Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Biomimetic Optimal Tracking Control using Mean Field Games and Spiking Neural Networks

Zejian Zhou*, M. Sami Fadali*, Hao Xu*

***University of Nevada, Reno, Reno, NV 89557**
USA (Tel: 775-682-6873; e-mail: zejianz@nevada.unr.edu; {fadali;haoxu}@ unr.edu).

Abstract: This paper investigates decentralized optimal tracking control for multi-agent systems (MAS)s with a large population. Unlike conventional decentralized control, two major challenges must be addressed when the population size of the MAS is large: the "curse of dimensionality" and environmental uncertainties. The paper develops a novel online learning decentralized adaptive optimal control strategy to address these challenges by combining the emerging Mean Field Games (MFG) theory with a novel Biomimetic Actor-Critic-Mass (B-ACM) learning algorithm. Mean-field control is developed as a decentralized optimal controller that can effectively reduce the computational complexity and the communication effort. A Biomimetic neural network that mimics the human brain, which is much more efficient than traditional Artificial Neural Networks (ANNs), is designed using Spiking Neural Networks (SNN)s. The information is encoded into a sparse spikes vector similar to the human brain. The SNN technique and mean-field control are merged into one unified framework, B-ACM. The B-ACM includes three regions of neurons in coordination with mean-field control: 1) ***Reward*** region to approximate the optimal cost function, 2) ***MAS Population Estimation*** region to predict the effects from other agents, and 3) ***Action*** region to compute the optimal control. Moreover, the paper introduces a novel SNN weight update law based on gradient descent. The effectiveness of the proposed scheme is validated through numerical simulations.

Keywords: Biomimetic control, Intelligent control, Stochastic systems, optimal control, neural networks

1. INTRODUCTION

Multi-Agent Systems (MAS)s are increasing in popularity because of the many advantages of having a large population of agents. However, two major challenges are restricting the effectiveness of most existing MAS tracking control algorithms. The first challenge is the "curse of dimensionality", which causes an explosion of computational effort when a large-scale system is considered. In tracking control and "leader-follower" algorithms, such as (Li *et al.*, 2015), the dimension of the problem and the complexity of computing the optimal control increase drastically with the number of agents. The second challenge is the harsh environment where high-quality wireless communication is unavailable.

To overcome these challenges, several methods have been proposed including (Peng et al., 2018; Cui et al., 2019). However, theses decentralized algorithms can neither guarantee system optimality nor completely handle low-quality communication or inaccurate observation. Recently, a novel Mean Field Game (MFG) theory was proposed to compute the population's states' density directly through the Fokker-Planck-Kolmogorov (FPK) equation, a partial differential equation that only needs local information and the initial agent distribution (Caines, Huang and Malhamé, 2018).

The theoretical analysis of MFG-based optimal control integrates the FPK equation into the optimal control problem (Guéant, Lasry and Lions, 2011; Caines, Huang and Malhamé, 2018). The optimal control policy and optimal cost can be obtained by solving the Hamilton-Jacobi-Bellman (HJB) equation (Lewis and Vrabie, 2009). (Caines, Huang and Malhamé, 2018) developed the coupled HJB-FPK equation, also called the Mean Field equations, by replacing the effect of the MAS population in the Hamiltonian with the probability density function (PDF). Although this provides effective control, the Mean Field equation is nearly impossible to solve analytically because the FPK and HJB are two coupled high dimensional PDEs.

In recent years, Adaptive Dynamic Programming (ADP) has appeared as a promising technique to solve the HJB equation (Vamvoudakis and Lewis, 2010). We extend the ADP framework to a novel Actor-Critic-Mass learning structure by introducing an extra Mass Neural Network to approximate the solution of the FPK equation online. We replace the conventional ANNs with Spiking Neural Networks (SNNs) inspired by the human brain to obtain a Biomimetic ADP that we christen B-ACM. Human brains can encode huge amount of information using small populations of spikes and consume significantly less energy than ANNs (Wolfe, Houweling and Brecht, 2010).

SNNs are the third generation of neural networks (Bing et al., 2018), that simulate the elementary processes in the human brain according to the functional similarity to biological brains. Recent studies in neuromorphic computing have shown that SNNs are easily implemented in hardware and much more efficient than traditional neural networks (Das, Schulze and Ganguly, 2018; Bouvier *et al.*, 2019). Moreover, SNNs are popular in many reinforcement learning structures, e.g., (Frémaux, Sprekeler, and Gerstner, 2013; Friedrich and Lengyel,

2016). However, none of these methods is designed for multi-agent reinforcement learning or massive multi-agent problems. This paper proposes a massive multi-agent decentralized optimal tracking control where the SNNs adaptively approximate the optimal control using a novel Actor-Critic-Mass structure.

The contribution of this paper can be summarized as:

1) It proposes a novel online Biomimetic Actor-Critic-Mass (B-ACM) algorithm for massive MAS decentralized tracking control to overcome the "Curse of Dimensionality" and handle a harsh environment. This is accomplished by integrating the Mean Field Games (MFG) theory with Adaptive Dynamic Programming (ADP).

2) It uses a novel Spiking Neural Network (SNN) in the ACM. Because the SNN is easily implemented with RC circuits (Das, Schulze and Ganguly, 2018; Bouvier *et al.*, 2019), the proposed B-ACM can be applied on massive robotic systems without the need for GPUs for online learning that other neural networks require.

2. BACKGROUD AND PROBLEM FORMULATION

Consider N agents with stochastic dynamics defined as:

$$dx_i = [f_s(x_i) + g_s(x_i)u_i]dt + \Sigma_i dw_i \tag{1}$$

where $x_i \in \mathbb{R}^l$ is the state containing the i^{th} agent's position, $u_i \in \mathbb{R}^l$ is the control input, $w_i \in \mathbb{R}^l$ denotes a set of independent Wiener processes representing environmental noise, and Σ_i is a diagonal matrix of diffusion coefficients. The functions $f_S(x_i)$ and $g_s(x_i)$ are the intrinsic dynamic functions of all agents. Decentralized tracking control is needed to force the large MAS to follow a given reference trajectory.

Let $x_r(t)$ be the desired reference trajectory, the tracking error for each agent can be defined as $e_i(t) = x_i(t) - x_r(t)$. Thus, the tracking error dynamics can be derived as:

$$de_i = [f(e_i) + g(e_i)u_i] + \Sigma_i dw_i \tag{2}$$

where $f(e_i) = f_s(e_i + x_r) - dx_r/dt$, and $g(e_i) = g_s(e_i + x_r)$ represent the tracking error dynamics function.

The objective of each agent is to follow the reference trajectory while considering the effect from other agents, that is to minimize a cost function

$$V_i(e_i, m) = \mathbb{E}\left\{\int_0^\infty [L(e_i, u_i) + \Phi(m, e_i)]dt\right\} \tag{3}$$

where $m = m(e_i, t), i = 1,2, \dots, N$ is the nonstationary PDF of the tracking error for the i^{th} agent and has the same form for all agents, $\Phi(m, e_i)$ is the mean field coupling function that represents the influence of other agents, $Q \geq 0$, and $R > 0$ are weight matrices of compatible dimensions. Note that the value of the cost function depends on the tracking error so we assume that tracking remains sufficiently accurate for a bounded cost function.

The optimal control is obtained by solving the HJB equation (Lewis and Vrabie, 2009)

$$\Phi(e_i, m) = -\partial_t V_i^*(e_i, m, t) - \frac{\sigma_i^2}{2}\Delta V_i^*(e_i, m, t) + H[e_i, \partial_e V_t^*(e_i, m, t)] \tag{4}$$

where $\partial_t V_i^*(e_i, m, t) = \frac{\partial}{\partial_t} V_i^*(e_i, m, t)$, $\Phi(m_i, e_i)$ is the mean field coupling function that describes the coupling effects of agent i and all other agents, the function $H(\cdot)$ is the Hamiltonian defined as

$$H[e_i, \partial_e V_i(e_i, m, t)] = L(e_i, u_i) + \Phi(m, e_i) + \partial_e V_i(e_i, m, t)^T[f(e_i) + g(e_i)u_i] \tag{5}$$

Then, similar to (Lewis and Vrabie, 2009), the optimal control for each individual agent is

$$u_i^*(e_i) = -\frac{1}{2}R^{-1}g^T(e_i)\partial_e V_i^*(e_i, m, t) \tag{6}$$

Since individual agents minimize their own cost to obtain the decentralized optimal control u_i^*, the problem can be considered as a nonzero-sum stochastic differential game. Therefore, there exists a Nash equilibrium $\{u_1^*, \dots, u_N^*\}$ such that individual agent cost is optimal, i.e., $V_i(u_i; u_{-i}) \geq V_i(u_i^*; u_{-i})$, where u_{-i} represents the control input of all agents other than i (Carmona and Delarue, 2013).

To solve the HJB equation (4), the population's tracking error PDF distribution $m(e_i, t)$ is needed. Recall to MFG (Caines, Huang and Malhamé, 2018), the $m(e_i, t)$ can be attained by solving the Fokker-Planck-Kolmogorov (FPK) equation based on the "law of large numbers", i.e.,

$$\partial_t m(e_i, t) - \frac{\sigma_i^2}{2}\Delta m(e_i, t) - \text{div}\{mD_p H[e_i, \partial_e V_i^*(e_i, m, t)]\} = 0 \tag{7}$$

where $D_p(.)$ denotes $\partial(.)/\partial_e u_i^*(e_i)$.

Definition 1: (ϵ_N-Nash equilibrium)(Nourian and Peter E. Caines, 2013) Given $\epsilon_N > 0$, the admissible control laws $\{u_1^*, \dots, u_N^*\}, u_i \in U_i, i = 1, \dots, N$, for N agents generate an ϵ_N-Nash equilibrium with respect to the cost V_i, $1 \leq i \leq N$, if $V_i(u_i^*; u_{-i}) - \epsilon_N \leq \inf_{u_i \in U_i} V_i(u_i; u_{-i}), i = 1, \dots, N$.

Theorem 1: (ϵ_N-Nash equilibrium)(Nourian and Peter E Caines, 2013) If there exists a unique solution to the coupled HJB-FPK equation system, then the optimal control laws $\{u_1^*, \dots, u_N^*\}$ generate an $\boldsymbol{\epsilon_N}$-Nash equilibrium such that $\lim_{N\to\infty} \epsilon_N = 0$.

Remark 1: Theorem 1 requires the solution of the couple HJB-FPK equations to uniquely exist. The solution exist and is unique under mild conditions (Guéant, Lasry and Lions, 2011; Caines, Huang and Malhamé, 2018). To obtain the optimal design, the coupled HJB-FPK equations must be solved simultaneously. However, the HJB equation (4) and the FPK equation (7) are nonlinear Partial Differential Equations (PDEs) whose solutions are difficult to obtain

analytically. In this paper, a novel Spiking Neural Network (SNN) Biomimetic Actor-Critic-Mass (B-ACM) learning algorithm is developed to solve the coupled HJB-FPK equations online.

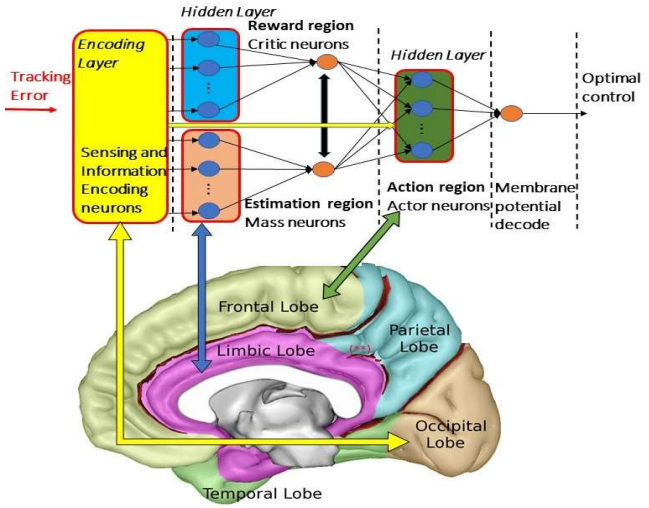

Fig. 1. An illustration of the proposed Biomimetic ACM algorithm based on SNN. The corresponding regions in human brain are marked on the cerebral cortex image.

3. DECENTRALIZED ADAPTIVE OPTIMAL CONTROL

In this section, the proposed SNN-based B-ACM controller is developed. The structure of the B-ACM controller, as well as the corresponding brain regions, are shown in Figure 1. The controller includes one set of sensing neurons and three regions: 1) the sensing neurons sense the environment and encode the information into spikes, which is similar to the occipital lobe, 2) the reward region and population estimation region approximate the optimal cost function and the population's tracking error PDF, which serves the same purpose as the limbic lobe, 3) the action region calculates the control input, which is similar to the frontal lobe. Next, the B-ACM controller is discussed in detail.

3.1 Spiking Neural Network Model Design

The Spike Response Model (SRM) neuron model (Gerstner *et al.*, 2014) requires minimal computational resources without degrading learning performance compared to other models such as (Nagumo, Arimoto and Yoshizawa, 1962; Izhikevich, 2003), etc. We first define the activation function for the encoding neurons that encode the inputs into the spike times for the output spikes of the i^{th} agent's encoding neuron:

$$\begin{cases} \boldsymbol{t}_{v,i} = l_v(e_i, m) \\ \boldsymbol{t}_{m,i} = l_m(e_i, t) \\ \boldsymbol{t}_{u,i} = l_u(e_i, m) \end{cases} \tag{8}$$

where $\boldsymbol{t}_{v,i}$ is the sequence of spike times (spike train) for the critic neurons in the reward region, $\boldsymbol{t}_{m,i}$ is the spike time for the mass neurons in the population estimation region, and $\boldsymbol{t}_{u,i}$ is the spike time for the actor neurons in the action region.

The spike train is written in terms of the Dirac delta function as:

$$S(t) = \sum_k \delta(t - t^k) \tag{9}$$

with $t^k \in \boldsymbol{t}_{\cdot,i}$ are the firing times. The spike trains from a set of neurons $b_n, n = 1,2,\dots$ are transmitted to another connected neuron, say a, in the form of synaptic input current that charges the neuron a by increasing its membrane potential μ_a. The neuron a fires a spike when the membrane potential exceeds a threshold value μ^{th}. Then the membrane voltage returns to the resting potential μ^r. Thus, the SRM model can be written as a function that maps the input spike trains and output spike trains to the membrane potential

$$\mu_a(t) = \sum_{b_n} w_{ab_n} \left(\epsilon_a * S_{b_n}(t)\right) + (\eta_a * S_a(t)) \tag{10}$$

where w_{ab_n} is the synaptic weight of the connection between neuron a and b_i, $S_a(t)$ is the output spike train fired by neuron a, $S_{b_n}(t)$ is the input spike train from neuron b_n, η_a is a convolution kernel that models the effect of output spike dynamics, the kernels $\epsilon_a(t), \eta_a$ describe the postsynaptic potential caused by the input spike train and output spike train, respectively.

3.2 SNN-Based B-ACM Algorithm

In this section, the actor-critic reinforcement learning algorithm (Lewis and Vrabie, 2009) is adopted and extended to the Spiking Neural Network (SNN) Biomimetic Actor-Critic-Mass (B-ACM) algorithm for massive MAS decentralized tracking control. To obtain the decentralized optimal tracking control, the optimal cost function and the PDF must first be obtained. Therefore, the reward region and population estimation region in Fig. 1 are constructed to approximate the solution of the HJB equation (4) as well as FPK equation (7). The postsynaptic potentials of the neurons in these two regions are then sent to the action region to approximate the optimal control input. The mathematical representation of this process is demonstrated as follows.

The cost function, PDF, and decentralized optimal control can be approximated as functions of the postsynaptic membrane potential of the output neurons in the corresponding regions:

$$\begin{cases} V_i^*(e_i, m, t) = \phi_{v,i}\left(\mu_{V,i}(t)\right) + V_{0,i} \\ u_i^*(e_i, m, t) = \phi_{u,i}\left(\mu_{u,i}(t)\right) + u_{0,i} \\ m_i^*(e_i, t) = \phi_{m,i}\left(\mu_{m,i}(t)\right) + m_{0,i} \end{cases} \quad (11)$$

where $\mu_{V,i}(t)$, $\mu_{u,i}(t)$, and $\mu_{m,i}(t)$ are the postsynaptic potentials of the output neurons of the Critic NN, the Actor NN, and the Mass NN, respectively, $V_{0,i}$, $u_{0,i}$, and $m_{0,i}$ are the membrane potentials in the absence of an input spike train.

Equation (10) shows that the synaptic weights w_{ab_n} for all neuron connections, which are unknown, are required to calculate the postsynaptic membrane potential. We estimate the synaptic weights and use the estimates to obtain estimates of the membrane potentials. The estimates of the membrane potentials can be derived by substituting (8) and (9)into (10) to obtain

$$\begin{cases} \hat{\mu}_{V,i}(e_i, \hat{m}_i) = \sum_{b_n} \hat{w}_{V,ib_n}\left(\epsilon_V * l_{V,i}(e_i, \hat{m}_i)\right) + (\eta_v * S_{V,i}(t)) \\ \hat{\mu}_{m,i}^p(e_i, t) = \sum_{b_n} \hat{w}_{m,ib_n}\left(\epsilon_m * l_{m,i}(e_i, \mathrm{t})\right) + (\eta_m * S_{m,i}(t)) \\ \hat{\mu}_{u,i}^p(e_i, \hat{m}_i) = \sum_{b_n} \hat{w}_{u,ib_n}\left(\epsilon_u * l_{u,i}(e_i, \hat{m}_i)\right) + (\eta_u * S_{u,i}(t)) \end{cases} \quad (12)$$

where $\hat{w}_{V,\mathrm{i}}$, $\hat{w}_{m,i}$, $\hat{w}_{u,i}$ are the estimates of the synaptic weights' vectors. Next, we rewrite (11) using the estimates of the membrane potentials of the output neurons as

$$\begin{cases} \hat{V}_i(e_i, \hat{m}_i, t) = \phi_{v,i}\left(\hat{\mu}_{V,i}(e_i, \hat{m}_i)\right) + V_{0,i} \\ \hat{u}_i(e_i, \hat{m}_i, t) = \phi_{u,i}\left(\hat{\mu}_{u,i}^p(e_i, \hat{m}_i)\right) + u_{0,i} \\ \hat{m}_i(e_i, t) = \phi_{m,i}\left(\hat{\mu}_{m,i}^p(e_i, t)\right) + m_{0,i} \end{cases} \quad (13)$$

Substituting (13) and (12) into (4), (7), and (6) gives the residual errors

$$e_{v,i} = \Phi_i(\hat{m}_i, e_i) + \partial_t\phi_{v,i}(e_i, \hat{m}_i) + \frac{\sigma_i^2}{2}\Delta\phi_{v,i}(e_i, \hat{m}_i) - H[e_i, \partial_e\phi_{v,i}(e_i, \hat{m}_i)] \quad (14)$$

$$e_{m,i} = \partial_t\phi_{m,i}(e_i, t) - \frac{\sigma_i^2}{2}\Delta\phi_{m,i}(e_i, t) - \mathrm{div}\{\phi_{m,i}(e_i, t)D_pH[e_i, \partial_e\hat{V}_i(e_i, \hat{m}_i)]\} \quad (15)$$

$$e_{u,i} = \phi_{u,i}(e_i, \hat{m}_i(e_i, t) + \frac{1}{2}R_i^{-1}(e_i)\partial_e\phi_{V,i}(e_i, \hat{m}_i) \quad (16)$$

According to the gradient descent algorithm, the B-ACM neurons' update law can be obtained as

Critic neurons in reward region:

$$\dot{\hat{W}}_{v,i} = -\alpha_{h,i}\nabla_{W_{v,i}}e_{v,i}(e_i, \hat{m}_i) \quad (17)$$

Mass neurons in reward region:

$$\dot{\hat{W}}_{m,i} = -\alpha_{m,i}\nabla_{W_{m,i}}e_{m,i}(e_i, t) \quad (18)$$

Actor neurons in reward region:

$$\dot{\hat{W}}_{u,i} = -\alpha_{u,i}\nabla_{W_{u,i}}e_{u,i}(e_i, \hat{m}_i) \quad (19)$$

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

$\dot{\hat{W}}_{v,i} = [\hat{w}_{V,ib1} \quad \hat{w}_{V,ib2} \quad \hat{w}_{V,ib3} \quad \ldots]^T$

$\hat{w}_{v,ib_n}$ represents weight between the n^{th} neuron in the hidden layer and the output neuron the i^{th} agent. $\hat{w}_{m,ib_n}$, $\hat{w}_{u,ib_n}$ have similar definitions.

Finally, the closed-loop stability analysis is given as follows.

Theorem 1. (Closed-loop Stability) Given an admissible initial control input, actor, critic, and a Mass NN whose synaptic weights are selected within a compact set and tuned with the tuning laws of (19), (17), and (18). There exist constants $\boldsymbol{\alpha_{h,i} > 0}$, $\boldsymbol{\alpha_{m,i} > 0}$, and $\boldsymbol{\alpha_{u,i} > 0}$ such that the system tracking error $\boldsymbol{e_i}$, actor, critic, and mass neurons' synaptic weights estimation errors, $\dot{\hat{\boldsymbol{W}}}_{\boldsymbol{v,i}}$, $\dot{\hat{\boldsymbol{W}}}_{\boldsymbol{m,i}}$, and $\dot{\hat{\boldsymbol{W}}}_{\boldsymbol{u,i}}$ are all uniformly ultimately bounded (UUB). In addition, the estimated cost function, PMF, and control inputs are all UUB.

Proof. Omitted due to page limit.

4. SIMULATIONS

In this section, the proposed decentralized adaptive optimal Mean Field control is evaluated under the noised environment. The map we use is a 2-D map. A total of 1000 agents were employed, with initial velocities set to zero, and positions randomly distributed on the map. We designed a search and rescue mission on campus, where the planned reference trajectory is : $x_r(t) = [0.01t \ \sin(0.1t)]^T$.

Next, the intrinsic dynamics functions in (1) for all agents are selected as

$$f_s(x) = \begin{bmatrix} -x_1 + x_2 \\ -\frac{1}{2}\cos(x_1)^2 - \frac{1}{2}x_2 \end{bmatrix}, g_s(x) = \begin{bmatrix} 0 \\ \cos x_1 \end{bmatrix}$$

where $x \in \mathbb{R}^2$ represents the agent's position.

The parameters are selected as $R = Q = I_2$, $\alpha_h = 1 \times 10^{-3}$, $\alpha_m = 1 \times 10^{-6}$, $\alpha_u = 1 \times 10^{-4}$ and the total simulation time is $140s$.

The Mean Field coupling function in cost function (3) is selected as the distance between the population mean tracking error and the agent's current tracking error, i.e.,

$$\Phi(m_i, e_i, t) = \|e_i(t) - \mathbb{E}\{m(t)\}\|^2 \quad (20)$$

The agents' initial position distribution follows a 2-variant normal distribution where the mean vector is $[1 \quad 1]^T$ and the covariance matrix is the identity matrix I_2.

To estimate the solution of the HJB equation V_i^*, the solution of the FPK equation m_i, and the MF control input u_i^*, the two-layer SNNs with critic neurons, actor neurons, and mass neurons are constructed. The functions $\phi_v(\cdot)$, $\phi_m(\cdot)$, and $\phi_u(\cdot)$ of (11) and (13) are selected as linear functions $\phi(z) = z$. The terms V_0, u_0, and m_0 are set to zero for all agents.

Next, the information encoding functions (i.e., the coding neurons) for the critic neurons are defined as

$$l_{v,j}(e_i, m_i) = \text{tansig}(Z_j) + t, j = 1,2,3,\cdots,\frac{(M+3)!}{M!\,3!} \qquad (21)$$

where Z_j represents the j^{th} term of the expansion of the polynomial $\sum_{\beta=1}^{M}(e_1 + e_2 + m_1 + m_2)^{\beta}$, $M = 4$ for actor neurons and $M = 5$ for critic neurons.

The coding neurons for population estimation region are

$$l_{m,j}(e_i, t) = \text{tansig}(Z_j) + t, \quad j = 1,2,3,\cdots,461 \qquad (22)$$

where Z_j is the terms of the expansion of the polynomial $\sum_{\beta=1}^{M}(e_1 + e_2 + m_1 + m_2 + t)^{\beta}$ with $M = 6$.

Finally, the number of neurons in the hidden layer for the critic neurons, actor neurons and mass neurons are the same as the corresponding coding neurons. The synaptic weights between neurons in the encoding layer and hidden layer are set equal to 1 while the weights between the neurons in the hidden layer and output layer are randomly initialized. The ϵ functions in these neurons, (see (12)), are all selected as the linear function $\epsilon(t) = t$. Similarly to (Zenke and Ganguli, 2018) and without loss of generality, the second term in (12) is drooped for simplicity because it only contributes a small correction to the membrane potential. To increase all agents' exploration rate, random noise is injected into their control input from the beginning to 50 seconds. The random noise can accelerate learning and avoid local optimal solution.

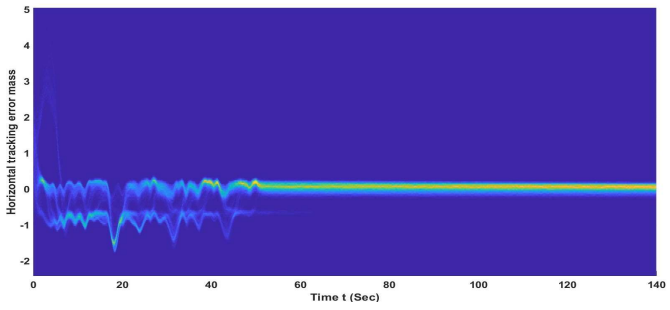

Fig. 2. The tracking error PDF plot of vertical tracking error. The blue region represents lower density while the yellow region represents high density.

The simulation results are shown in Figs. 2-4. First, the tracking error PDF with respect to time is plotted in the Fig. 2 and Fig. 3. The purple represents the least common error values and the yellow represents the most common error values. The plots show that the initial tracking error is high and randomly distributed. However, after 53 seconds, the agents' mean tracking errors are bounded near zero and the variance of tracking error distribution decreases near zero. This also shows that the system can track the reference trajectory.

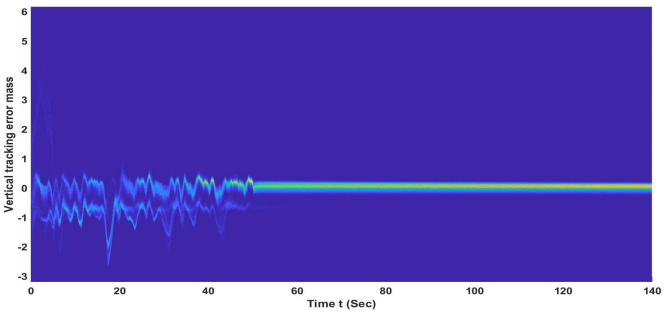

Fig. 3. The tracking error PDF plot of horizontal tracking error. The blue region represents lower density while the yellow region represents high density.

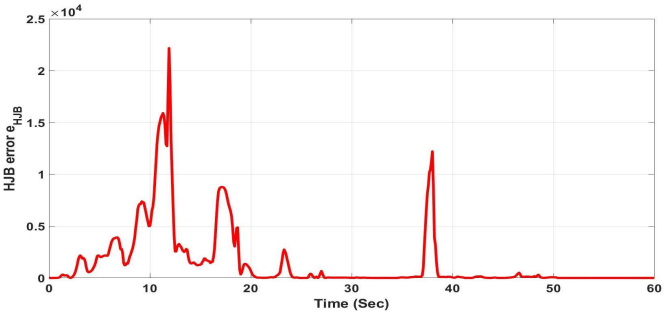

Fig. 4. The time evolution of HJB equation error for agent 1.

Then, the performance of the SNNs is studied. The errors of the HJB equation, i.e., e_{HJBi} in (14) is plotted to show the convergence of the synaptic weights. For brevity, and because the plots are similar for all agents, we only plot the HJB equation error for agent 1 in Fig. 4 where the error is bounded near zero after 50 seconds.

The convergence of the tracking error and SNN learning error, i.e. the HJB equation error, indicates that the online B-ACM algorithm effectively approximates the cost function, PDF, and optimal control input respectively. The simulation is conducted when each agent has limited observation capability due to the harsh environment. Each agent only needs to solve two 2-dimensional PDEs instead of a 1000-dimensional HJB equation. Thus, the computational complexity is reduced significantly.

5. CONCLUSIONS

In this paper, a novel Actor-Critic-Mass (ACM) learning structure has been developed along with Biomimetic

neural networks, i.e. Spiking Neural Networks (SNNs). The proposed algorithm yields a decentralized optimal tracking control for large scale multi-agent systems through approximating the solution of coupled HJB-FPK equation in real-time. Moreover, the developed scheme can effectively reduce the computational complexity due to its decentralized structure. Furthermore, a novel SNN mechanism has been designed for approximation. The SNN contains three critical regions that are the reward region, the PDF population estimation region and the action region. The three regions can efficiently approximate the solution of HJB, FPK, and decentralized optimal control policy respectively. Finally, a series of numerical simulations demonstrate the effectiveness and efficiency of the proposed SNN based B-ACM algorithm. In future work, the proposed design will be implemented and evaluated through a real-time massive MAS testbed at University of Nevada, Reno with the support of the U.S. Federal Aviation Administration (FAA).

REFERENCES

Bing, Z. *et al.* (2018) 'A Survey of Robotics Control Based on Learning-Inspired Spiking Neural Networks', *A Survey of Robotics Control Based on Learning-Inspired Spiking Neural Networks. Front. Neurorobot*, 12, p. 35. doi: 10.3389/fnbot.2018.00035.

Bouvier, M. *et al.* (2019) 'Spiking neural networks hardware implementations and challenges: A survey', *ACM Journal on Emerging Technologies in Computing Systems*, 15(2). doi: 10.1145/3304103.

Caines, P. E., Huang, M. and Malhamé, R. P. (2018) 'Mean field games', in *Handbook of Dynamic Game Theory*. doi: 10.1007/978-3-319-44374-4_7.

Carmona, R. and Delarue, F. (2013) 'Probabilistic analysis of mean-field games', *SIAM Journal on Control and Optimization*. doi: 10.1137/120883499.

Cui, D. *et al.* (2019) 'Decentralized Formation Control of Multiple Autonomous Underwater Vehicles with Input Saturation Using RISE Feedback Method', in *OCEANS 2018 MTS/IEEE Charleston, OCEAN 2018*. doi: 10.1109/OCEANS.2018.8604743.

Das, B., Schulze, J. and Ganguly, U. (2018) 'Ultra-low energy LIF neuron using Si NIPIN diode for spiking neural networks', *IEEE Electron Device Letters*. Institute of Electrical and Electronics Engineers Inc., 39(12), pp. 1832–1835. doi: 10.1109/LED.2018.2876684.

Frémaux, N., Sprekeler, H. and Gerstner, W. (2013) 'Reinforcement Learning Using a Continuous Time Actor-Critic Framework with Spiking Neurons', *PLoS Computational Biology*, 9(4). doi: 10.1371/journal.pcbi.1003024.

Friedrich, J. and Lengyel, M. (2016) 'Goal-directed decision making with spiking neurons', *Journal of Neuroscience*, 36(5), pp. 1529–1546. doi: 10.1523/JNEUROSCI.2854-15.2016.

Gerstner, W. *et al.* (2014) *Neuronal dynamics: From single neurons to networks and models of cognition*, *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*. doi: 10.1017/CBO9781107447615.

Guéant, O., Lasry, J. M. and Lions, P. L. (2011) 'Mean field games and applications', *Lecture Notes in Mathematics*. doi: 10.1007/978-3-642-14660-2_3.

Izhikevich, E. M. (2003) 'Simple model of spiking neurons', *IEEE Transactions on Neural Networks*. doi: 10.1109/TNN.2003.820440.

Lewis, F. L. and Vrabie, D. (2009) 'Reinforcement learning and adaptive dynamic programming for feedback control', *IEEE Circuits and Systems Magazine*, 9(3), pp. 32–50. doi: 10.1109/MCAS.2009.933854.

Li, H. *et al.* (2015) 'Event-Triggering Sampling Based Leader-Following Consensus in Second-Order Multi-Agent Systems', *IEEE Transactions on Automatic Control*. doi: 10.1109/TAC.2014.2365073.

Nagumo, J., Arimoto, S. and Yoshizawa, S. (1962) 'An Active Pulse Transmission Line Simulating Nerve Axon*', *Proceedings of the IRE*. doi: 10.1109/JRPROC.1962.288235.

Nourian, M. and Caines, Peter E (2013) 'NONLINEAR STOCHASTIC DYNAMICAL SYSTEMS Copyright © by SIAM . Unauthorized reproduction of this article is prohibited .', 51(4), pp. 3302–3331. doi: 10.1137/090750688.

Nourian, M. and Caines, Peter E. (2013) 'ε-nash mean field game theory for nonlinear stochastic dynamical systems with major and minor agents', *SIAM Journal on Control and Optimization*. Society for Industrial and Applied Mathematics, 51(4), pp. 3302–3331. doi: 10.1137/120889496.

Peng, L. *et al.* (2018) 'Decentralized Multi-Robot Formation Control with Communication Delay and Asynchronous Clock', *Journal of Intelligent and Robotic Systems: Theory and Applications*. doi: 10.1007/s10846-017-0557-y.

Vamvoudakis, K. G. and Lewis, F. L. (2010) 'Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem', *Automatica*, 46(5), pp. 878–888. doi: 10.1016/j.automatica.2010.02.018.

Wolfe, J., Houweling, A. R. and Brecht, M. (2010) 'Sparse and powerful cortical spikes', *Current Opinion in Neurobiology*. doi: 10.1016/j.conb.2010.03.006.

Zenke, F. and Ganguli, S. (2018) 'SuperSpike: Supervised learning in multilayer spiking neural networks', *Neural Computation*, 30(6), pp. 1514–1541. doi: 10.1162/neco_a_01086.