A Parallel Control Method for Zero-Sum Game with Unknown Time-Varying System

Qinglai Wei, Zhenhua Zhu, Jie Zhang, and Fei-Yue Wang

***Abstract*—In this paper, based on ACP (ACP: artificial societies, computational experiments, and parallel execution) approach, a parallel control method is proposed for zero-sum games of unknown time-varying systems. The process of constructing a sequence of artificial systems, implementing the computational experiments, and conducting the parallel execution is presented. The artificial systems are constructed to model the real system. Computational experiments adopting adaptive dynamic programming (ADP) are shown to derive control laws for a sequence of artificial systems. The purpose of the parallel execution step is to derive the control laws for the real system. Finally, simulation experiments are provided to show the effectiveness of the proposed method.**

***Index Terms*—Zero-sum games, parallel control, ACP (ACP: artificial societies, computational experiments, and parallel execution), adaptive dynamic programming (ADP)**

I. Introduction

Zero-sum game (ZSG) is a classical problem in the game theory. For the linear systems, the algebraic Riccati equation is solved for ZSG. For the nonlinear systems, the Hamilton-Jacobi-Isaacs equation (HJIE) is employed, which is difficult to solve directly. Hence, in Refs. [1−3], utilizing the successive approximation method, the HJIE is converted to a series of linear differential equations for known systems.

With the research for the ZSG developing, model predictive control (MPC) methods are used to solve the ZSG with known dynamics. In Ref. [4], MPC is used for solving the ZSG between two heterogeneous players, which are an unmanned aerial vehicle and an unmanned ground vehicle. Besides, adaptive dynamic programming (ADP) is introduced to solve ZSG. In Ref. [5], two structures about ADP are proposed for solving the ZSG of known systems. In Ref. [6], the value iteration with history information is proposed for solving the ZSG of known discrete-time systems. For the unknown systems, in Ref. [7], by using the neural networks to approximate the systems, the policy iteration is proposed for solving the HJIE. In Ref. [8], utilizing the fuzzy models to identify the unknown systems, ADP is used to solve the ZSG.

In most of the above papers, the systems are time-invariant and known. For the unknown systems, the traditional model based methods utilize neural networks and fuzzy systems to identify the systems. However, for the unknown time-varying systems, the accurate physical models are difficult to obtain for whole time horizon. Inspired by Ref. [9], the ACP-based (ACP: artificial societies, computational experiments, and parallel execution) parallel control is introduced to handle the ZSG for unknown time-varying systems [10].

The ACP approach is originally proposed for modeling, analysis, and control of complex systems [10]. It consists of the following three steps. (1) Constructing artificial systems: As the accurate physical model for a nonlinear system is difficult to obtain, the artificial systems are necessary to model the dynamics of the real system. Note that the artificial systems are not meant to be accurate to the real system but for the purpose of analysis. (2) Implementing the computational experiments on the artificial systems: Because of the complexity of the real system, it is difficult to carry out computational experiments on the real system. Therefore, it is necessary to find the results of the equivalent problems in the real system from the artificial systems. (3) Implementing the parallel execution: Because an artificial system does not approximate the real system, the interation is necessary to control and manage the real system. The control policies for the real system are derived through the parallel execution.

In this paper, an ACP-based parallel control method is proposed to solve the ZSG of the unknown time-varying systems. Using the ACP approach, the method obtains the control laws for the real system through constructing artificial systems, implementing computational experiments, and conducting parallel execution.

The rest of this paper is organized as follows. In Section II, the problem about ZSG is formulated. In Section III, the details of the proposed method are presented. Finally, in Section IV, simulation results and analysis are presented to demonstrate the effectiveness of the proposed method.

II. Problem Formulation

Consider a class of unknown time-varying systems

$$s_{k+1} = F(s_k, u_{1,k}, u_{2,k}, k), \quad k = 0, 1, 2, \ldots \tag{1}$$

where $F(\cdot)$ is the system function, $u_{1,k} \in \mathbb{R}^m$ and $u_{2,k} \in \mathbb{R}^q$ are the control vectors of two players, and $s_k \in \mathbb{R}^n$ is the state vector of the system, where m, q, and n are the dimensions of the three vectors, respectively.

The performance index function $\mathcal{J}$ of Eq. (1) is defined as follows

Manuscript received: 9 May 2023; revised: 20 September 2023; accepted: 5 January 2024. *(Corresponding authors: Jie Zhang and Fei-Yue Wang.)*

Citation: Q. Wei, Z. Zhu, J. Zhang, and F.-Y. Wang, A parallel control method for zero-sum game with unknown time-varying system, *IJICS*, 2024, 29(1), 37–41.

Qinglai Wei, Zhenhua Zhu, Jie Zhang, and Fei-Yue Wang are with State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: qinglai.wei@ia.ac.cn; zhuzhenhua 2020@ia.ac.cn; jie.zhang@ia.ac.cn; feiyue.wang@ia.ac.cn).

Qinglai Wei and Fei-Yue Wang are also with Institute of Systems Engineering, Macau University of Science and Technology, Macao 999078, China.

Digital Object Identifier 10.62678/IJICS202403.10104

$$\mathcal{J}(s_k,u_{1,k},u_{2,k},k)=\sum_{i=k}^{\infty}U(s_i,u_{1,i},u_{2,i},i) \tag{2}$$

where $U(s_i,u_{1,i},u_{2,i},i)$ is the utility function.

The goal of the ZSG is to obtain the control pair $(u_{1,k}^*,u_{2,k}^*)$ such that

$$\mathcal{J}^*(s_k,k)=\min_{u_{1,k}}\max_{u_{2,k}}\mathcal{J}(s_k,u_{1,k},u_{2,k},k) \tag{3}$$

where $\mathcal{J}^*(s_k,k)$ is the optimal performance index.

According to the Bellman optimality principle in Ref. [11], the optimal performance index $\mathcal{J}^*(s_k,k)$ is written as

$$\mathcal{J}^*(s_k,k)=\min_{u_{1,k}}\max_{u_{2,k}}(U(s_k,u_{1,k},u_{2,k},k)+\mathcal{J}^*(s_{k+1},k+1)) \tag{4}$$

It is difficult to solve Eq. (4) directly. To overcome the difficulty, we propose a parallel control method to derive the control laws for the real system in Eq. (1).

III. Method

In this section, the proposed method is presented in detail. First, a sequence of artificial systems are constructed to model the system in Eq. (1). Second, we implement computational experiments via ADP to obtain the control laws for the artificial systems. Third, through the interaction parallel execution step, the suboptimal control laws for the real system are derived.

A. Artificial System

In many real world scenarios, precise physical models are difficult to obtain. Hence, artificial systems are introduced to model the real system in Eq. (1). Since the real system in Eq. (1) is time-varying, a sequence of artificial systems are constructed to model the real system. The artificial system $\hat{s}_{k+1}$ is written in the following form

$$\hat{s}_{k+1}=\mathcal{F}(s_k,u_{1,k},u_{2,k},k)=\begin{cases}\mathcal{F}_0(s_k,u_{1,k},u_{2,k}), & k_0\leqslant k<k_1;\\ \mathcal{F}_1(s_k,u_{1,k},u_{2,k}), & k_1\leqslant k<k_2;\\ \vdots & \\ \mathcal{F}_\tau(s_k,u_{1,k},u_{2,k}), & k_\tau\leqslant k<k_{\tau+1};\\ \vdots & \end{cases} \tag{5}$$

where $\mathcal{F}_\tau(\cdot),\ \tau=0,1,2,...$ is an artificial system function in time horizon $[k_\tau,k_{\tau+1})$, and $k_0,k_1,k_2,...$ are positive integers.

In this paper, multilayer perceptrons (MLPs) are used to construct the artificial systems, and an artificial system function in time horizon $[k_\tau,k_{\tau+1})$ is expressed as

$$\hat{s}_{k+1}^{\tau}=\mathcal{F}_\tau(s_k,u_{1,k},u_{2,k})=W_\tau^{\mathrm{T}}\phi_\tau(s_k,u_{1,k},u_{2,k}) \tag{6}$$

where W_τ is the weight matrix of MLPs and $\phi_\tau(\cdot)$ is the activation function of MLPs, respectively.

In the process of constructing artificial systems, the gradient descending algorithm is used to update the weights of MLPs [12]. When the weights of MLPs converge, the real system in Eq. (1) is modelled by the artificial system. The process of constructing artificial systems is shown in Algorithm 1.

Algorithm 1 Process of constructing artificial systems

Initialization:

1: Collect observational data from the real system

2: Construct MLPs like Eq. (6)

3: Choose the precision $\epsilon_1>0$ and learning rate α

Iteration:

4: Use the gradient descending algorithm to update the weights of MLPs

5: If the L2-norm $\|W_{\tau,i}-W_{\tau,(i-1)}\|_2>\epsilon$ holds, go to step 4

Return: weight matrix $W_{\tau,i}$

B. Computational Experiment

Since the real system is complex and unknown, carrying out computational experiments on the real system is difficult. Therefore, computational experiments are implemented on the artificial systems via ADP to evaluate and derive control laws for the artificial systems.

To distinguish from the state of the real system in Eq. (1), we introduce the artificial system in time horizon $[k_\tau,k_{\tau+1})$

$$z_{k+1}=\mathcal{F}_\tau(z_k,u_{1,k},u_{2,k}) \tag{7}$$

where $z_k\in\mathbb{R}^n$ denotes the state in the artificial system.

Assuming saddle point exists [13], computational experiments via ADP are used to derive the control laws for the artificial system in a certian time horizon. According to the ADP method [14], the value function and control laws are obtained. For a sequence of artificial systems, there are a sequece of value functions V and control laws U_1 and U_2, which are written as

$$V(z_k,k)=\begin{cases}V^0(z_k), & k_0\leqslant k<k_1;\\ V^1(z_k), & k_1\leqslant k<k_2;\\ \vdots & \\ V^\tau(z_k), & k_\tau\leqslant k<k_{\tau+1};\\ \vdots & \end{cases} \tag{8}$$

$$U_1(z_k,k)=\begin{cases}U_1^0(z_k), & k_0\leqslant k<k_1;\\ U_1^1(z_k), & k_1\leqslant k<k_2;\\ \vdots & \\ U_1^\tau(z_k), & k_\tau\leqslant k<k_{\tau+1};\\ \vdots & \end{cases} \tag{9}$$

$$U_2(z_k,k)=\begin{cases}U_2^0(z_k), & k_0\leqslant k<k_1;\\ U_2^1(z_k), & k_1\leqslant k<k_2;\\ \vdots & \\ U_2^\tau(z_k), & k_\tau\leqslant k<k_{\tau+1};\\ \vdots & \end{cases} \tag{10}$$

In practical applications for computational experiments, MLPs are used to approximate the control laws and the value functions. The structure diagram of the computational experiments is given in Fig. 1. The critic module approximates the value functions, two actor modules approximate the control laws of two players, respectively, and the artificial system module is the artificial systems constructed in Section III.

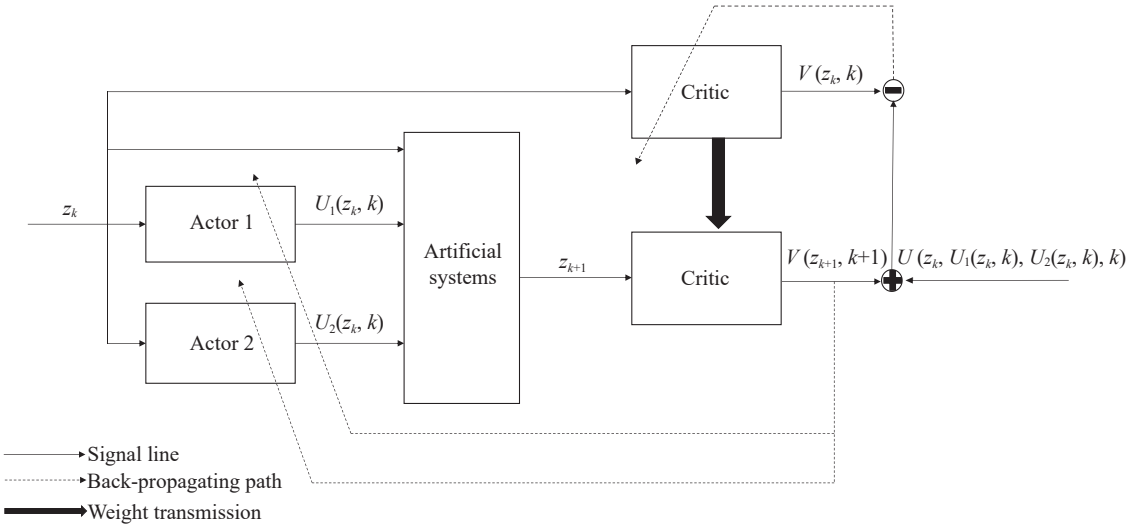


Figure 1 Structure diagram of the computational experiments for a sequence of artificial systems.

For $k \in [k_\tau, k_{\tau+1})$, $\tau = 0, 1, 2, \ldots$, the approximation functions of the value functions $\hat{V}^\tau$ and the control laws $\hat{U}_1^\tau$ and $\hat{U}_2^\tau$ are expressed as

$$\hat{V}^\tau(z_k) = (W_{\mathrm{c}}^\tau)^{\mathrm{T}} \phi_{\mathrm{c}}(z_k) \tag{11}$$

$$\hat{U}_1^\tau(z_k) = (W_{u_1}^\tau)^{\mathrm{T}} \phi_{u_1}(z_k) \tag{12}$$

$$\hat{U}_2^\tau(z_k) = (W_{u_2}^\tau)^{\mathrm{T}} \phi_{u_2}(z_k) \tag{13}$$

where W_{c}^τ, $W_{u_1}^\tau$, and $W_{u_2}^\tau$ are the weight matrices of MLPs, and $\phi_{\mathrm{c}}(\cdot)$, $\phi_{u_1}(\cdot)$, and $\phi_{u_2}(\cdot)$ are the activation functions of MLPs, respectively.

In the process of iteration for computational experiments, the weights of MLPs are tuned by the gradient descending algorithm. When the weights of MLPs converge, the control laws for the artificial systems are derived. The details about the process of computational experiments are presented in Algorithm 2.

Algorithm 2 Computational experiment algorithm

Initialization:

1: Give an initial value function $V_0^\tau(z_k) = 0$

2: Give the precision $\epsilon_2 > 0$

3: Construct three MLPs for approximating the value functions and control laws

Iteration:

4: Compute the iterative value functions and control laws by using ADP

5: Use the gradient descending algorithm to update the weights of three MLPs

6: If $\|W_{\mathrm{c},(i+1)} - W_{\mathrm{c},i}\|_2 > \epsilon$ holds, go to step 4

Return: $W_{u_1,i}^\tau, W_{u_2,i}^\tau$, and $W_{\mathrm{c},i}^\tau$

C. Parallel Execution

After constructing the artificial systems and implementing the computational experiments, the control laws for the artificial systems are derived. As mentioned above, an artificial system in a certain time horizon does not approximate the real system, hence, feedback and interaction are necessary to obtain the control laws for the real system.

In parallel execution step, the control laws derived from computational experiments are implemented not only on the artificial systems but also on the real system, then the difference between performances of the artificial systems and the real system is evaluated. If the diffirence is small, the control laws derived from computational experiments continue to be applied to the real system. Otherwise, it is necessary to update the artificial systems and conduct the computational experiments again.

By constructing the artificial systems, implementing the computational experiments, and conducting the parallel execution, the control laws for the real system are obtained until the state of the real system is stable.

We present a detailed process of parallel execution in Algorithm 3.

Algorithm 3 Parallel execution algorithm

Initialization:

1: Give precisions ϵ_1 and ϵ_2

2: Let $\tau = 0$, $k = 0$, and $k_\tau = k$

3: Give initial states s_0 and z_0, let $z_0 = s_0$

Iteration:

4: Implement Algorithm 1 to construct the artificial system in Eq. (6) under ϵ_1

5: Obtain the weights of two action modules W_u^τ and W_w^τ and the weight of the critic module W_{c}^τ by using Algorihtm 2

6: Implement the control laws on the artificial system in Eq. (6) and the real system in Eq. (1) and evaluate the difference:

if the difference is larger than a threshold

go to step 4

else

if $\|s_k\| > \epsilon_2$, go to step 6

IV. Simulation Study

In this section, to show the effectiveness of the proposed method, we choose an example obtained from Ref. [9] with some modifications, and the dynamics of the system is as follows

$$\begin{pmatrix} s_{1,(k+1)} \\ s_{2,(k+1)} \end{pmatrix} = \begin{pmatrix} s_{1,k} + 0.1 s_{2,k} \\ -0.1 s_{1,k} + 1.1 s_{2,k} - 0.1 s_{1,k}^2 s_{2,k} \end{pmatrix} + B_k u_{1,k} + C_k u_{2,k} \tag{14}$$

where s_1 and s_2 are state variables, u_1 and u_2 are control vectors

$$B_k = \begin{bmatrix} -0.5 + 0.2\sin(k) & 0 \\ 0 & -0.5 + 0.2\sin(k) \end{bmatrix} \tag{15}$$

and

$$C_k = \begin{bmatrix} 0.1 + 0.05\cos(k) & 0 \\ 0 & 0.1 + 0.05\cos(k) \end{bmatrix} \tag{16}$$

Let utility function $U(s_i, u_{1,i}, u_{2,i}) = s_i^{\mathrm{T}} Q s_i + u_{1,i}^{\mathrm{T}} R u_{1,i} - \gamma^2 u_{2,i}^{\mathrm{T}} u_{2,i}$, where $Q = I_1 \in \mathbb{R}^{2\times 2}$ and $R = I_2 \in \mathbb{R}^{2\times 2}$. γ, I_1, and I_2 are identity matrices.

In the artificial systems, MLPs are used to model the real system. For the MLPs, the structure is 6-8-2, with 6 inputs, 8 hidden neurons, and 2 outputs, and the activation functions of the output layer and the hidden layer are purelin functions and tansig functions, respectively. The bias of the output layer and the hidden layer is zero. The training process of weights for MLPs is shown in Algorithm 1 with precision ϵ_1 of 0.001 and learning rate α of 0.02.

Next, in the computational experiments, three MLPs, which are two action networks and one critic network, are used as basis functions to approximate the control laws of two players and the value function. The structures of two action networks and one critic network are chosen as 2-8-2, 2-8-2, and 2-8-1, respectively. For all three MLPs, the activation functions of the output layer and the hidden layer are purelin functions and tansig functions, respectively. The bias of the output layer and the hidden layer is zero. Computational experiments by Algorithm 2 with precision ϵ_2 of 0.001 and learning rate α of 0.02 are implemented to evaluate and derive the control laws for multiple artificial systems.

Then, in the parallel execution, Algorithm 3 is implemented to derive the control laws for the real system. After implementing the proposed method, the trajectory of the states for the real system is shown in Fig. 2. According to Fig. 2, the states of system converge to the equilibrium, showing the effectiveness of the proposed method.

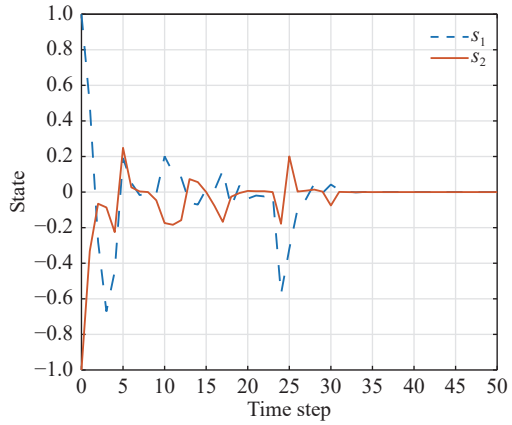


Figure 2 Ttrajectory of the states for the real system.

ACKNOWLEDGMENT

This work was supported in part by the National Key R&D Program of China (No. 2021YFE0206100), the National Natural Science Foundation of China (Nos. 62073321 and 62273036), the National Defense Basic Scientific Research Program (No. JCKY2019203C029), the Science and Technology Development Fund, Macao SAR (Nos. FDCT-22-009-MISE and 0060/2021/A20015/2020/AMJ), and the State Key Lab of Rail Traffic Control & Safety (No. RCS2021K005).

REFERENCES

[1] A. J. Van Der Schaft, L_2-gain analysis of nonlinear-systems and nonlinear state feedback-H_∞ control, *IEEE Trans. Autom. Control*, 1992, 37(6), 770–784.

[2] M. Abu-Khalaf, F. L. Lewis, and J. Huang, Policy iterations on the Hamilton-Jacobi-Isaacs equation for H_∞ state feedback control with input saturation, *IEEE Trans. Autom. Control*, 2006, 51(12), 1989–1995.

[3] S. Mehraeen, T. Dierks, S. Jagannathan, and M. L. Crow, Zero-sum two-player game theoretic formulation of affine nonlinear discrete-time systems using neural networks, *IEEE Trans. Cybern*., 2013, 43(6), 1641–1655.

[4] G. Tzannetos, P. Marantos, and K. J. Kyriakopoulos, A competitive differential game between an unmanned aerial and a ground vehicle using model predictive control, in *Proc. 24th Mediterranean Conference on Control and Automation*, Athens, Greece, 2016, 1053–1058.

[5] A. Al-Tamimi, M. Abu-Khalaf, and F. L. Lewis, Adaptive critic designs for discrete-time zero-sum games with application to H_∞ control, *IEEE Trans. Syst. Man Cybern. Part B Cybern*., 2007, 37(1), 240–247.

[6] M. M. Zhao, D. Wang, M. M. Ha, and J. F. Qiao, Evolving and incremental value iteration schemes for nonlinear discrete-time zero-sum games, *IEEE Trans. Cybern*., 2023, 53(7), 4487–4499.

[7] H. G. Zhang, C. B. Qin, B. Jiang, and Y. H. Luo, Online adaptive policy learning algorithm for H_∞ state feedback control of unknown affine nonlinear discrete-time systems, *IEEE Trans. Cybern*., 2014, 44(12), 2706–2718.

[8] H. G. Zhang, H. G. Su, K. Zhang, and Y. H. Luo, Event-triggered adaptive dynamic programming for non-zero-sum games of unknown nonlinear systems via generalized fuzzy hyperbolic models, *IEEE Trans. Fuzzy Syst*., 2019, 27(11), 2202–2214.

[9] Q. L. Wei, L. X. Wang, J. W. Lu, and F. Y. Wang, Discrete-time self-learning parallel control, *IEEE Trans. Syst. Man Cybern. Syst*., 2022, 52(1), 192–204.

[10] F. Y. Wang, Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications, *IEEE Trans. Intell. Transp. Syst*., 2010, 11(3), 630–638.

[11] R. Bellman and R. Kalaba, *Dynamic Programming and Modern Control Theory*. New York, NY, USA: Academic Press, 1965.

[12] J. Nocedal and S. J. Wright, *Numerical Optimization*. New York, NY, USA: Springer, 1999.

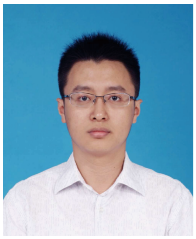
[13] S. Tadelis, *Game Theory: An Introduction*. Princeton, NJ, USA: Princeton University Press, 2013.

[14] Q. L. Wei, D. R. Liu, Q. Lin, and R. Z. Song, Adaptive dynamic programming for discrete-time zero-sum games, *IEEE Trans. Neural Netw. Learn. Syst*., 2018, 29(4), 957–969.

Qinglai Wei received the BS degree in automation and the PhD degree in control theory and control engineering from Northeastern University, Shenyang, China, in 2002 and 2009, respectively. From 2009 to 2011, he was a postdoctoral fellow at State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China. He is currently a professor at the institute and the associate director of the laboratory. He is the secretary of IEEE Computational Intelligence Society Beijing Chapter since 2015. He was guest editor for several international journals. He was a recipient of the IEEE/CAA Journal of Automatica Sinica Best Paper Award, the IEEE System, Man, and Cybernetics Society Andrew P. Sage Best Transactions Paper Award, the IEEE Transactions on Neural Networks and Learning Systems Outstanding Paper Award, the Outstanding Paper Award of Acta Automatica Sinica, the IEEE 6th Data Driven Control and Learning Systems Conference Best Paper Award, and the Zhang Siying Outstanding Paper Award of Chinese Control and Decision Conference. He was a recipient of the Shuang-Chuang Talents

in Jiangsu Province, China, the Young Researcher Award of Asia Pacific Neural Network Society, and the Young Scientst Award and Yang Jiachi Tech Award of Chinese Association of Automation. He is a board of governor member of the International Neural Network Society and a council member of CAA. He has authored four books and published over 80 international journal papers. His research interests include adaptive dynamic programming, neural networks based control, optimal control, nonlinear systems, and their industrial applications.

Zhenhua Zhu received the BS degree in mechanical engineering from Beihang University, Beijing, China, in 2019. He is currently pursuing the MS degree at State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China, and at School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China. His research interests include adaptive dynamic programming and optimal control.



Jie Zhang received the BSc degree in information and computing science from Tsinghua University, Beijing, China, in 2005, and the PhD degree in technology of computer application from University of Chinese Academy of Sciences, Beijing, in 2015. He is currently an associate professor at State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China, since 2016. His research interests include parallel control, mechanism design, optimal control, and multiagent reinforcement learning.

Fei-Yue Wang received the PhD degree in computer and systems engineering from Rensselaer Polytechnic Institute, Troy, NY, USA, in 1990. He joined University of Arizona, Tucson, AZ, USA, in 1990, and became a professor and director at Robotics and Automation Laboratory and Program in Advanced Research for Complex Systems. In 1999, he founded Intelligent Control and Systems Engineering Center, Institute of Automation, Chinese Academy of Sciences (CAS), Beijing, China, under the support of the Outstanding Chinese Talents Program from the State Planning Council, and in 2002, he was appointed as the director of Key Laboratory of Complex Systems and Intelligence Science, CAS, China, and vice president of Institute of Automation, CAS, China, in 2006. In 2011, he became the state specially appointed expert and founding director at State Key Laboratory of Management and Control for Complex Systems, China. He has been the chief judge of Intelligent Vehicles Future Challenge since 2009 and director of China Intelligent Vehicles Proving Center, Changshu, China, since 2015. He is currently the director at Intel's International Collaborative Research Institute on Parallel Driving with CAS and Tsinghua University, Beijing, China.

His research interests include methods and applications for parallel intelligence, social computing, and knowledge automation. He is a fellow of International Council on Systems Engineering (INCOSE), International Federation of Automatic Control (IFAC), American Society of Mechanical Engineers (ASME), and American Association for the Advancement of Science (AAAS). In 2007, he was the recipient of the National Prize in Natural Sciences of China, numerous best papers awards from *IEEE Transactions*, and became an Outstanding Scientist of Association for Computing Machinery (ACM) for his work in intelligent control and social computing. He was the recipient of the IEEE Intelligent Transportation Systems (ITS) Outstanding Application and Research Awards in 2009, 2011, and 2015, respectively, the IEEE Systems, Man, and Cybernetics (SMC) Norbert Wiener Award in 2014, and became the IFAC Pavel J. Nowacki Distinguished Lecturer in 2021.

Since 1997, he has been the general or program chair of more than 30 IEEE, Institute for Operations Research and the Management Sciences (INFORMS), IFAC, ACM, and ASME conferences. He was the president of the IEEE ITS Society from 2005 to 2007, IEEE Council of Radio Frequency Identification (RFID) from 2019 to 2021, Chinese Association for Science and Technology, USA, in 2005, American Zhu Kezhen Education Foundation from 2007 to 2008, vice president of the ACM China Council from 2010 to 2011, vice president and the secretary general of Chinese Association of Automation (CAA) from 2008 to 2018, and vice president of IEEE Systems, Man, and Cybernetics Society from 2019 to 2021. He was the founding editor-in-chief (EiC) of the *International Journal of Intelligent Control and Systems* from 1995 to 2000, *IEEE ITS Magazine* from 2006 to 2007, *IEEE/CAA Journal of Automatica Sinica* from 2014 to 2017, *Journal of Command and Control* from 2015 to 2021, and *Chinese Journal of Intelligent Science and Technology* from 2019 to 2021. He was the EiC of the *IEEE Intelligent Systems* from 2009 to 2012, *IEEE Transactions on Intelligent Transportation Systems* from 2009 to 2016, and *IEEE Transactions on Computational Social Systems* from 2017 to 2020. He is currently the president of CAA's Supervision Council and new EiC of the *IEEE Transactions on Intelligent Vehicles.*