The approximated weights are then tuned by the residual errors, i.e.,

$$e_{HJB,i} = p(\hat{m}_i, z) + \hat{W}_{V,i}^T(t) \begin{bmatrix} \frac{\partial}{\partial t}\phi_{V,i}(x_i, \hat{m}_i) + \frac{\sigma^2}{2}\frac{\partial^2}{\partial t^2}\phi_{V,i}(x_i, \hat{m}_i) \\ \left(-\eta\hat{\beta}_i(t) + c(t) + 1\right)\frac{\partial}{\partial x_i}\phi_{V,i}(x_i, \hat{m}_i) \end{bmatrix} \tag{4.15}$$

$$e_{FPK,i} = \hat{W}_{m,i}^T(t) \begin{bmatrix} -\frac{\partial}{\partial t}\phi_{m,i}(x_i, t) + \frac{\sigma^2}{2}\frac{\partial^2}{\partial x^2}\phi_{m,i}(x_i, t) \\ -\frac{\partial}{\partial x}\left(\phi_{m,i}(x_i, t)\frac{-\eta}{2R}\frac{\partial}{\partial x_i}V_i(x_i, \hat{m}_i)\right) \end{bmatrix} \tag{4.16}$$

$$e_{u,i} = \hat{W}_{u,i}^T(t)\phi_{u,i}(x_i, \hat{m}_i) + \frac{\eta}{2R}\frac{\partial}{\partial x_i}V_i(x_i, \hat{m}_i) \tag{4.17}$$

Next, let us denote

$$\Psi_{V,i}(x_i, \hat{m}_i) = \begin{bmatrix} \frac{\partial}{\partial t}\phi_{V,i}(x_i, \hat{m}_i) + \frac{\sigma^2}{2}\frac{\partial^2}{\partial t^2}\phi_{V,i}(x_i, \hat{m}_i) \\ \left(-\eta\hat{\beta}_i(t) + c(t) + 1\right)\frac{\partial}{\partial x_i}\phi_{V,i}(x_i, \hat{m}_i) \end{bmatrix}$$

$$\Psi_{m,i}(x_i, \hat{V}_i) = \begin{bmatrix} -\frac{\partial}{\partial t}\phi_{m,i}(x_i, t) + \frac{\sigma^2}{2}\frac{\partial^2}{\partial x^2}\phi_{m,i}(x_i, t) \\ -\frac{\partial}{\partial x}\left(\phi_{m,i}(x_i, t)\frac{-\eta}{2R}\frac{\partial}{\partial x_i}V_i(x_i, \hat{m}_i)\right) \end{bmatrix}$$

$$\tilde{p}(\tilde{m}_i, z) = p(\hat{m}_i, z) - p(m, z)$$

The estimation errors (4.15) and (4.16) can be represented as:

$$e_{HJB,i} = p(\tilde{m}_i, z) + p(m, z) + \hat{W}_{V,i}^T(t)\Psi_{V,i}(x_i, \hat{m}_i) \tag{4.18}$$

$$e_{FPK,i} = \hat{W}_{m,i}^T(t)\Psi_{m,i}(x_i, \hat{V}_i) \tag{4.19}$$

Then, we will consider the effect of the reconstruction errors to the approximation error and further derive their relationships. By substituting (4.13) into the mean field equations (4.10) and (4.12), one obtains

$$p(m,z) + \hat{W}_{V,i}^T(t)\Psi_{V,i}(x_i,\hat{m}_i) + \varepsilon_{HJB,i} = 0 \tag{4.20}$$

$$W_{m,i}^T(t)\Psi_{m,i}(x_i,V_i) + \varepsilon_{FPK,i} = 0 \tag{4.21}$$

where the errors $\varepsilon_{HJB,i}$ and $\varepsilon_{FPK,i}$ are caused by the reconstruction errors. They will converge to zeros when the reconstruction errors go to zero.

Substituting (4.20) into (4.18), and (4.21) into (4.19), we derive

$$p(\tilde{m}_i,z) - \tilde{W}_{V,i}^T\Psi_{V,i}(x_i,\hat{m}_i) - W_{V,i}^T\tilde{\Psi}_{V,i}(x_i,\tilde{m}_i) - \varepsilon_{HJBi} = e_{HJBi} \tag{4.22}$$

$$-\tilde{W}_{m,i}^T\Psi_{m,i}(x_i,\hat{V}_i) - W_{m,i}^T\tilde{\Psi}_{m,i}(x_i,\tilde{V}_i) - \varepsilon_{FPKi} = e_{FPKi} \tag{4.23}$$

Similarly, we can obtain

$$-\tilde{W}_{u,i}^T\phi_{u,i}(x_i,\hat{m}_i) - W_{u,i}^T\tilde{\phi}_{u,i}(x_i,\tilde{m}_i) - \frac{\eta}{2R}\frac{\partial}{\partial x_i}V_i(x_i,\hat{m}_i)\partial_x\tilde{V}_i - \varepsilon_{ui} = e_{ui} \tag{4.24}$$

with

$$\begin{aligned}
&\varepsilon_{ui} = \varepsilon_{mi} + \frac{1}{2}R^{-1}g^T(x_i)\partial_x\varepsilon_{HJBi}\\
&\tilde{W}_{V,i} = W_{V,i} - \hat{W}_{V,i}(t)\\
&\tilde{W}_{m,i} = W_{m,i} - \hat{W}_{m,i}(t)\\
&\tilde{W}_{u,i} = W_{u,i} - \hat{W}_{u,i}(t)\\
&\tilde{\Psi}_{V,i}(x_i,\tilde{m}_i) = \Psi_{V,i}(x_i,m,t) - \Psi_{V,i}(x_i,\hat{m}_i)\\
&\tilde{\Psi}_{m,i}(x_i,\tilde{V}_i) = \Psi_{m,i}(x_i,m,V_i,t) - \Psi_{m,i}(x_i,\hat{V}_i)\\
&\tilde{\phi}_{u,i}(x_i,\tilde{m}_i) = \phi_{u,i}(x_i,m,t) - \phi_{u,i}(x_i,\hat{m}_i)
\end{aligned}$$

According to the gradient descent algorithm [91], the NN update law for the ACM can be derived as

$$\text{Critic NN: } \dot{\hat{W}}_{V,i}(t) = -\alpha_{h,i}\frac{\Psi_{V,i}(x_i,\hat{m}_i)e_{HJBi}^T}{1+\Psi_{V,i}^T(x_i,\hat{m}_i)\Psi_{V,i}(x_i,\hat{m}_i)} \tag{4.25}$$

$$\text{Mass NN: } \dot{\hat{W}}_{m,i}(t) = -\alpha_{m,i}\frac{\Psi_{m,i}(x_i, \hat{V}_i)e^T_{FPKi}}{1 + \Psi_{m,i}(x_i, \hat{V}_i)\Psi_{m,i}(x_i, \hat{V}_i)} \tag{4.26}$$

$$\text{Actor NN: } \dot{\hat{W}}_{u,i}(t) = -\alpha_{u,i}\frac{\phi_{u,i}(x_i, \hat{m}_i)e^T_{ui}}{1 + \phi^T_{u,i}(x_i, \hat{m}_i)\phi_{u,i}(x_i, \hat{m}_i)} \tag{4.27}$$

with $\alpha_{h,i}$, $\alpha_{m,i}$, and $\alpha_{u,i}$ being the learning rates.

### 4.3.3 The Performance Analysis of ACM-based Adaptive Learning Algorithm

According to the NN weight update laws, i.e. (4.25), (4.27), and (4.26), we obtain the first derivatives of the NN estimation errors as

$$\text{Critic NN: } \dot{\tilde{W}}_{V,i}(t) = -\dot{\hat{W}}_{V,i}(t) = \alpha_{h,i}\frac{\Psi_{V,i}(x_i, \hat{m}_i)e^T_{HJBi}}{1 + \Psi^T_{V,i}(x_i, \hat{m}_i)\Psi_{V,i}(x_i, \hat{m}_i)} \tag{4.28}$$

$$\text{Mass NN: } \dot{\tilde{W}}_{m,i}(t) = -\dot{\hat{W}}_{m,i}(t) = \alpha_{m,i}\frac{\Psi_{m,i}(x_i, \hat{V}_i)e^T_{FPKi}}{1 + \Psi_{m,i}(x_i, \hat{V}_i)\Psi_{m,i}(x_i, \hat{V}_i)} \tag{4.29}$$

$$\text{Actor NN: } \dot{\tilde{W}}_{u,i}(t) = -\dot{\hat{W}}_{u,i}(t) = \alpha_{u,i}\frac{\phi_{u,i}(x_i, \hat{m}_i)e^T_{ui}}{1 + \phi^T_{u,i}(x_i, \hat{m}_i)\phi_{u,i}(x_i, \hat{m}_i)} \tag{4.30}$$

Next, the performance of three neural networks is analyzed through the following theorems.

**Theorem 12.** *(Critic NNs' convergence)* Let $\hat{W}_{V,i}(t)$ be updated as (4.25), assume the learning rate $\alpha_{h,i} > 0$ then

1) the error between the actual and approximated critic NN's weights, i.e., $\tilde{W}_{V,i}$,

2) the optimal evaluation function approximation error, i.e., $\tilde{V}_i = V_i - \hat{V}_i$,

are uniformly ultimately bounded (UUB). The corresponding bounds $b_{W_{V,i}}$, $b_{V,i}$ are trivial when the reconstruction error is sufficiently small [96]. Similar to [91, 2], $\tilde{W}_{V,i}$ and $\tilde{V}_i$ will be asymptotically stable if the neuron network structure has been selected perfectly.

*Proof.* See Appendix A. □

**Theorem 13.** *(Mass NN's convergence)*: Let $\hat{W}_{m,i}(t)$ be updated as (4.26) shows, assume the learning rate $\alpha_{m,i} > 0$, then

1) the error between the actual and approximated critic NN's weights, i.e., $\tilde{W}_{m,i}$,

2) the FPK equation's approximation error, i.e., $\tilde{m}_i = m_i - \hat{m}_i$,

are uniformly ultimately bounded (UUB). The corresponding bounds $b_{W_{m,i}}$, $b_{m,i}$ are trivial when the reconstruction error is sufficiently small [96]. Similar to [91, 2], $\tilde{W}_{m,i}$ and $\tilde{m}_i$ will be asymptotically stable if the neuron network structures are selected perfectly.

*Proof.* See Appendix B. □

**Theorem 14.** *(Actor NN's convergence)*: Let $\hat{W}_{u,i}(t)$ be updated as (4.27) shows, assume the learning rate $\alpha_{u,i} > 0$, then

1) the error between the actual and approximated critic NN's weights, i.e., $\tilde{W}_{u,i}$,

2) the optimal provisioning rate approximation error, i.e., $\tilde{\beta}_i = \beta_i - \hat{\beta}_i$,

are uniformly ultimately bounded (UUB). The corresponding bounds $b_{W_{u,i}}$, $b_{u,i}$ are trivial when the reconstruction error is sufficiently small [96]. Similar to [91, 2], $\tilde{W}_{u,i}$ and $\tilde{\beta}_i$ will be asymptotically stable if the neuron network structures are selected perfectly.

*Proof.* See Appendix C. □

**Remark 3.** While the actor, critic, and mass NNs are learning the optimal provisioning rate, evaluation function, and PDF of all electric buses' SOC respectively, the bounds will reduce significantly and only depend on the NNs' reconstruction errors, which can be ignored when the perfect neuron numbers are selected [96].

Eventually, we analyze the closed-loop stability of proposed scheme. Before that, a lemma, which has been widely used in adaptive learning [21], is given as

**Lemma 3.** Given the stochastic system dynamic equations of the optimal control in (4.4), there exists optimal provisioning rate $\beta_i^*$ which satisfies

$$x_i \left[ -\beta_i^*(t) + c(t) + \sigma \frac{dW_i}{dt} \right] \leq -\gamma \left\| x_i \right\|^2 \tag{4.31}$$

where $\gamma > 0$.

**Theorem 15.** *(Closed-loop Stability)* Let the critic, actor, and mass NNs' weights being updates as (4.25), (4.27), (4.26), and assume the learning rates $\alpha_{h,i}, \alpha_{m,i}, \alpha_{u,i} > 0$, then $\tilde{W}_{V,i}$, $\tilde{W}_{m,i}$, $\tilde{W}_{u,i}, \tilde{V}_i, \tilde{m}_i, \tilde{\beta}_i$, $x_i$ are all UUB. Moreover, if all neuron networks' structures are selected perfectly [91], $\tilde{W}_{V,i}$, $\tilde{W}_{m,i}$, $\tilde{W}_{u,i}, \tilde{V}_i, \tilde{m}_i, \tilde{\beta}_i$, $x_i$ are all asymptotically stable.

*Proof.* See Appendix D. □

Additionally, the bounds of the approximation errors, i.e., $\tilde{V}_i, \tilde{m}_i, \tilde{\beta}_i$, can be calculated below

$$\begin{aligned} \|\tilde{V}_i(t)\| &= \|\tilde{W}_{V,i}^T(t)\phi_{V,i}(x_i, \hat{m}_i) + W_{V,i}^T\tilde{\phi}_{V,i}(x_i, \tilde{m}_i) + \varepsilon_{V,i}\| \\ &\leq \|\tilde{W}_{V,i}^T(t)\|\|\hat{\phi}_{V,i}\| + l_{\phi_{V,i}}\|W_{V,i}\|\|\tilde{m}_i\| + \|\varepsilon_{V,i}\| \\ &\leq b_{W_{V,i}}(t)\|\hat{\phi}_{V,i}\| + l_{\phi_{V,i}}\|W_{V,i}\|b_{m,i}(t) + \|\varepsilon_{V,i}\| \\ &\equiv b_{V,i}(t) \end{aligned} \tag{4.32}$$

where $\tilde{\phi}_{V,i}(x_i, \tilde{m}_i) = \phi_{V,i}(x_i, m_i) - \phi_{V,i}(x_i, \hat{m}_i)$, $l_{\phi_{V,i}}$ is the Lipschitz constant of the critic activation function $\phi_{V,i}(\cdot)$.

$$
\begin{aligned}
&\|\tilde{m}_i(t)\| = \|\tilde{W}_{m,i}^T(t)\phi_{m.i}(x_i,t) + \varepsilon_{m,i}\| \\
&\leq \|\tilde{W}_{m,i}\|\|\phi_{m,i}\| + \|\varepsilon_{m,i}\| \leq b_{W_{m,i}}(t)\|\phi_{m,i}\| + \|\varepsilon_{m,i}\| \\
&\equiv b_{m,i}(t)
\end{aligned} \tag{4.33}
$$

$$
\begin{aligned}
&\|\tilde{\beta}_i(t)\| = \|\tilde{W}_{u,i}^T(t)\hat{\phi}_{u,i} + W_{u,i}^T(t)\tilde{\phi}_{u,i}(x_i,\tilde{m}_i) + \varepsilon_{u,i}\| \\
&\leq \|\tilde{W}_{u,i}\|\|\hat{\phi}_{u,i}\| + \|W_{u,i}\|\|\tilde{\phi}_{u,i}(x_i,\tilde{m}_i)\| + \|\varepsilon_{u,i}\| \\
&\leq b_{W_{u,i}}(t)\|\hat{\phi}_{u,i}\| + l_{\phi_{u,i}}\|W_{u,i}\|\|\tilde{m}_i\| + \|\varepsilon_{u,i}\| \\
&\equiv b_{u,i}(t)
\end{aligned} \tag{4.34}
$$

where $l_{\phi_{u,i}}$ is the Lipschitz constant of the actor activation function $\phi_{u,i}(\cdot)$.

The complete ACM adaptive learning algorithm is summarised as a pseudo-code shown in Algorithm 1.

## 4.4 Simulation

In this section, we validate the effectiveness of the developed Actor-Critic-Mass (ACM) algorithm under a realistic simulation setup. Consider a city with 1000 electric buses that are operated on a fixed schedule. They are running during the day-time and going to various charging stations across the city when necessary. To reduce the impacts from increment charging power to the city's power grid, the time-of-use (TOU) price is enforced by the utility company. That is, the electricity price goes up as the total consumption (electric bus provisioning rate and residential energy consumption) goes up. In this chapter, we assume that there is no centralized monitoring and communication center for all individual vehicles due

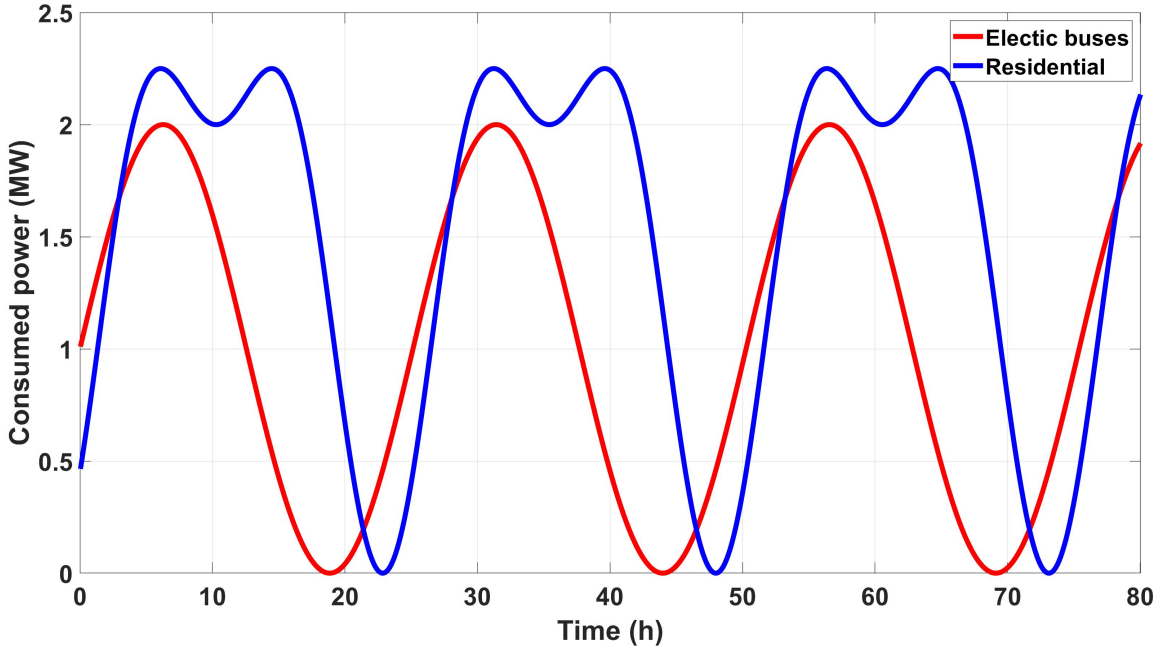



Figure 4.3: The plot of electricity consumption rate. The red curve represents the total consumption of all buses and the blue curve marks the residential electricity consumption. Both are averaged over 5 minutes.

to the budget limit. In other words, the charging strategy is purely self-controlled based on TOU (4.8), i.e.,

$$p(m,,z) = \frac{-N\frac{d}{dt}\mathbb{E}_x\{m(x,t)\} + c(t)}{\eta} + z(t)$$

It is clear in (4.8) that the TOU price also depends on two other factors, i.e., the eclectic buses' charging consumption rate $c(t)$ and the residential electricity consumption in the city. We define these consumption terms as periodic functions for a more realistic setup, which are plotted in fig. 4.3. In fig. 4.3, the red curve represent the empirical mean of all buses' power consumption's summation, i.e.,

$$c(t) = \sum_{i=1}^{N} \frac{1}{M} \sum_{k=1}^{M} c_i^{(k)}(t), \quad t \in [0,T], i = 1,2,\cdots,N$$

where $T$ is the period duration, and $c_i^{(k)}(t)$ represents $i$-th bus's consumption at

time $t$ in the $k$-th data instance with the assumption that $M$ consumption data instances are given. The blue curve similarly indicates the empirical mean of the residential consumption rate. Consider the practical operation of electric buses and most peoples' schedules, we selected one day (24 hours) as a period. The first peak appears at around 6:30 when the first batch residences are awake, and the electric buses start normal operation. The second peak is reached at 15:00 when the major appliances (e.g., AC and fridge) consume more electricity due to high temperatures. Therefore, to lower the TOU price, all electric buses are encouraged to purchase the electricity at the times that satisfy the two requirements, i.e., 1) the residential usage is low, 2) the electric buses are charged asynchronously to avoid creating a charging peak.

According to [76], the percentage of remaining power in the battery after a fixed distance can be modeled by a normal distribution. Therefore, we set the initial state of charge (SOC) as a normal distribution, i.e., $x(0) \sim \mathcal{N}(\mu = 0.2, \delta = 0.2)$. Note that the negative initials are replaced by their absolute value. To consider the power degradation, the discount factor in (4.3) is selected as $\eta = 0.8$ in this simulation. The parameters in the cost function (4.9) are selected as $R = 1$, $Q = 1$, and the terminal cost encourage to maximize the battery level, i.e.,

$$\gamma(x) = \frac{2}{(1 - x(T))^2}$$

The total simulation time is 80 hours.

Each electric bus maintains three neural networks, i.e., the critic neural network to approximate the optimal cost function, the mass neural network to approximate the SOC PDF, the actor neural network to estimate the optimal control (energy provisioning rate). The activation functions are selected from the expansion terms of polynomial $\sum_{k=1}^{M}(\sum_{j=1}^{n} z_j)k$ where $n$ represents the input number of the neural

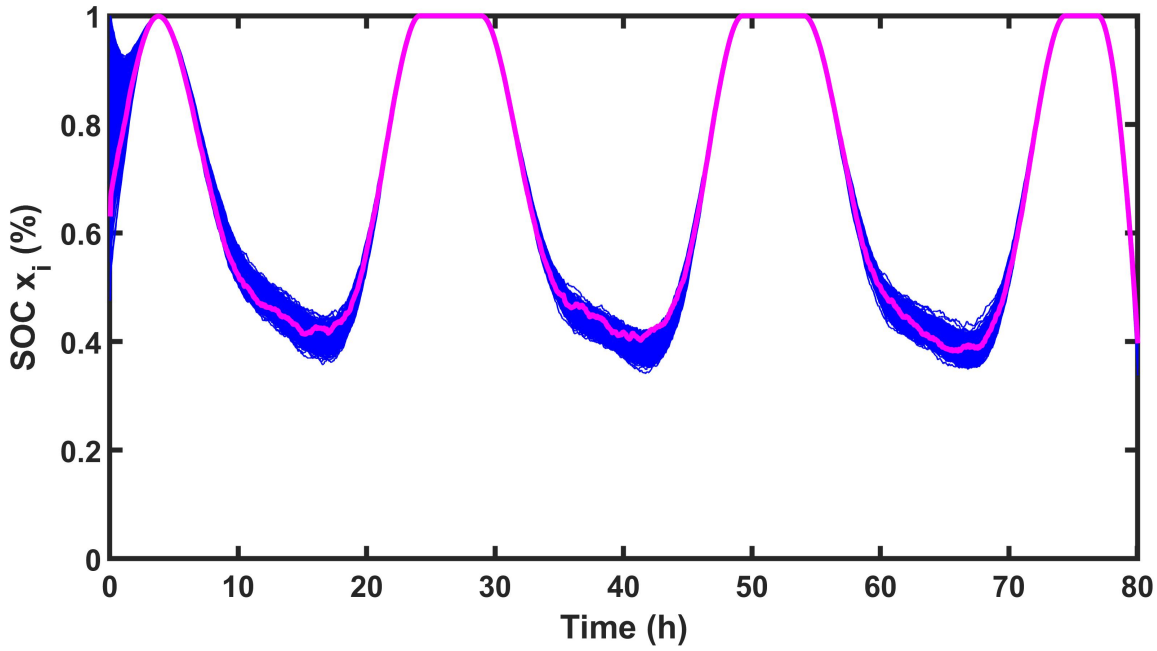



Figure 4.4: All buses' state of charge (SOC). The blue curves represent all individual bus's SOC trajectory. The magenta curve marks the average SOC.

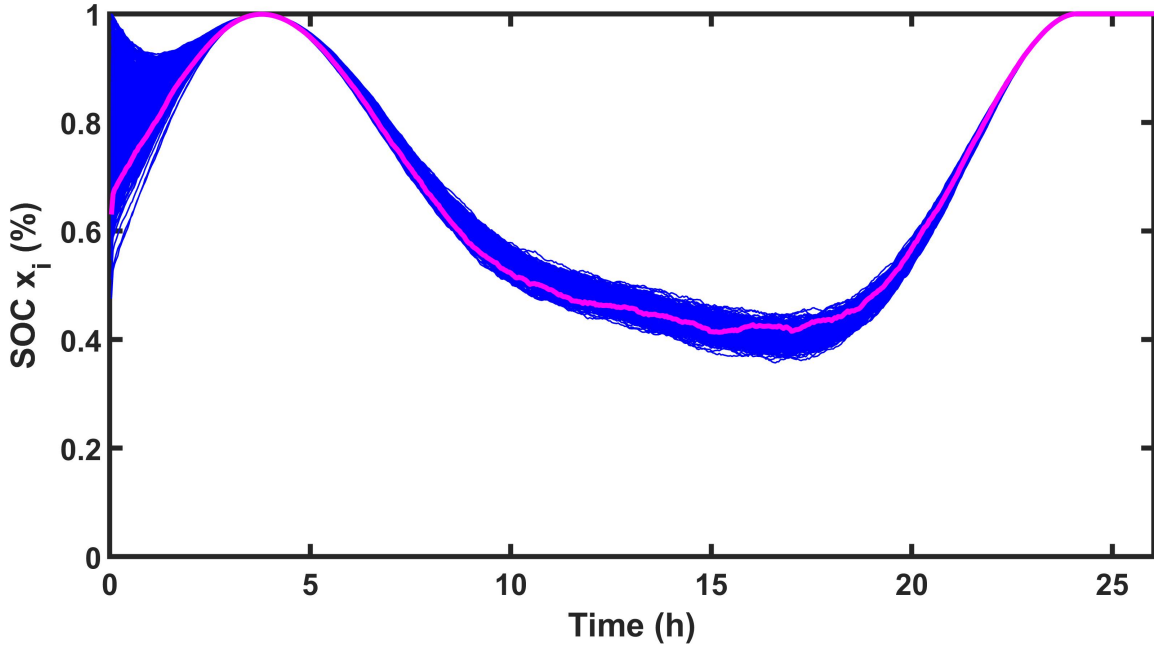



Figure 4.5: All buses' state of charge (SOC) in the first day. The blue curves represent all individual bus's SOC trajectory. The magenta curve marks the average SOC.

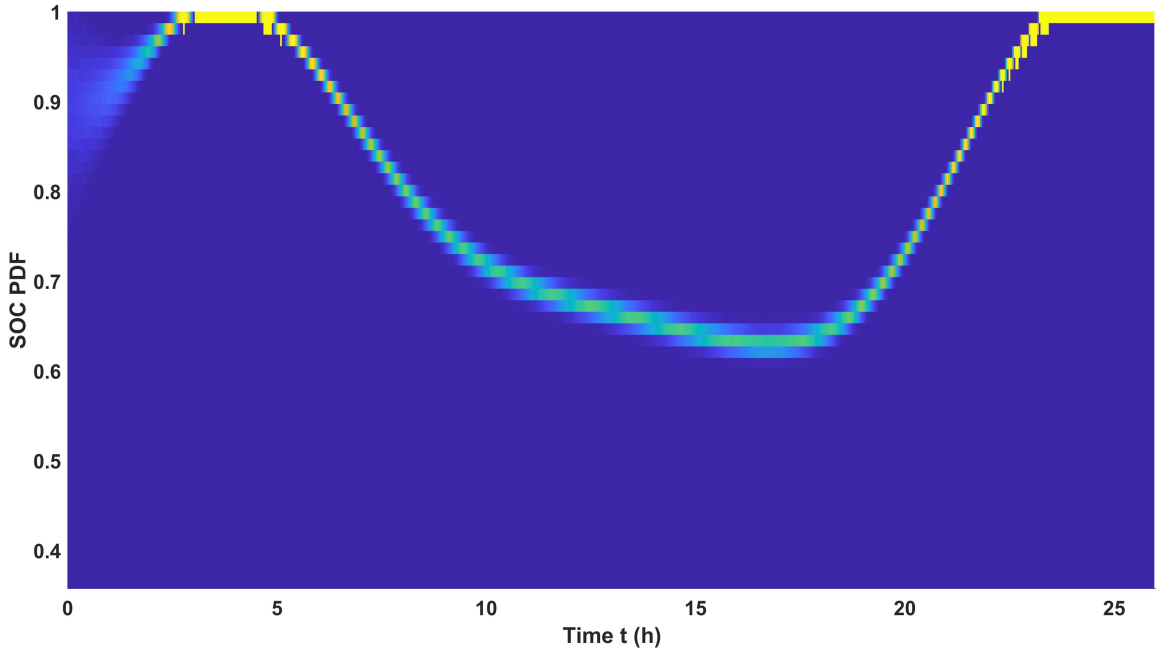



Figure 4.6: The plot of SOC PDF in first day.

network and $M$ stands for the estimation error. For the critic and actor neural networks, we selected $M = 4$. For the mass neural network, the constant $M$ is set to $5$.

Firstly, the plot of the time evolution of all buses' SOC, i.e., $1 - x$, in fig. 4.4. It is clear that the batteries are full when the residential consumption is high, which indicates that all buses will not purchase electricity when the residences' demand is rising. To demonstrate the evolution process in detail, the SOC on the first day is depicted in fig. 4.5. It is interesting to observe that all buses tend to charge collectively when the residential electricity demand is low. However, when the residential consumption is high, all buses' SOC varies (see 10-20 h), which means that they are coordinated to avoid inflating the TOU price by purchasing asynchronously. Figure 4.6 shows the time evolution of the PDF $m(1 - x, t)$ which reveals the same conclusion.

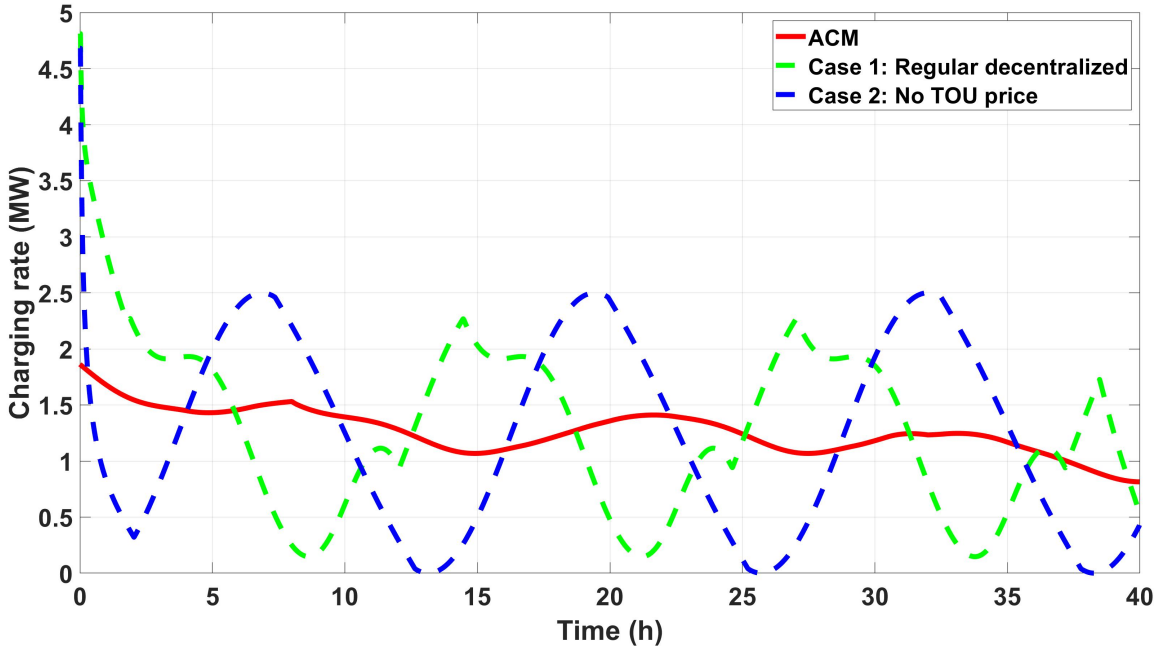



Figure 4.7: Summation of all buses' energy provisioning rate.

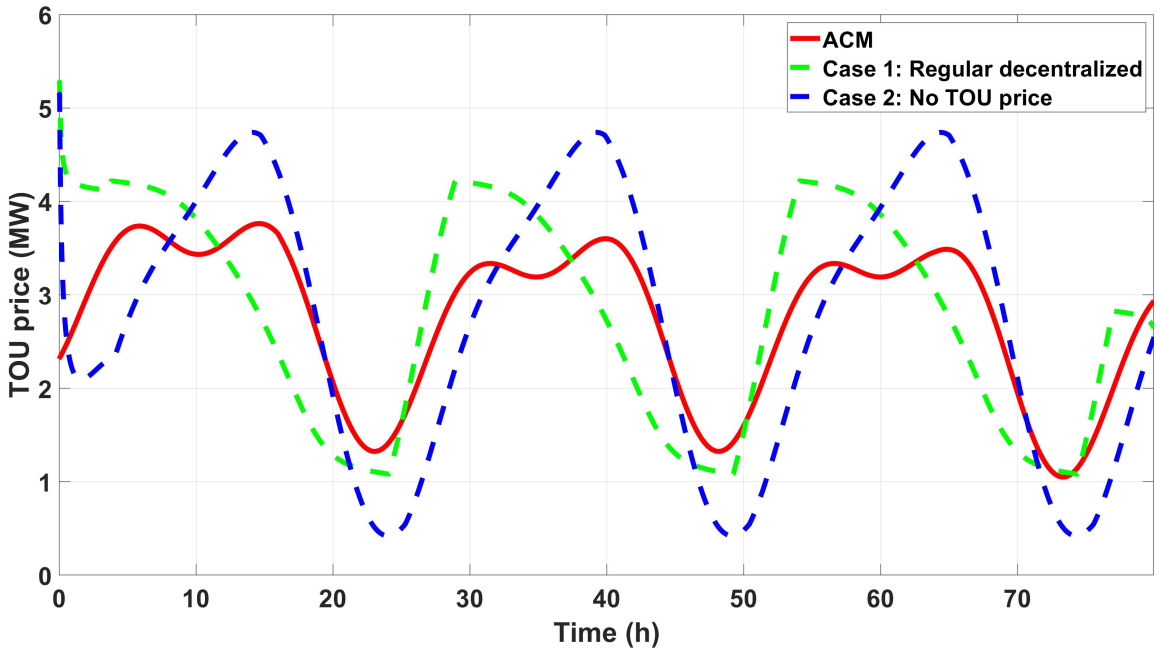



Figure 4.8: Summation of all buses and residential load

Next, we compare buses' energy purchase behavior of the developed ACM algorithm with two different non-optimal cases: 1) all buses purchase electricity without coordination but considering the TOU price, 2) all buses purchase without considering the TOU price. In fig. 4.7, the summation of the energy provisioning rate of all buses is plotted. It is clear to observe that the ACM algorithm's control is smooth, while the other two algorithms have caused undesirable surges to the power grid (e.g., 28h for case 1 and 40h for case 2. If TOU price is not considered (blue dashed curve), all buses charge at similar times, which will cause a more significant impact on the power grid. This can be further verified in fig. 4.8 where the charging power and the residential power usage are summed. In fig. 4.8, the developed ACM algorithm lower the peak demand and compensates the lowest point, which would significantly benefit the grid. The other two methods, however, have all caused surges.

Finally, we will demonstrate the optimality and the NNs' learning performance through the HJB equation error, i.e., critic NN's estimation error. To investigate more details, we show a single bus's HJB equation error in Figs. 4.9. We can see that the HJB equation error is bounded near zero after 25h. The HJB equation convergence confirms that the MFG equations are successfully estimated, and the $\epsilon$ Nash Equilibrium, which is considered optimal strategy for large scale electric bus charging game in this chapter, is reached. Provided by the SOC, TOU price, and the HJB equation's estimation error, the results demonstrate that the optimal strategy is reached for the very large scale electric bus charging problem.

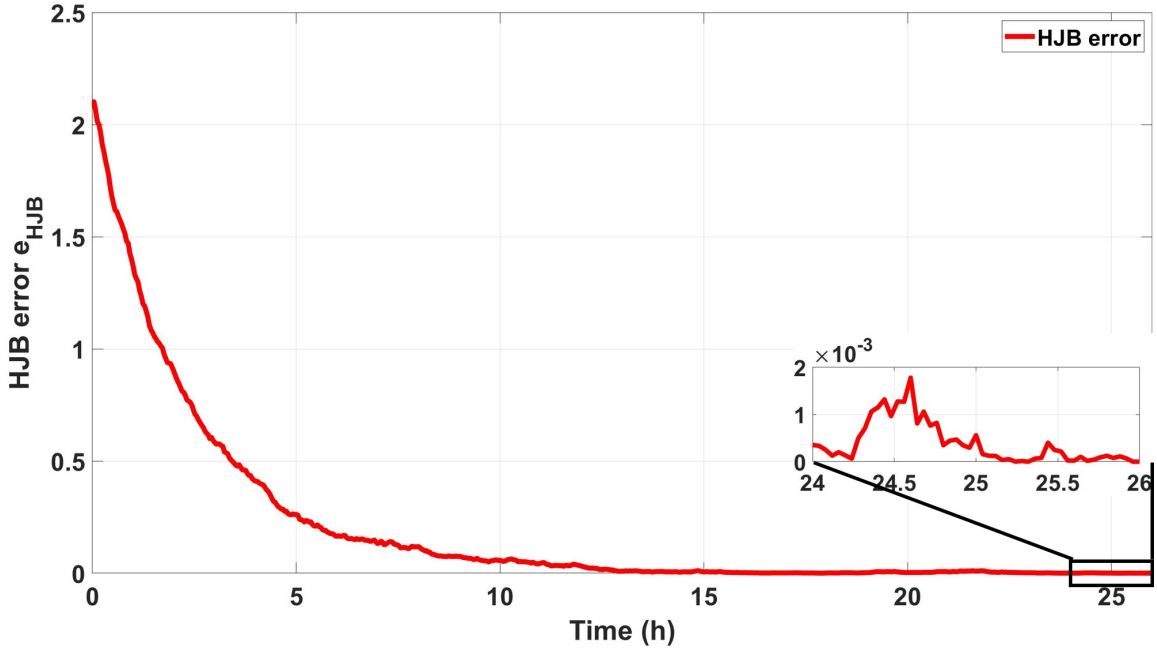



Figure 4.9: Bus 1's HJB error.

## 4.5 Conclusions

In this chapter, a novel Actor-Critic-Mass (ACM) based adaptive learning structure has been developed to solve the large scale of electric bus charging control problems by considering the TOU price. The developed decentralized algorithm can effectively obtain the optimal charging control strategy for each individual bus without communication. Specifically, the optimal $\epsilon_N$ Nash Equilibrium strategy is solved online by combining Mean Field Games (MFG) with optimal control theory. Moreover, a novel ACM algorithm which contains three neural networks has been developed to solve the coupled mean field equations online. The three neural networks include 1) the critic neural network to approximate the solution of Hamilton-Jacobi-Bellman (HJB), 2) the mass neural network to estimate the SOC PDF via Fokker-Planck-Kolmogorov (FPK) equation, and 3) the actor neural network to approximate the optimal provisioning rate. Finally, a series of numerical

simulations have been conducted to demonstrate the effectiveness and efficiency of the developed ACM based adaptive learning scheme.

Algorithm 1: ACM online optimal charging control

Initialize NN weights $\hat{W}_{V,i}$,$\hat{W}_{m,i}$,$\hat{W}_{u,i}$ randomly

Initialize $e_{FPKi}, e_{HJBi}, e_{ui}$ to be $\infty$

**while** True **do**

  **while** $e_{FPKi} > \delta_{FPKi}$ or $e_{HJBi} > \delta_{HJBi}$ or $e_{ui} > \delta_{ui}$ **do**

    Update mass NN weights by solving Eq. 4.26, i.e.,

$$\dot{\hat{W}}_{m,i} = -\alpha_{m,i} \frac{\hat{\Psi}_{m,i} e_{FPKi}^T}{1 + \hat{\Psi}_{m,i}^T \hat{\Psi}_{m,i}}$$

    Update critic NN weights by solving Eq. 4.25, i.e.,

$$\dot{\hat{W}}_{V,i} = -\alpha_{h,i} \frac{\hat{\Psi}_{V,i} e_{HJBi}^T}{1 + \hat{\Psi}_{V,i}^T \hat{\Psi}_{V,i}}$$

    Update actor NN weights by solving Eq. 4.27, i.e.,

$$\dot{\hat{W}}_{u,i} = -\alpha_{u,i} \frac{\hat{\phi}_{u,i} e_{ui}^T}{1 + \hat{\phi}_{u,i}^T \hat{\phi}_{u,i}}$$

    Update NNs' approximation errors by Eq. 4.15, 4.16, and 4.17, i.e.,

$$e_{HJBi} \leftarrow \Phi(m, x_i) + \hat{W}_{V,i}^T \hat{\Psi}_{V,i}$$

$$e_{FPKi} \leftarrow \hat{W}_{m,i}^T \hat{\Psi}_{m,i}$$

$$e_{ui} \leftarrow \hat{W}_{m,i}^T(t) \hat{\phi}_{u,i} + \frac{\eta}{2R} \frac{\partial}{\partial x_i} V_i \partial_x \tilde{V}_i$$

  **end while**

  $\hat{u}_i \leftarrow \hat{W}_{u,i}^T \hat{\phi}_{u,i} + \frac{\eta}{2R} \frac{\partial}{\partial x_i} V_i \partial_x \tilde{V}_i$

  Execute $\hat{u}_i$ and observe new state $x_i$

  Update NNs' approximation errors by Eq. 4.15, 4.16, and 4.17, i.e.,

$$e_{HJBi} \leftarrow \Phi(m, x_i) + \hat{W}_{V,i}^T \hat{\Psi}_{V,i}$$

$$e_{FPKi} \leftarrow \hat{W}_{m,i}^T \hat{\Psi}_{m,i}$$

$$e_{ui} \leftarrow \hat{W}_{m,i}^T(t) \hat{\phi}_{u,i} + \frac{\eta}{2R} \frac{\partial}{\partial x_i} V_i \partial_x \tilde{V}_i$$

**end while**

CHAPTER 5

# LARGE-SCALE MULTI-AGENT REINFORCEMENT LEARNING WITH APPLICATIONS IN OPTIMAL COMMUNICATION POWER CONTROL [115]

## 5.1 Introduction

The next generation wireless networks aim to support massive number of users with faster data rate and higher data quality. In order to achieve this vision, power allocation in next-generation wireless networks is one of the critical issues. However, it is challenging to adjust the wireless transmitter's power for satisfying individual communication connection's Quality of Service (QoS) under uncertain channel fading, user mobility, and large number of wireless connections. A balance must be achieved for the trade-off between the desire for users maximizing their individual QoS and the need for minimizing interference to other users. Furthermore, because of the mobility and features in mobile ad hoc network (MANET) at tactical edge and environmental uncertainties in the battlefield, many existing optimal power allocation algorithms are neither efficient nor practical. Therefore, a new type of decentralized intelligent dynamic power allocation is needed.

Early works on wireless communication power allocation focused on balancing the Signal-to-interference-plus-noise ratio (SINR) for all users by distributed control or centralized control. For example, the author in [109] developed distributed power allocation algorithms to maintain the QoS of limited number of users. Huang, Caines, and Malhame [38] proposed a centralized optimal power allocation algorithm by formulating the SINR requirement as resource allocation cost. However, at tactical edge, the communication environment is dynamic and the wireless network with massive users is often , also known as "ad hoc" [81],

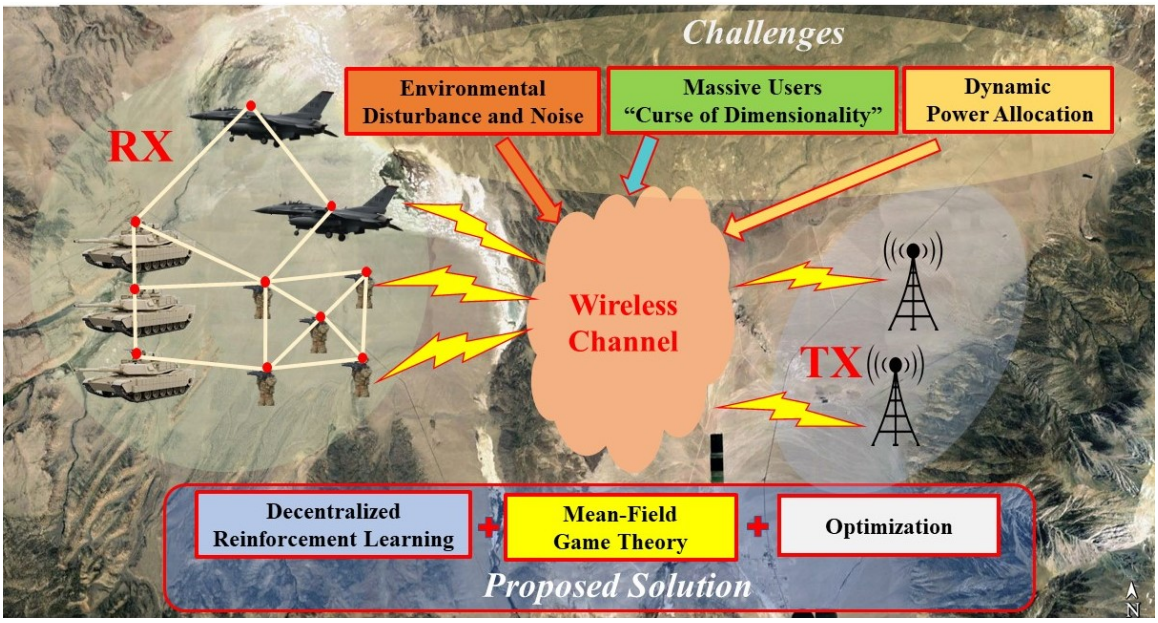



Figure 5.1: Proposed design for MANET in IoBT at tactical edge

which needs to be designed in a decentralized fashion. It is also worth noting that the efficiency of centralized algorithm would decrease significantly when the users' population is extremely large.

To overcome these challenges, a novel decentralized intelligent dynamic power allocation is developed in this chapter (see Fig. 5.1). Recently, a new type of decentralized multi-agent decision-making theory for non-cooperative games named the Mean Field Game has been developed by Gueant, Lasry and Lions [48]. It has been implemented in different areas successfully such as for electric vehicle charging power control [121], and for formation control [85]. The key idea of Mean Field control theory is to design a decentralized controller based only on local information and the impact from the whole population, i.e., mass. The mass is a global state averaged across all agents that is essentially the probability distribution function of all agent's states. Individual agent can focus on interacting with the global

state distribution instead of every other single agents to reduce the computational cost as well as communication cost. As a result, to obtain the Mean Field optimal control, each agent needs to minimize cost function under the mass effect. In optimal control [52], the optimal cost function can be attained by solving the Hamilton-Jacobian-Bellman (HJB) equation. Note that the HJB equation can only be solved backward-in-time due to the nature of the Bellman equation [52]. Moreover, the mass distribution can be obtained by solving the Fokker-Plank-Kolmogorov (FPK) equation under the assumption that the initial mass distribution is known. Thus, the solution of the HJB and FPK equation system is the decentralized optimal control for massive multi-agent system. It is shown in [48] that the coupled solution of HJB and FPK equations is the Nash equilibrium.

Although the Mean Field control theory is promising, solving the two coupled partial differential equations (PDEs), i.e. HJB and FPK, symbolically is computationally intensive. Especially, the FPK equation is solved forward-in-time whereas the HJB is solved backward-in-time. Meanwhile, Reinforcement learning and approximate dynamic programming (ADP) has been proved as an effective approach to numerically solve nonlinear HJB equations forward-in-time [2]. Specifically, Abu-Khalaf et.al have developed the actor-critic algorithm along with stability analysis to approximate the solution of HJB equation [2].

Inspired by those works, the decentralized dynamic power allocation problem for massive users has been formulated as an optimal control problem for massive multi-agent system in this study. Furthermore, a novel mean-field game theory based intelligent decentralized dynamic power allocation is proposed which can solve the coupled HJB and FPK equations through a novel actor-critic-mass algorithm including three neural networks (NN), denoted as the *Mass NN*, *Critic NN*,

and *Mass NN*, respectively. Through approximating optimal power allocation policy, optimal cost function, and power mass distribution by the three neural networks, a decentralized power allocation can be obtained in real-time that not only optimize the communication performance but also handle the uncertainties from harsh environment at tactical edge. The main contributions of this chapter can be summarized as,

- The proposed actor-critic-mass power allocation algorithm for wireless networks is designed in a decentralized fashion to handle extremely large scale users by integrating Mean Field Games with optimal control;
- The proposed actor-critic-mass algorithm is a new type of reinforcement learning and ADP framework that can solve the adaptive optimal design for wireless systems with large population through approximating the solution of FPK and HJB equation system simultaneously;
- The proposed actor-critic-mass algorithm can handle the uncertainties from harsh environment at tactical edge by dynamically adjust its tuning laws.

## 5.2 Problem formulation

### 5.2.1 System Model

In this chapter, we consider the channel model for wireless networks as lognormal fading. The stochastic channel power attenuation dynamics from the transmitter of the $j$-th link to the receiver of $i$-th link can therefore be modeled as

$$dx_{ij}(t) = -A\left(x_{ij}(t) + B\right)dt + \sigma_1 dw_i, \quad 1 \leq i \leq N \tag{5.1}$$

where $x_{ij}(t) = -D_{ij}$ with $D_{ij}$ being defined as the distance between transmitter $j$ and receiver $i$, $A, B \in \mathbb{R}$ are the system parameters related to the users' mobility and other losses [16], $w_i$ denotes the independent Wiener processes, and $N$ denotes the number of agents. Additionally, $\sigma_1$ is related to the volatility of the underlying lognormal shadowing effects.

Thus, the dynamics of power gain (loss) $g_{ij}$ can be obtained by substituting $g_{ij} = e^{x_{ij}}$ into Eq. 5.1 and get

$$dg_{ij} = g_{ij}\left[-A\left(\ln g_{ij} + B\right)\right]dt + g_{ij}\sigma_1 dw_i \tag{5.2}$$

and the received power can be represented as $g_{ij}p_i$.

Based on [38], the dynamics for transmitter power adjustment for the $i$-th user can be modeled as

$$dp_i(t) = u_i dt + \sigma_2 dw_i', \quad |u_i| \leq u_{i\max}, \quad 1 \leq i \leq N \tag{5.3}$$

where $p_i \in [0, p_{max}]$ represents the transmitted power with $p_{max}$ being maximum transmission power, $u_i$ represents the power adjustment rate, and $\sigma_2 dw_i'$ provides the random noise from the transmitter. The actual power attenuation loss (gain) can thus be computed by $g_{ij}(t) = e^{x_{ij}} \in (0, 1]$ and the actual transmitted power is $g_{ij}p_i$.

According to [40], the Signal-to-interference-plus-noise ratio (SINR) in large population users scenario is preferred to be defined as

$$\xi_i(t) = \frac{g_{ii}(t)p_i(t)}{I_i(t) + \eta_i} = \frac{g_{ii}p_i}{\sum_{j\neq i}^{N} \beta_N g_{ij}p_j + \eta_i} \tag{5.4}$$

where $I_i$ is the interference, $\eta_i \geq 0$ denotes the variance power of the noise at its receiver node, and $\beta_N \approx 1/N$. The objective of communication QoS requires that the SINR is higher than or equal to a desired threshold, i.e., $\xi_i \geq \mu_i$. Meanwhile,

the power allocation objective is to minimize the total power consumption for all users, i.e., $\min \sum_{j=1}^{N} p_j$. Based on [38], the solution of the total power minimization subject to the QoS constraint is defined as

$$\frac{g_{ii}p_i}{\sum_{j\neq i}^{N} \beta_N g_{ij}p_j + \eta_i} = \mu_i > 0 \tag{5.5}$$

which is equivalent to

$$\lim_{N\to\infty} \frac{g_{ii}p_i}{\sum_{j=1}^{N} \beta_N g_{ij}p_j + \eta_i} = \lim_{N\to\infty} \frac{\mu_i}{1+\beta_N\mu_i} = \mu_i \tag{5.6}$$

Therefore, we can define the cost function as

$$V_i(\boldsymbol{p}, \boldsymbol{g_i}, u_i) = \mathbb{E}\int_0^{\infty} \left\{ \left[ g_{ii}p_i - \mu_i \left( \beta_N \sum_{j=1}^{n} g_{ij}p_j + \eta_i \right) \right]^2 + Q_i p_i^2 + R_i u_i^2 \right\} dt$$

where $\boldsymbol{p}$ and $\boldsymbol{g_i}$ represent the sets of all users' power and power loss of the $i$th agent respectively, $R_i u_i^2$ represents the penalty of abrupt power adjustment and $Q_i p_i^2$ represents the additional penalty of high power. The optimal policy's objective for each user is to minimize Eq. 5.7.

### 5.2.2 Mean Field optimal control representation

Mean Field Game (MFG) theory [48] is an emerging technique that can effectively solve stochastic decision-making problem with a large population of agents in a decentralized manner.

Consider the given wireless channel and power allocation models, we can use the Probability Density Function (PDF) of the power gain and transmitter power for all users, i.e., $m_{g,i}(g,t)$ and $m_p(p,t)$ to compute the PDF of received power of $i$-th user, i.e. $m_{gp}(gp,t)$ with $gp \in \Theta = \{gp | 0 \leq gp \leq p_{max}\}$. While $m_{g,i}(g,t)$ for the $i$-th user is fixed since the dynamics of $g_{ij}$ for all agents are independent

of power adjustment, $m_p(p,t)$ is changing according to power adjustment policy. Let $\mho_i$ denote the power adjustment policy for the $i$-th user, i.e., $\mho_i = \{u_i(t)\big|t \in [0,\infty)\}$. Then the set of power adjustment policy for all users can be defined as $\mho = \{\mho_i\big|i \in [1,N]\}$. Given a $\mho$, the summation in the QoS constraint (i.e., Eq. 5.6) can be replaced by the expected value as

$$\mu_i(t) = \frac{g_{ii}(t)p_i(t)}{\mathbb{E}_{\mho}(g(t)p(t)) + \eta_i} \tag{5.7}$$

where

$$\mathbb{E}_{\mho}(g(t)p(t)) = \int_{\Theta} gpm_{g,i}(g,t)m_{p,\mho}(p,t)d(gp) \tag{5.8}$$

with the assumption that $g$ is independent of $p$ for massive users.

Then the cost function can be changed accordingly as

$$V_i(p_i, g_{ii}, u_i, m_{g,i}, m_{p,\mho}) = \mathbb{E}\int_0^{\infty} \left\{\Phi(p_i, g_{ii}, m_{g,i}, m_{p,\mho}) + Ru_i^2\right\} dt \tag{5.9}$$

with

$$\Phi(p_i, g_{ii}, m_{g,i}, m_{p,\mho}) = \left[g_{ii}p_i - \mu_i\left(\mathbb{E}_{\mho}(gp) + \eta_i\right)\right]^2 \tag{5.10}$$

According to Bellman's principle of optimality [52], the optimal cost for each agent from $t$ onward is defined as

$$V_i^*(t) = \min_{u_i}\left\{L(t) + V_i^*(t+dt)\right\} \tag{5.11}$$

where $L(t) = \Phi(p_i, g_{ii}, m_{g,i}, m_{p,\mho}) + Ru_i^2$ is the current running cost, and $V_i^*(t) = V_i(p_i(t), u_i(t), m_p(t))$. The optimal power adjustment policy $\mho_i^*$ that minimize Eq. 5.9 is the solution to Eq. 5.11. Because the power gain $g$ is independent of the power adjustment rate (i.e., $u_i$), we will write the cost function $V_i(p_i, g_{ii}, u_i, m_{g,i}, m_{p,\mho})$ as $V_i(p_i, u_i, m_{p,\mho})$ in the rest of the paper.

Next, the Hamiltonian can be defined as

$$H\left[p_i, \partial_p V_i(p_i, u_i, m_{p,\mho})\right] = L_i(p_i, u_i, m_{p,\mho}) + \partial_p V_i(p_i, u_i, m_{p,\mho}) u_i \tag{5.12}$$

By substituting optimal cost function $V_i^*(p_i, u_i, m_p)$ into Hamiltonian, Hamiltonian-Jacobian-Bellman (HJB) equation can be obtained as

$$-\partial_t V_i^*(p_i, u_i, m_p) - \frac{(\sigma_2)^2}{2} \partial_{pp} V_i^*(p_i, u_i, m_p) + H\left[p_i, \partial_p V_i^*(p_i, u_i, m_p)\right] = 0 \tag{5.13}$$

where $m_p$ represents the power mass distribution under the optimal power adjustment policy set (i.e., $\mho^*$).

The optimal power allocation for individual agent can be solved by

$$u_i^*(p_i, u_i, m_p) = -\frac{1}{2} R^{-1} \partial_p V_i^*(p_i, u_i, m_p) \tag{5.14}$$

Since each individual user is minimizing its own cost to attain decentralized optimal power allocation $u_i^*$, it can be considered as a nonzero-sum stochastic differential game. Therefore, there exists a Nash equilibrium (NE) point set such that the individual agent's cost is optimal [8].

To solve the HJB given in Eq. 5.13, the attenuation mass distribution $m_{g,i}(g_{ij}, t)$ and the transmitter power mass distribution $m_p(p_i, t)$ is needed. Recall the MFG and Eq. 5.2, the $m_{g,i}(g_{ij}, t)$ can be attained by solving the FPK equation based on the "Law of large numbers", i.e.,

$$\partial_t m_{g,i}(g_{ij}, t) = \partial_{gg}\left[\frac{(g_{ij}\sigma_1)^2}{2} m_{g,i}(g_{ij}, t)\right] - \partial_g\left\{g_{ij}\left[-A\left(\ln g_{ij} + B\right)\right]\right\} \tag{5.15}$$

Similarly, the $m_p(p_i, t)$ can be calculated by solving the following FPK equation,

$$\partial_t m_p(p_i, t) = \frac{(\sigma_2)^2}{2} \partial_{pp} m_p(p_i, t) - \partial_p u_i m_p(p_i, t) \tag{5.16}$$

According to relevant studies (see [48]), the existence and uniqueness of coupled HJB-FPK equation's solution can be guaranteed. Moreover, the optimal strategy given by the solution of the coupled HJB-FPK equation can be implemented into the game with finitely many players to have $\varepsilon_N$-Nash equilibrium, namely,

$$V_i(u_i; u_{-i}) \geq V_i(u_i^*; u_{-i}) - \varepsilon_N \tag{5.17}$$

where $\varepsilon_N$ is a bounded constant related to the population size $N$, $u_i$ can be any feasible control, and $u_{-i}$ represents power adjustment rate of all users other than $i$. See [48] for the detailed proofs.

**Remark 4.** To obtain the optimal design, one has to solve the coupled HJB-FPK equation simultaneously. However, the HJB equation (Eq. 5.13) is a Partial Differential Equation (PDE) that is solved backward-in-time whereas the FPK equation (Eq. 5.16) is solved forward-in-time. It makes the Mean Field type of design complicated and even impossible to solve directly in real-time. Therefore, in this chapter, a novel reinforcement learning and approximate dynamic programming (ADP) technique has been developed to learn the coupled HJB-FPK equation solution online.

## 5.3 Actor-critc-Mass Based Optimal Decentralized Power Allocation Design

In this section, a novel Actor-Critic-Mass (ACM) framework along with the reinforcement learning ADP technique is developed.Three Neural Networks (NNs) have been used not only to estimate the cost function and optimal power adjustment rate, but also to approximate the mass distribution of all users' power, i.e.

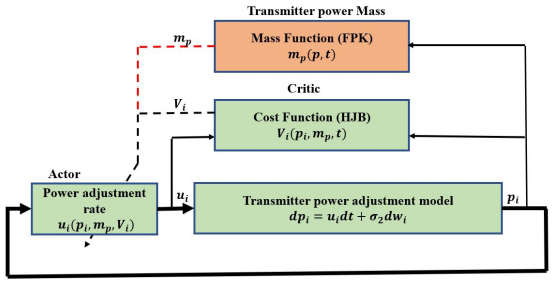



Figure 5.2: Structure of Actor-Critic-Mass system

$m_p(p_i, t)$. Specifically, three NNs in ACM are used to approximate the solutions of Eqs. 5.13, 5.14 and 5.15. The optimal cost function, design and mass distribution can be expressed as

$$\begin{cases} V_i^*(p_i, m_p, t) = W_{V,i}^T \phi_{V,i}(p_i, m_p, t) + \varepsilon_{V,i} \\ u_i^*(p_i, m_p, t) = W_{u,i}^T \phi_{u,i}(p_i, m_p, t) + \varepsilon_{u,i} \\ m_p(p_i, t) = W_{m,i}^T \phi_{m,i}(p_i, \bar{m}_p, t) + \varepsilon_{m,i} \end{cases} \tag{5.18}$$

where with $\bar{m}_i$ being the historical average power adjustment rate defined as $\bar{m}_p(t) = \frac{1}{\hat{t}} \int_{[t-\hat{t}]_+}^{t} \bar{p}_i(\tau) d\tau$, and $\hat{t}$ is a constant historical window, $\phi(\cdot)$ is a bounded and continues activation function, and $\varepsilon$ is the reconstruction error. Next, the optimal cost function, decentralized design and mass distribution function can be approximated as

$$\begin{cases} \hat{V}_i(p_i, \hat{m}_{p,i}, t) = \hat{W}_{V,i}^T(t) \phi_{V,i}(p_i, \hat{m}_{p,i}, t) \\ \hat{u}_i(p_i, \hat{m}_{p,i}, t) = \hat{W}_{u,i}^T(t) \phi_{u,i}(p_i, \hat{m}_{p,i}, t) \\ \hat{m}_{p,i}(p_i, \bar{m}_{p,i}, t) = \hat{W}_{m,i}^T(t) \phi_{m,i}(p_i, \bar{m}_{p,i}, t) \end{cases} \tag{5.19}$$

Substituting Eq. 5.19 into Eqs. 5.13, 5.14 and 5.15, equations will not hold. The residual errors will be introduced and used to tune the actor, critic, and mass NNs along with time, i.e.

$$e_{HJBi} = \hat{W}_{V,i}^T(t) \left[ \partial_t \hat{\phi}_{V,i} + \frac{(\sigma_2)^2}{2} \partial_{pp} \hat{\phi}_{V,i} - \hat{H}_{WV,i} \right] \tag{5.20}$$

$$e_{FPKi} = \hat{W}_{m,i}^T(t)\left[\partial_t\phi_{m,i} - \frac{(\sigma_2)^2}{2}\partial_{pp}\phi_{m,i} + \partial_p(\hat{u}_i\phi_{m,i})\right] \tag{5.21}$$

$$e_{ui} = \hat{W}_{u,i}^T(t)\hat{\phi}_{u,i} + \frac{1}{2}R_i^{-1}(p_i)\partial_p\hat{\phi}_{V,i} \tag{5.22}$$

where $\hat{\phi}_{V,i} = \phi_{V,i}(p_i, \hat{m}_{p,i}, t)$, $\hat{\phi}_{u,i} = \phi_{u,i}(p_i, \hat{m}_{p,i}, \hat{V}_i, t)$, and $\hat{H}_{WV,i} = H_{WV}\left[p_i, \partial_p\hat{\phi}_{V,i}\left(p_i, \hat{m}_{p,i}, t\right)\right]$ is the left term such that $\hat{W}_{V,i}^T(t)\hat{H}_{WV,i} = H\left(p_i, \partial_p V_i\left(p_i, \hat{m}_{p,i}, t\right)\right)$.

According to the the gradient descent algorithm, the ACM NNs' update laws can be derived as

$$\text{Critic NN: } \dot{\hat{W}}_{V,i}(t) = -\alpha_{h,i}\frac{\Psi_{V,i}(p_i, \hat{m}_{p,i}, t)e_{HJBi}^T}{1 + \Psi_{V,i}^T(p_i, \hat{m}_{p,i}, t)\Psi_{V,i}(p_i, \hat{m}_{p,i}, t)} \tag{5.23}$$

$$\text{Mass NN: } \dot{\hat{W}}_{m,i}(t) = -\alpha_{m,i}\frac{\Psi_{m,i}(p_i, \bar{m}_{p,i}, \hat{V}_i, t)e_{FPKi}^T}{1 + \Psi_{m,i}^T(p_i, \bar{m}_{p,i}, \hat{V}_i, t)\Psi_{m,i}(p_i, \bar{m}_{p,i}, \hat{V}_i, t)} \tag{5.24}$$

$$\text{Actor NN: } \dot{\hat{W}}_{u,i}(t) = -\alpha_{u,i}\frac{\phi_{u,i}(p_i, \hat{m}_{p,i}, t)e_{ui}^T}{1 + \phi_{u,i}^T(p_i, \hat{m}_{p,i}, t)\phi_{u,i}(p_i, \hat{m}_{p,i}, t)} \tag{5.25}$$

where $\alpha_{h,i}$, $\alpha_{m,i}$, and $\alpha_{u,i}$ are the learning rates, and

$$\Psi_{V,i}\left(p_i, \hat{m}_{p,i}, t\right) = \partial_t\hat{\phi}_{V,i} + \frac{\sigma_i^2}{2}\partial_{pp}\hat{\phi}_{V,i} - \hat{H}_{WV,i}$$

$$\Psi_{m,i}\left(p_i, \bar{m}_{p,i}, \hat{u}_i, t\right) = \partial_t\phi_{m,i} - \frac{\sigma_i^2}{2}\partial_{pp}\phi_{m,i} + \partial_p\left(\hat{u}_i\phi_{m,i}\right)$$

Note that the *Critic NN* and the *Actor NN* are updated continuously while the *Mass NN* is only updated between a fixed interval (i.e., $\Delta t$) to reduce computation. The complete ACM algorithm is summarized as a pseudo-code shown in Algorithm 1. Moreover, the structure of Actor-Critic-Mass framework is given in Fig. 5.2.

## 5.4 Simulation

In this section, the proposed intelligent decentralized dynamic power allocation algorithm has been evaluated under an Internet of Battlefield Things (IoBT)

Algorithm 2: Actor-Critic-Mass online optimal power allocation

1: Acquire agent number $i$

2: Initialize NN weights $\hat{W}_{V,i}$,$\hat{W}_{m,i}$,$\hat{W}_{u,i}$ randomly

3: Initialize $e_{FPKi}, e_{HJBi}, e_{ui}$ to be $\infty$

4: **while** True **do**

5: Update NNs' approximation errors by Eq. 5.20, 5.21, and 5.22, i.e.,

$$e_{HJBi} \leftarrow \Phi(m_i, x_i) + \hat{W}_{V,i}^T \hat{\Psi}_{V,i}$$

$$e_{FPKi} \leftarrow \hat{W}_{m,i}^T \hat{\Psi}_{m,i}$$

$$e_{ui} \leftarrow \hat{W}_{m,i}^T(t)\hat{\phi}_{u,i} + \frac{1}{2} R_i^{-1}(x_i)\partial_p \hat{\phi}_{V,i}$$

6: Update critic NN weights by solving Eq. 5.23, i.e.,

$$\dot{\hat{W}}_{V,i} = -\alpha_{h,i} \frac{\hat{\Psi}_{V,i} e_{HJBi}^T}{1 + \hat{\Psi}_{V,i}^T \hat{\Psi}_{V,i}}$$

7: Update actor NN weights by solving Eq. 5.25, i.e.,

$$\dot{\hat{W}}_{u,i} = -\alpha_{u,i} \frac{\hat{\phi}_{u,i} e_{ui}^T}{1 + \hat{\phi}_{u,i}^T \hat{\phi}_{u,i}}$$

8: **if** Current time $t = k\Delta t$, $k = 1, 2, 3, ...$ **then**

9: Update mass NN weights by solving Eq. 5.24, i.e.,

$$\dot{\hat{W}}_{m,i} = -\alpha_{m,i} \frac{\hat{\Psi}_{m,i} e_{FPKi}^T}{1 + \hat{\Psi}_{m,i}^T \hat{\Psi}_{m,i}}$$

10: **end if**

11: $\hat{u}_i \leftarrow \hat{W}_{u,i}^T \hat{\phi}_{u,i}$

12: Execute $\hat{u}_i$ and observe new transmitter power $p_i$

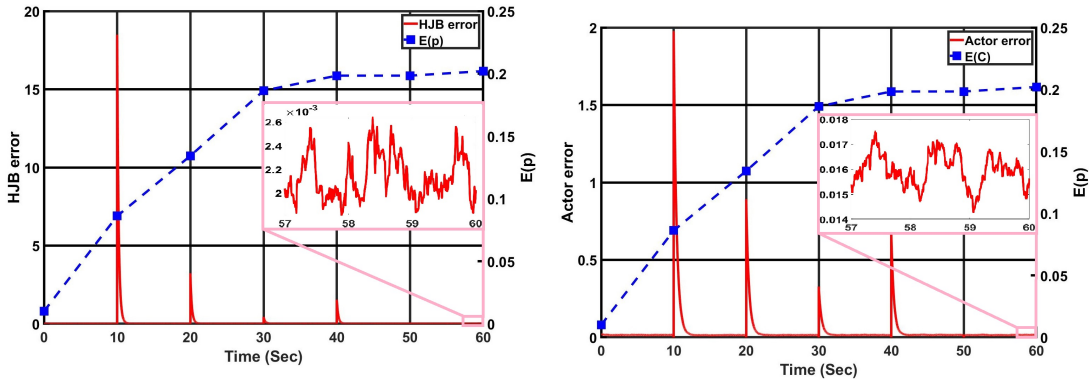
13: **end while**

wireless network with 10,000 agents. All agents in the network are expected to achieve a desired SINR value at $\mu = 0.6$. We selected the total transmitter power, power adjustment rate, and SINR as the metric to compare the performance between the proposed Actor-Critic-Mass algorithm and other algorithms.

The attenuation model and power adjustment model are given in Eq. 5.1 and Eq. 5.3 with $A = 4, B = 0.3$, $\sigma_1 = 0.3$, and $\sigma_2 = 0.01$. The noise variance in SINR (i.e., Eq. 5.4) is set to $\eta = 0.1$. The coefficient in the cost function Eq. 5.9 is selected as $R = 1$, $S = 1000$, and $Q = 1$. All agents' transmitters' power are set to zero at the beginning of the experiment. The channel attenuation of all ten agents are randomly initialized from a normal distribution with mean value of $0.2$, variance value of $0.01$ and can be varying due to the environment uncertainties, i.e., $m_0 \sim \mathcal{N}(-0.2, 0.01)$.

To estimate the solution of the HJB equation (i.e., Eq. 5.13), the FPK equation (i.e., Eq. 5.15), and the optimal power adjustment rate $u_i$, the *Critic NN*, *Mass NN*, and *Actor NN* are designed. The elements in activation functions for all NNs are constructed from the logsig of the expansion of the polynomial, i.e., $\text{logsig}(q_k(z))(z-\mathbb{E}(p))+z$, where $q_k(z)$ is the element of the expansion of $\sum_{\beta=1}^{M}(\sum_{j=1}^{n} z_j)^{\beta}$. $M = 5$ is the order of approximation, and $n = 2$ is the dimension of the input for *actor* and *critic* NNs, $n = 3$ for *Mass NN*. The weights of all neural networks are initialized randomly.

Firstly, the FPK equation (i.e., Eq. 5.15) has been solved for the power attenuation's PDF $m_{g,i}(g, t)$ with the boundary condition given above. Next, the *Critic NN*s and *Actor NN*s of all agents start to update continuously while the *Mass NNs* are updated every 10 seconds. Fig. 5.3 depicts the time evolution of agents' average NNs' estimation error. The red curves in Fig. 5.3 show that the *Critic NN* and

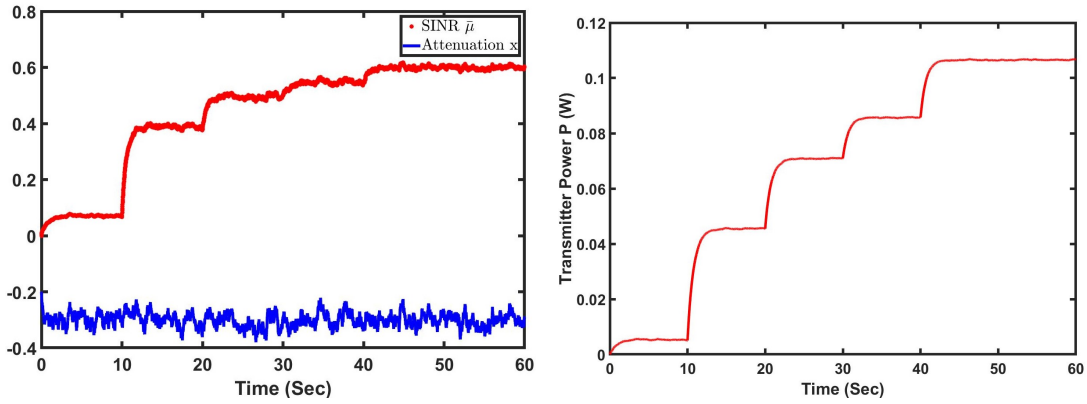



(a) *Critic NN* error (b) *Actor NN* error

Figure 5.3: The time evolution of agents' average NNs estimation error. The $m_p$ is updated every 10 seconds and blue dash line shows the time evolution of $\mathbb{E}(p)$. The red curve in (a) and (b) depicts the *Critic NN*'s and *Actor NN*'s error evolution with respect to time.

*Actor NN* errors converge to zero after 50 seconds. In other words, the *Critic NN* and *Actor NN* correctly approximate the solution of the HJB equation, particularly the optimal cost function and optimal power adjustment rate. The blue dash line in Fig. 5.3 is the plot of the expected value of power (i.e., $\mathbb{E}(p)$). The stability of $\mathbb{E}(p)$ proves that the *Mass NN* can approximate the density function of all agents' transmitter power distribution $m_p$ effectively, which is the solution of FPK equation. It is worth noting that the *Actor NN* and *Critic NN* had significant error increases at 10s which is caused by the update of the *Mass NN* and environment uncertainties. However, the level of increase in error decreases over time, i.e. the increase level at 20s is less than 10s. This also verifies that all three NNs are converging to the solution of Mean Field equation system.

We also plot the average SINR of agents along with channel attenuation $x$ for a randomly picked link, and the average transmitter power of agents in Fig. 5.4. It's easy to observe that the target SINR was reached after 40s despite of the uncertainty from channel attenuation. After the target SINR was reached, the transmitter




(a) SINR and $x$ (b) Transmitter power $p$

Figure 5.4: (a) The average SINR of agents is shown as the red curve. The blue curve represents the channel attenuation $x$ of one link. (b) Average transmitter power $p$ of agent is represented in red curve

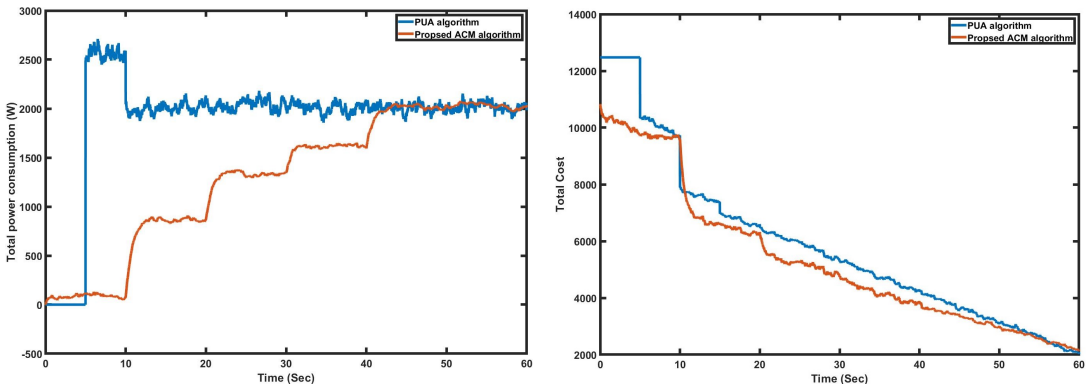
power remains stable as the result of the Nash equilibrium being satisfied.

Finally, the performance of proposed ACM algorithm is compared with another game theoretical algorithm which is the Parallel Update Algorithm (PUA) introduced in [5]. The parameters for PUA algorithm are picked as $L = u = 1$, $\sigma^2 = 0.1$, and $\lambda = 1.89$ in order to achieve SINR at $0.6$. The two algorithms are evaluated by the total power and total cost, which are defined as $P(t) = \sum_{i=1}^{N} p_i(t)$ and

$$J(t) = \int_t^T \left[ \sum_{i=1}^{N} (\hat{\mu}_i(\tau) - \mu) + \sum_{i=1}^{N} p_i(\tau) + \sum_{i=1}^{N} u_i(\tau) \right] d\tau \tag{5.26}$$

where $\hat{\mu}_i$ is the actual SINR of $i$-th user, and $T = 60s$ is the end time of simulation.

From Fig. 5.5(a), we can observe that both proposed algorithm and the PUA algorithm reached the same power level, which is the Nash Equilibrium. Note that although the PUA algorithm has a significant slower update rate compared to the ACM algorithm (5 seconds vs. continuously update), it reached the Nash Equilibrium in a shorter time. However, Fig. 5.5(b) demonstrates that the total cost of PUA




(a) Total power $P$ (b) Total cost $J$

Figure 5.5: (a) All transmitters' total power with respect to time. The PUA algorithm is updated every $5s$. (b) Total cost. The blue curve shows the performance of the PUA algorithm while the red curve shows the performance of proposed algorithm

is higher than ACM's. This verifies that the ACM algorithm's power adjustment policy outperforms PUA's policy in the sense of the summation of total power, QoS, and power adjustment rate (i.e. Eq. 5.26). Instead of an aggressive control policy in 5s of Fig. 5.5 from PUA, the ACM algorithm estimated the power mass distribution to avoid overshoot in total power. The Mean Field game provides necessary theoretic basis for an agent to estimate other agents' behavior (mass) so that all agents can avoid competing with each other by constantly increasing the transmitters' power. However in other decentralized game theory based algorithms, all agents can only adjust the power with delayed information about other agents which will often lead to destructive competition (see PUA's 5s in Fig. 5.5(a)). Intuitively, the Mean Field game serves as coordination in non-cooperative games so that all agents can reach Nash equilibrium with lower cost dynamic power allocation problems. It is also important to note that there's no information exchange between agents therefore the ACM algorithm consumes significantly less energy and channel usage compared to traditional distribute and centralized algorithm

especially with large scale agent population.

## 5.5 Conclusions

In this chapter, a novel Mean Field game theory based intelligent decentralized dynamic power allocation algorithm named the Actor-Critic-Mass (ACM) algorithm has been developed for massive users in MANET with application to IoBT at tactical edge. A cost function which can satisfy the SINR constraint as well as representing total wireless communication power consumption is constructed. The coupled HJB-FPK equation system is then derived and the solution represents the optimal cost function, the density function of massive agents' power, and the optimal power adjustment rate in the sense of Nash equilibrium. The *Actor NN*, the *Critic NN*, and the *Mass NN* are designed to effectively approximate the solution of the HJB-FPK equation system online. Compared with conventional centralized and distribute power allocation algorithm, the developed technique can significantly reduce the communication traffic as well as computational complexity for practical real-time MANET with massive number of users. A series of numerical simulations has been conducted to show the effectiveness and efficiency of the proposed design.

# CHAPTER 6
# CONCLUSIONS AND FUTURE WORK

## 6.1 Conclusions

In this dissertation, a decentralized reinforcement learning algorithm has been developed to solve the previously unsolvable large-scale multi-agent systems (MAS) optimal control problem. The traditional MAS optimal control algorithms suffer from the well-known "Curse of Dimensional" problem and communication problem. In the developed ACM algorithm, the mean field games are introduced to deal with the problems caused by the increasing agent number.

The general framework and theoretical foundations of the ACM algorithm are described in Chapter 2. The large-scale MAS optimal control problem is first designed and reformulated into the mean field type optimal control. By replacing all agents' states with the probability density function (PDF), the algorithm's computational complexity is decoupled with the agent number. Moreover, the PDF can be obtained by locally solve the Fokker-Planck-Kolmogorov (FPK) equation. As a result, communication between all agents is not necessary. Following the state-of-art reinforcement learning algorithm, a learning-based framework is designed to approximate the optimal solution set. Specifically, the actor, critic, and mass neural networks are proposed to approximate the optimal control, optimal cost function, PDF, respectively. In Chapter 2, the numerical simulations on both linear and nonlinear systems are conducted to show the effectiveness of the ACM algorithm. The Lyapunov stability analysis is also provided to demonstrate the stability.

In Chapter 3, the mean field type optimal control with heterogeneous agents

is discussed in the pursuit-evasion game setup. The original mean field game theory is designed based on particles that have homogeneous physical dynamics. However, in practical applications, the agents often have different dynamics. To leverage that, the ACM-Opponent algorithm has been designed to deal with a large-scale MAS with two different types of agents, i.e., the pursuers and evaders.

Provided by the theoretical support in the past two chapters. Two case-studies from the practical large-scale MAS optimal control problems has been provided in Chapter 4 and 5. In Chapter 4, the optimal schedule of the electrical buses' charging problem is considered. As electrical buses are getting popular in smart cities, the charging time of the buses has become critical to the stability of the city's power grid. The ACM algorithm can effectively coordinate the buses so that the power consumption of the charging schedule compensates for the valley time of the residential and industrial power consumption. Chapter 5 demonstrates another application example of applying the ACM algorithm to optimize the transmission power of devices in a wireless network. It is proved in the experiment section that the targeted Quality of Service (QoS) can be achieved with the minimum transmission power.

## 6.2 Future Work

Although effective in some cases, the ACM algorithm still suffers from unrealistic assumptions due to the limit of its core theory, i.e., the mean field game theory. For example, the current ACM algorithm cannot deal with constrained control or state space. Moreover, the neural networks used in this dissertation are naive. It is more desired to try a deeper neural network to enhance the approximation per-

formance. Finally, since the Kolmogorov equation has the ability to leverage the uncertainties in the physical dynamics, it is also possible to expand the domain of a stochastic policy space by noise. Therefore, it is promising to explore mean field-based transfer learning methods.

## BIBLIOGRAPHY

[1] Judah Aber. Electric Bus Analysis for New York City Transit, 2016.

[2] Murad Abu-Khalaf and Frank L. Lewis. Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach. *Automatica*, 2005.

[3] Mamta Agiwal, Abhishek Roy, and Navrati Saxena. Next Generation 5G Wireless Networks: A Comprehensive Survey. *IEEE Communications Surveys Tutorials*, 18(3):1617–1655, 2016. Conference Name: IEEE Communications Surveys Tutorials.

[4] Noha Almulla, Rita Ferreira, and Diogo Gomes. Two numerical approaches to stationary mean-field games. *Dynamic Games and Applications*, 7(4):657–682, 2017.

[5] Tansu Alpcan, Tamer Başar, Rayadurgam Srikant, and Eitan Altman. Cdma uplink power control as a noncooperative game. *Wireless Networks*, 8(6):659–670, 2002.

[6] Andrew G Barto, Richard S Sutton, and Charles W Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE transactions on systems, man, and cybernetics*, 5:834–846, 1983.

[7] Richard Bellman. Dynamic programming. *Science*, 153(3731):34–37, 1966.

[8] A Bensoussan and J Frehse. Stochastic games for n players. *Journal of optimization theory and applications*, 105(3):543–565, 2000.

[9] Alain Bensoussan, Jens Frehse, and Phillip Yam. *Mean Field Games and Mean Field Type Control Theory*. SpringerBriefs in Mathematics. Springer-Verlag, New York, 2013.

[10] Lucian Buşoniu, Robert Babuška, and Bart De Schutter. A comprehensive survey of multiagent reinforcement learning. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 38(2):156–172, March 2008.

[11] Lucian Buşoniu, Robert Babuška, and Bart De Schutter. Multi-agent reinforcement learning: An overview. *Studies in Computational Intelligence*, 310:183–221, 2010. Publisher: Springer, Berlin, Heidelberg ISBN: 9783642144349.

[12] Luis Búrdalo, Andrés Terrasa, Vicente Julián, and Ana García-Fornes. The Information Flow Problem in multi-agent systems. *Engineering Applications of Artificial Intelligence*, 70:130–141, April 2018. Publisher: Elsevier Ltd.

[13] Peter E. Caines, Minyi Huang, and Roland P. Malhamé. Mean Field Games. In Tamer Basar and Georges Zaccour, editors, *Handbook of Dynamic Game Theory*, pages 1–28. Springer International Publishing, Cham, 2017.

[14] Efe Camci and Erdal Kayacan. Game of drones: UAV pursuit-evasion game with type-2 fuzzy logic controllers tuned by reinforcement learning. In *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 618–625, July 2016.

[15] Pierre Cardaliaguet and Saeed Hadikhanloo. Learning in mean field games: the fictitious play. *ESAIM: Control, Optimisation and Calculus of Variations*, 23(2):569–591, 2017.

[16] Charalambos D Charalambous and Nickie Menemenlis. Stochastic models for long-term multipath fading channels and their statistical properties. In *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No. 99CH36304)*, volume 5, pages 4947–4952. IEEE, 1999.

[17] Jiejie Chen, Boshan Chen, and Zhigang Zeng. Synchronization and Consensus in Networks of Linear Fractional-Order Multi-Agent Systems via Sampled-Data Control. *IEEE Transactions on Neural Networks and Learning Systems*, 31(8):2955–2964, August 2020. Conference Name: IEEE Transactions on Neural Networks and Learning Systems.

[18] Romain Couillet, Samir M. Perlaza, Hamidou Tembine, and Mérouane Debbah. Electrical vehicles in the smart grid: A mean field game analysis. *IEEE Journal on Selected Areas in Communications*, 30(6):1086–1096, 2012. arXiv: 1110.1732.

[19] Hong-Ning Dai, Raymond Chi-Wing Wong, Hao Wang, Zibin Zheng, and Athanasios V. Vasilakos. Big Data Analytics for Large-scale Wireless Networks: Challenges and Opportunities. *ACM Computing Surveys*, 52(5):99:1–99:36, September 2019.

[20] Antonio De Paola, Vincenzo Trovato, David Angeli, and Goran Strbac. A Mean Field Game Approach for Distributed Control of Thermostatic Loads Acting in Simultaneous Energy-Frequency Response Markets. *IEEE Trans-*

*actions on Smart Grid*, 10(6):5987–5999, November 2019. Conference Name: IEEE Transactions on Smart Grid.

[21] T. Dierks and S. Jagannathan. Optimal control of affine nonlinear continuous-time systems using an online Hamilton-Jacobi-Isaacs formulation. In *49th IEEE Conference on Decision and Control (CDC)*, pages 3048–3053, December 2010. ISSN: 0191-2216.

[22] L. Ding, S. Li, H. Gao, Y. Liu, L. Huang, and Z. Deng. Adaptive neural network-based finite-time online optimal tracking control of the nonlinear system with dead zone. *IEEE Transactions on Cybernetics*, pages 1–11, 2019.

[23] Boualem Djehiche, Alain Tcheukam, and Hamidou Tembine. A mean-field game of evacuation in multilevel building. *IEEE Transactions on Automatic Control*, 62(10):5154–5169, 2017.

[24] Khac Duc Do. Flocking for multiple elliptical agents with limited communication ranges. *IEEE transactions on robotics*, 27(5):931–942, 2011.

[25] A. Dorri, S. S. Kanhere, and R. Jurdak. Multi-Agent Systems: A Survey. *IEEE Access*, 6:28573–28593, 2018. Conference Name: IEEE Access.

[26] Chun-Xia Dou and Bin Liu. Multi-Agent Based Hierarchical Hybrid Control for Smart Microgrid. *IEEE Transactions on Smart Grid*, 4(2):771–778, June 2013. Conference Name: IEEE Transactions on Smart Grid.

[27] Y. Feng, W. Zhang, J. Xiong, H. Li, and L. Rutkowski. Event-triggering interaction scheme for discrete-time decentralized optimization with nonuniform step sizes. *IEEE Transactions on Cybernetics*, pages 1–10, 2020.

[28] E. Garcia, D. W. Casbeer, A. Von Moll, and M. Pachter. Multiple Pursuer Multiple Evader Differential Games. *IEEE Transactions on Automatic Control*, pages 1–1, 2020. Conference Name: IEEE Transactions on Automatic Control.

[29] Xiaohu Ge, Song Tu, Guoqiang Mao, Cheng-Xiang Wang, and Tao Han. 5g ultra-dense cellular networks. *IEEE Wireless Communications*, 23(1):72–79, 2016.

[30] Diogo A. Gomes and João Saúde. Mean Field Games Models—A Brief Survey. *Dynamic Games and Applications*, 4(2):110–154, June 2014.

[31] Ivo Grondman, Lucian Busoniu, Gabriel AD Lopes, and Robert Babuska. A survey of actor-critic reinforcement learning: Standard and natural policy gradients. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):1291–1307, 2012.

[32] Nallappan Gunasekaran, Guisheng Zhai, and Qiang Yu. Sampled-data synchronization of delayed multi-agent networks and its application to coupled circuit. *Neurocomputing*, 413:499–511, November 2020.

[33] Jayesh K. Gupta, Maxim Egorov, and Mykel Kochenderfer. Cooperative Multi-agent Control Using Deep Reinforcement Learning. In Gita Sukthankar and Juan A. Rodriguez-Aguilar, editors, *Autonomous Agents and Multiagent Systems*, Lecture Notes in Computer Science, pages 66–83, Cham, 2017. Springer International Publishing.

[34] Olivier Guéant, Jean-Michel Lasry, and Pierre-Louis Lions. Mean Field Games and Applications. In Areski Cousin, Stéphane Crépey, Olivier Guéant, David Hobson, Monique Jeanblanc, Jean-Michel Lasry, Jean-Paul Laurent, Pierre-Louis Lions, and Peter Tankov, editors, *Paris-Princeton Lectures on Mathematical Finance 2010*, Lecture Notes in Mathematics, pages 205–266. Springer, Berlin, Heidelberg, 2011.

[35] Jiequn Han, Arnulf Jentzen, and E. Weinan. Solving high-dimensional partial differential equations using deep learning. *Proceedings of the National Academy of Sciences of the United States of America*, 115(34):8505–8510, 2018. arXiv: 1707.02568.

[36] Hado Hasselt. Double q-learning. *Advances in neural information processing systems*, 23:2613–2621, 2010.

[37] Yiguang Hong, Jiangping Hu, and Linxin Gao. Tracking control for multi-agent consensus with an active leader and variable topology. *Automatica*, 2006.

[38] Minyi Huang, Peter E Caines, and Charalambos D Charalambous. Stochastic power control for wireless systems: Classical and viscosity solutions. In *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, volume 2, pages 1037–1042. IEEE, 2001.

[39] Minyi Huang, Peter E Caines, and Roland P Malhamé. Large-population cost-coupled lqg problems with nonuniform agents: individual-mass behav-

ior and decentralized $varepsilon$-nash equilibria. *IEEE transactions on automatic control*, 52(9):1560–1571, 2007.

[40] Minyi Huang, Roland P Malhamé, and Peter E Caines. Stochastic power control in wireless communication systems: analysis, approximate control algorithms and state aggregation. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*, volume 4, pages 4231–4236. IEEE, 2003.

[41] Minyi Huang, Caines PE, and R.P. Malhame. Individual and mass behaviour in large population stochastic wireless power control problems: centralized and Nash equilibrium solutions. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, volume 1, pages 98–103 Vol.1, December 2003. ISSN: 0191-2216.

[42] Ying-Chao Hung and George Michailidis. Modeling and Optimization of Time-of-Use Electricity Pricing Systems. *IEEE Transactions on Smart Grid*, 10(4):4116–4127, July 2019. Conference Name: IEEE Transactions on Smart Grid.

[43] Ayesha Ijaz, Lei Zhang, Maxime Grau, Abdelrahim Mohamed, Serdar Vural, Atta U Quddus, Muhammad Ali Imran, Chuan Heng Foh, and Rahim Tafazolli. Enabling massive iot in 5g and beyond systems: Phy radio frame design considerations. *IEEE Access*, 4:3322–3339, 2016.

[44] Qi Kang, JiaBao Wang, MengChu Zhou, and Ahmed Chiheb Ammari. Centralized Charging Strategy and Scheduling Algorithm for Electric Vehicles Under a Battery Swapping Scenario. *IEEE Transactions on Intelligent Transportation Systems*, 17(3):659–669, March 2016. Conference Name: IEEE Transactions on Intelligent Transportation Systems.

[45] Max Katsev, Anna Yershova, Benjamín Tovar, Robert Ghrist, and Steven M. LaValle. Mapping and Pursuit-Evasion Strategies For a Simple Wall-Following Robot. *IEEE Transactions on Robotics*, 27(1):113–128, February 2011. Conference Name: IEEE Transactions on Robotics.

[46] Adia Khalid, Nadeem Javaid, Abdul Mateen, Manzoor Ilahi, Tanzila Saba, and Amjad Rehman. Enhanced Time-of-Use Electricity Price Rate Using Game Theory. *Electronics*, 8(1):48, January 2019. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute.

[47] Bahare Kiumarsi, Frank L Lewis, Hamidreza Modares, Ali Karimpour, and

Mohammad-Bagher Naghibi-Sistani. Reinforcement q-learning for optimal tracking control of linear discrete-time systems with unknown dynamics. *Automatica*, 50(4):1167–1175, 2014.

[48] Jean-Michel Lasry and Pierre-Louis Lions. Mean field games. *Japanese Journal of Mathematics*, 2(1):229–260, March 2007.

[49] Frank L Lewis and Kyriakos G Vamvoudakis. Reinforcement learning for partially observable dynamic processes: Adaptive dynamic programming using measured output data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(1):14–25, 2010.

[50] Frank L Lewis and Draguna Vrabie. Reinforcement learning and adaptive dynamic programming for feedback control. *IEEE Circuits and Systems Magazine*, 9(3):32–50, 2009.

[51] Frank L Lewis, Draguna Vrabie, and Vassilis L Syrmos. *Optimal Control, 3rd Edition.* John Wiley & Sons, 2012. OCLC: 940552625.

[52] Frank L. Lewis, Draguna Vrabie, and Kyriakos G. Vamvoudakis. Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers. *IEEE Control Systems*, 32(6):76–105, 2012. Publisher: IEEE.

[53] Huaqing Li, Xiaofeng Liao, Tingwen Huang, and Wei Zhu. Event-Triggering Sampling Based Leader-Following Consensus in Second-Order Multi-Agent Systems. *IEEE Transactions on Automatic Control*, 60(7):1998–2003, July 2015. Conference Name: IEEE Transactions on Automatic Control.

[54] Kuo Li, Changchun Hua, Xiu You, and Xinping Guan. Distributed output-feedback consensus control for nonlinear multiagent systems subject to unknown input delays. *IEEE Transactions on Cybernetics*, 2020.

[55] Weihua Li, Huaguang Zhang, Shaoxin Sun, and Juan Zhang. Fully distributed event-triggered consensus protocols for multi-agent systems with physically interconnected network. *Neurocomputing*, 418:191–199, December 2020.

[56] Wenjin (Jason) Li, Xiaoqi Tan, Bo Sun, and Danny H. K. Tsang. Optimal power dispatch of a centralised electric vehicle battery charging station with renewables. *IET Communications*, 12(5):579–585, December 2017. Publisher: IET Digital Library.

[57] Xiaolei Li, Changyun Wen, and Ci Chen. Adaptive formation control of networked robotic systems with bearing-only measurements. *IEEE Transactions on Cybernetics*, 2020.

[58] Ruiwen Liao, Liang Han, Xiwang Dong, Qingdong Li, and Zhang Ren. Finite-time formation-containment tracking for second-order multi-agent systems with a virtual leader of fully unknown input. *Neurocomputing*, 415:234–246, November 2020.

[59] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[60] Wei Lin, Zhihua Qu, and Marwan A. Simaan. Nash strategies for pursuit-evasion differential games involving limited observations. *IEEE Transactions on Aerospace and Electronic Systems*, 51(2):1347–1356, April 2015. Conference Name: IEEE Transactions on Aerospace and Electronic Systems.

[61] C. Liu, H. Zhang, Y. Luo, and H. Su. Dual heuristic programming for optimal control of continuous-time nonlinear systems using single echo state network. *IEEE Transactions on Cybernetics*, pages 1–12, 2020.

[62] M. Liu, Y. Wan, F. L. Lewis, and V. G. Lopez. Adaptive Optimal Control for Stochastic Multiplayer Differential Games Using On-Policy and Off-Policy Reinforcement Learning. *IEEE Transactions on Neural Networks and Learning Systems*, 31(12):5522–5533, December 2020. Conference Name: IEEE Transactions on Neural Networks and Learning Systems.

[63] Ryan Liu, Luther Dow, and Edwin Liu. A survey of PEV impacts on electric utilities. In *ISGT 2011*, pages 1–8, January 2011.

[64] Yifan Liu, Tieshan Li, Qihe Shan, Renhai Yu, Yue Wu, and C. L. Philip Chen. Online optimal consensus control of unknown linear multi-agent systems via time-based adaptive dynamic programming. *Neurocomputing*, 404:137–144, September 2020.

[65] Victor G. Lopez, Frank L. Lewis, Yan Wan, Edgar N. Sanchez, and Lingling Fan. Solutions for Multiagent Pursuit-Evasion Games on Communication Graphs: Finite-Time Capture and Asymptotic Behaviors. *IEEE Transactions on Automatic Control*, 65(5):1911–1923, May 2020. Conference Name: IEEE Transactions on Automatic Control.

[66] Yongfeng Lv, Xuemei Ren, and Jing Na. Online optimal solutions for multi-player nonzero-sum game with completely unknown dynamics. *Neurocomputing*, 283:87–97, March 2018.

[67] Yongfeng Lv, Xuemei Ren, and Jing Na. Adaptive optimal tracking controls of unknown multi-input systems based on nonzero-sum game theory. *Journal of the Franklin Institute*, 356(15):8255–8277, October 2019.

[68] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[69] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi. Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications. *IEEE Transactions on Cybernetics*, pages 1–14, 2020.

[70] Mojtaba Nourian, Peter E. Caines, Roland P. Malhamé, and Minyi Huang. Mean field LQG control in leader-follower stochastic multi-agent systems: Likelihood ratio based adaptation. *IEEE Transactions on Automatic Control*, 57(11):2801–2816, 2012.

[71] Kwang-Kyo Oh, Myoung-Chul Park, and Hyo-Sung Ahn. A survey of multi-agent formation control. *Automatica*, 53:424–440, March 2015.

[72] Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 2007.

[73] Liviu Panait and Sean Luke. Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems*, 11(3):387–434, November 2005. Publisher: Springer.

[74] Michal Pechoucek, Vladimir Marik, and Olga Stepankova. Towards Reducing Communication Traffic In Multi-Agent Systems. *Journal of Applied Systems Science: Special Issue*, 2(1):211–245, 2001.

[75] Kejun Qian, Chengke Zhou, Malcolm Allan, and Yue Yuan. Modeling of Load Demand Due to EV Battery Charging in Distribution Systems. *IEEE Transactions on Power Systems*, 26(2):802–810, May 2011. Conference Name: IEEE Transactions on Power Systems.

[76] Kejun Qian, Chengke Zhou, Malcolm Allan, and Yue Yuan. Modeling of

Load Demand Due to EV Battery Charging in Distribution Systems. *IEEE Transactions on Power Systems*, 26(2):802–810, May 2011. Conference Name: IEEE Transactions on Power Systems.

[77] Jiahu Qin, Qichao Ma, Yang Shi, and Long Wang. Recent advances in consensus of multi-agent systems: A brief survey. *IEEE Transactions on Industrial Electronics*, 64(6):4972–4983, 2017. Publisher: Institute of Electrical and Electronics Engineers Inc.

[78] Anil V Rao. A survey of numerical methods for optimal control. *Advances in the Astronautical Sciences*, 135(1):497–528, 2009.

[79] Craig W Reynolds. Flocks, herds and schools: A distributed behavioral model. In *ACM SIGGRAPH computer graphics*, volume 21, pages 25–34. ACM, 1987.

[80] Yara Rizk, Mariette Awad, and Edward W Tunstel. Decision making in multiagent systems: A survey. *IEEE Transactions on Cognitive and Developmental Systems*, 10(3):514–529, 2018.

[81] Subir Kumar Sarkar, Tiptur Gangaraju Basavaraju, and C Puttamadappa. *Ad hoc mobile wireless networks: principles, protocols, and applications*. CRC Press, 2016.

[82] Wenjing Shuai, Patrick Maillé, and Alexander Pelov. Charging Electric Vehicles in the Smart City: A Survey of Economy-Driven Approaches. *IEEE Transactions on Intelligent Transportation Systems*, 17(8):2089–2106, August 2016. Conference Name: IEEE Transactions on Intelligent Transportation Systems.

[83] Christoph Studer and Erik G Larsson. Par-aware large-scale multi-user mimo-ofdm downlink. *IEEE Journal on Selected Areas in Communications*, 31(2):303–313, 2013.

[84] Jian Sun and Zhanshan Wang. Event-triggered consensus control of high-order multi-agent systems with arbitrary switching topologies via model partitioning approach. *Neurocomputing*, 413:14–22, November 2020.

[85] Mohammad Amin Tajeddini, Hamed Kebriaei, and Luigi Glielmo. Robust decentralised mean field control in leader following multi-agent systems. *IET Control Theory & Applications*, 11(16):2707–2715, 2017.

[86] Ming Tang and Huchang Liao. From conventional group decision making to large-scale group decision making: what are the challenges and how to meet them in big data era? a state-of-the-art survey. *Omega*, page 102141, 2019.

[87] Omkar Thakoor, Jugal Garg, and Rakesh Nagi. Multiagent UAV Routing: A Game Theory Analysis With Tight Price of Anarchy Bounds. *IEEE Transactions on Automation Science and Engineering*, 17(1):100–116, January 2020. Conference Name: IEEE Transactions on Automation Science and Engineering.

[88] Xiaoyang Tong, Xiaoru Wang, Rui Wang, Fei Huang, Xueyuan Dong, Kenneth M. Hopkinson, and Gongyi Song. The Study of a Regional Decentralized Peer-to-Peer Negotiation-Based Wide-Area Backup Protection Multi-Agent System. *IEEE Transactions on Smart Grid*, 4(2):1197–1206, June 2013. Conference Name: IEEE Transactions on Smart Grid.

[89] Jacopo Torriti. Price-based demand side management: Assessing the impacts of time-of-use tariffs on residential electricity demand and peak shifting in Northern Italy. *Energy*, 44(1):576–583, August 2012.

[90] Vladimir Turetsky and Tal Shima. Target Evasion from a Missile Performing Multiple Switches in Guidance Law. *Journal of Guidance, Control, and Dynamics*, 39(10):2364–2373, 2016. Publisher: American Institute of Aeronautics and Astronautics _eprint: https://doi.org/10.2514/1.G000461.

[91] Kyriakos Vamvoudakis, Draguna Vrabie, and Frank Lewis. Online policy iteration based algorithms to solve the continuous- time infinite horizon optimal control problem. In *2009 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning, ADPRL 2009 - Proceedings*, pages 36–41, 2009.

[92] Kyriakos G. Vamvoudakis and Frank L. Lewis. Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica*, 46(5):878–888, May 2010.

[93] Kyriakos G. Vamvoudakis and Frank L. Lewis. Multi-player non-zero-sum games: Online adaptive learning solution of coupled Hamilton–Jacobi equations. *Automatica*, 47(8):1556–1569, August 2011. Publisher: Pergamon.

[94] Kyriakos G. Vamvoudakis, Frank L. Lewis, and Warren E. Dixon. Open-loop Stackelberg learning solution for hierarchical control problems. *International*

*Journal of Adaptive Control and Signal Processing*, 33(2):285–299, 2019. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/acs.2831.

[95] R. Vidal, S. Rashid, C. Sharp, O. Shakernia, Jin Kim, and S. Sastry. Pursuit-evasion games with unmanned ground and aerial vehicles. In *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, volume 3, pages 2948–2955 vol.3, May 2001. ISSN: 1050-4729.

[96] Draguna Vrabie, Kyriakos Vamvoudakis, and Frank Lewis. Adaptive optimal controllers based on generalized policy iteration in a continuous-time framework. In *2009 17th Mediterranean Conference on Control and Automation*, pages 1402–1409. IEEE, 2009.

[97] Draguna Vrabie, Kyriakos G. Vamvoudakis, and Frank L. Lewis. *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles*. The Institution of Engineering and Technology, London, November 2012.

[98] Qianyao Wang, Kexin Liu, Xiong Wang, Lulu Wu, and Jinhu Lü. Leader-following consensus of multi-agent systems under antagonistic networks. *Neurocomputing*, 413:339–347, November 2020.

[99] Xiaofei Wang, Yuhua Zhang, Victor C. M. Leung, Nadra Guizani, and Tianpeng Jiang. D2D Big Data: Content Deliveries over Wireless Device-to-Device Sharing in Large-Scale Mobile Networks. *IEEE Wireless Communications*, 25(1):32–38, February 2018.

[100] Xiaofei Wang, Yuhua Zhang, Victor CM Leung, Nadra Guizani, and Tianpeng Jiang. D2d big data: Content deliveries over wireless device-to-device sharing in large-scale mobile networks. *IEEE Wireless Communications*, 25(1):32–38, 2018.

[101] Yu Wang, Tung-Lam Nguyen, Yan Xu, Quoc-Tuan Tran, and Raphael Caire. Peer-to-Peer Control for Networked Microgrids: Multi-Layer and Multi-Agent Architecture Design. *IEEE Transactions on Smart Grid*, pages 1–1, 2020. Conference Name: IEEE Transactions on Smart Grid.

[102] Yuanda Wang, Lu Dong, and Changyin Sun. Cooperative control for multi-player pursuit-evasion games with reinforcement learning. *Neurocomputing*, 412:101–114, October 2020.

[103] Guo-Xing Wen, CL Philip Chen, Yan-Jun Liu, and Zhi Liu. Neural-network-

based adaptive leader-following consensus control for second-order nonlinear multi-agent systems. *IET Control Theory & Applications*, 9(13):1927–1934, 2015.

[104] Hongming Yang, Songping Yang, Yan Xu, Erbao Cao, Mingyong Lai, and Zhaoyang Dong. Electric Vehicle Route Optimization Considering Time-of-Use Electricity Price by Learnable Partheno-Genetic Algorithm. *IEEE Transactions on Smart Grid*, 6(2):657–666, March 2015. Conference Name: IEEE Transactions on Smart Grid.

[105] Peng Yang, Gongguo Tang, and Arye Nehorai. A game-theoretic approach for optimal time-of-use electricity pricing. *IEEE Transactions on Power Systems*, 28(2):884–892, May 2013. Conference Name: IEEE Transactions on Power Systems.

[106] Y. Yang, Y. Li, D. Yue, Y. Tian, and X. Ding. Distributed secure consensus control with event-triggering for multiagent systems under dos attacks. *IEEE Transactions on Cybernetics*, pages 1–13, 2020.

[107] Yaodong Yang, Rui Luo, Minne Li, Ming Zhou, Weinan Zhang, and Jun Wang. Mean field multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 5571–5580. PMLR, 2018.

[108] Dajie Yao, Chunxia Dou, Dong Yue, Nan Zhao, and Tingjun Zhang. Event-triggered adaptive consensus tracking control for nonlinear switching multi-agent systems. *Neurocomputing*, 415:157–164, November 2020.

[109] Jens Zander. Distributed cochannel interference control in cellular radio systems. *IEEE transactions on vehicular Technology*, 41(3):305–311, 1992.

[110] H. Zhang, T. Feng, G. Yang, and H. Liang. Distributed cooperative optimal control for multiagent systems on directed graphs: An inverse optimal approach. *IEEE Transactions on Cybernetics*, 45(7):1315–1326, 2015.

[111] Zhen Zhang, Dongbin Zhao, Junwei Gao, Dongqing Wang, and Yujie Dai. Fmrq—a multiagent reinforcement learning algorithm for fully cooperative tasks. *IEEE transactions on cybernetics*, 47(6):1367–1379, 2016.

[112] Zhengquan Zhang, Yue Xiao, Zheng Ma, Ming Xiao, Zhiguo Ding, Xianfu Lei, George K. Karagiannidis, and Pingzhi Fan. 6G Wireless Networks: Vision, Requirements, Architecture, and Key Technologies. *IEEE Vehicular Tech-*

*nology Magazine*, 14(3):28–41, September 2019. Conference Name: IEEE Vehicular Technology Magazine.

[113] Wanbing Zhao, Hao Liu, and Frank L Lewis. Robust formation control for cooperative underactuated quadrotors via reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[114] Z. Zhou and H. Xu. A Novel Mean-Field-Game-Type Optimal Control for Very Large-Scale Multiagent Systems. *IEEE Transactions on Cybernetics*, pages 1–1, 2020.

[115] Zejian Zhou, Lijun Qian, and Hao Xu. Intelligent Decentralized Dynamic Power Allocation in MANET at Tactical Edge based on Mean-Field Game Theory. In *MILCOM 2019 - 2019 IEEE Military Communications Conference (MILCOM)*, pages 604–609, November 2019. ISSN: 2155-7586.

[116] Zejian Zhou and Hao Xu. Decentralized optimal charging control for large scale of electrical vehicles: A mean-field game approach with reinforcement learning. *IEEE Transactions on Smart Grid*. under review.

[117] Zejian Zhou and Hao Xu. Decentralized optimal large scale multi-player pursuit-evasion strategies: A mean field game approach with reinforcement learning. *Neurocomputing*. in press.

[118] Zejian Zhou and Hao Xu. Large-scale multi-agent system tracking control using mean field games. *IEEE Transactions on Neural Networks and Learning Systems*. in press.

[119] Zejian Zhou and Hao Xu. Decentralized Adaptive Optimal Control for Massive Multi-agent Systems Using Mean Field Game with Self-Organizing Neural Networks. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 1225–1230, December 2019. ISSN: 2576-2370.

[120] Zejian Zhou and Hao Xu. Mean field game and decentralized intelligent adaptive pursuit evasion strategy for massive multi-agent system under uncertain environment. In *2020 American Control Conference (ACC)*, pages 5382–5387. IEEE, 2020.

[121] Ziming Zhu, Sangarapillai Lambotharan, Woon Hau Chin, and Zhong Fan. A Mean Field Game Theoretic Approach to Electric Vehicles Charging. *IEEE Access*, 4:3501–3510, 2016. Conference Name: IEEE Access.

[122] Lei Zou, Zidong Wang, Huijun Gao, and Fuad E. Alsaadi. Finite-Horizon $\mathcal H_infty $ Consensus Control of Time-Varying Multiagent Systems With Stochastic Communication Protocol. *IEEE Transactions on Cybernetics*, 47(8):1830–1840, August 2017. Conference Name: IEEE Transactions on Cybernetics.