Parallel Control for Continuous-Time Inverse System

Xiang Cheng and Qinglai Wei

***Abstract*—Disturbance compensation methods are widely used to design the robust controller. In order to achieve the robust parallel control, how to implement the disturbance compensation in parallel control laws is studied in this paper. First, the key points are the estimations of the total inputs via inverse systems and the application of system state derivatives. Then, the inverse system based parallel control (ISPC) method is proposed for the optimal control of nonlinear systems. The basic structure of the inverse system based parallel control method is explained and compared with the traditional parallel control methods. The adaptive dynamic programming (ADP) method based on an approximate value function is used to solve the parallel control law. Finally, numerical simulations demonstrate the feasibility of the inverse system based parallel control method.**

***Index Terms*—Parallel control, inverse system, adaptive dynamic programming, robust control**

I. Introduction

PARALLEL control is a new dynamic optimal control method for complex systems [1]. Compared with the classical control methods, the parallel control method takes the optimization of the dynamic performance of controllers into account. This point brings the advantage of continuous and smooth controller output, although it will sacrifice some rapidity. As a result, parallel control is a practical control method, especially in cascade control systems, which require that the output signals of upper-level controllers are continuous and executable. Recently, some researchers have studied the applications of the parallel control method in intelligent vehicle systems [2]. Wei et al. [3] developed a novel linear parallel control method for output regulation problems with disturbances, in which the time variation of the control was constructed instead of the control value itself, to stabilize the output of the system. Jiao et al. [4] investigated the robust parallel control theory and revealed that optimal control of the nominal system with sufficient feedback gain led to robust optimal control, and developed an adaptive dynamic programming (ADP) based event-triggered robust optimal parallel control method for nonlinear uncertain systems.

It is difficult to get the analytic parallel control law via the Hamilton-Jacobi-Bellman (HJB) equation for nonlinear systems [5]. Therefore, the adaptive dynamic programming is widely used to solve the optimal control problem in parallel control. ADP-based optimal control methods adopt the approximate functions, such as neural networks, to address this challenge and have shown the great potential of solving the dynamic programming problem forward-in-time [6–9]. In iterative ADP [6], value iteration and policy iteration are two commonly used techniques for discrete-time (DT) systems. For continuous-time (CT) systems with unknown dynamics, the time-based ADP scheme has been proposed to solve the optimal control problem [10]. By applying a robust item to suppress the unknown disturbances, a data-driven robust adaptive dynamic programming algorithm is proposed for wastewater treatment process to balance the control performance and energy consumption [11]. In Ref. [12], ADP is also applied in the prescribed-time dynamic event-triggered control method for the unknown multi-player nonzero-sum game. For applications, Ref. [13] presents a hybrid ADP approach for solving the optimal tracking control policy of unmanned surface vehicles (USVs), Ref. [14] develops an ADP-based optimal speed regulator for permanent magnet synchronous motors (PMSMs) with both disturbances and actuator faults. Reference [15] proposes a safe adaptive dynamic programming approach to solve the optimal avoidance control problem.

In practical applications, accurate models are commonly difficult to obtain and measurement noise widely exists in sensors. These challenges make it impossible to derive the optimal control law of the real system, thus the robust control law is selected as a substitute. To deal with such challenges, incremental nonlinear dynamic inversion (INDI) is developed by utilizing direct measurements of the state variable derivatives, and reduces the dependency on detailed and accurate mathematical models [16–18]. Currently, the majority of control laws in industry belong to the class of gain-scheduled proportional-integral-derivative (PID) control. Reference [19] analyzes the commonalities between robust hybrid incremental nonlinear dynamic inversion and proportional-integral-derivative flight control law design. Inspired by the improvement from nonlinear dynamic inversion control to incremental nonlinear dynamic inversion control, we apply inverse systems to the parallel control method. Inverse systems are considered as valuable artificial systems necessary for implementing robust parallel control. Thus, inverse system based parallel control (ISPC) method is developed in this paper. Finally, the control law is obtained by solving the optimal control problem of the augmented system that integrates the original system with inverse systems.

Manuscript received: 8 January 2025; revised: 19 March 2025; accepted: 22 April 2025. *(Corresponding author: Xiang Cheng.)*

Citation: X. Cheng and Q. Wei, Parallel control for continuous-time inverse system, *Int. J. Intell. Control Syst.*, 2025, 30(2), 164–168.

Xiang Cheng and Qinglai Wei are with School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China, and also with State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: xiang.cheng@ia.ac.cn; qinglai.wei@ia.ac.cn).

Digital Object Identifier 10.62678/IJICS202506.10180

The remaining sections are organized as follows. Section II introduces the basic framework and the implementation via ADP. In Section III, simulation results are given to verify the feasibility of our proposed method. Finally, Section IV concludes this paper.

II. Framework and Implementation

A. Basic Framework

The inverse system based parallel control method is presented for the following nonlinear system

$$\dot{x} = F(x, u+d) \tag{1}$$

where $x \in \mathbb{R}^n$ represents the state vector of the real system, $u \in \mathbb{R}^k$ is the control input vector, and $d \in \mathbb{R}^k$ represents the unknown disturbances. Assume that the system in Eq. (1) is controllable. The state vector of the virtual system model is represented by the symbol x_m and its dynamics is $\dot{x}_m$. Define the control vector of the system model as u_m, and we can yield the dynamics of the system model as

$$\dot{x}_m = F(x_m, u_m) \tag{2}$$

Based on the function $F(\cdot)$, one inverse system model of Eq. (2) is defined as

$$u_m = G(x_m, \dot{x}_m) \tag{3}$$

The inverse system in Eq. (3) can be used to compute the posteriori estimate $\hat{u}_m$ of total inputs $u+d$ based on the data of $\dot{x}$. In other words, we set $x_m = x$ and replace $\dot{x}_m$ by $\dot{x}$, then we can obtain

$$\hat{u}_m = G(x, \dot{x}) \tag{4}$$

Ideally, $\dot{x}$ is expected to be measured through sensors. When $\dot{x}$ cannot be measured directly, it can be calculated through historical data, which will cause delays. The inverse systems can be derived from the equations of the system model or established through data and neural networks.

Let $u_d = G(0,0)$. The inverse system based parallel control method developed in this paper aims at designing a negative feedback controller $\dot{u}$ as

$$\dot{u} = K(x, \hat{u}_m - u_d) \tag{5}$$

where the function $K(\cdot)$ is continuous and differentiable. As in Ref. [4], the traditional parallel control law depends on the output of the controller and has the following form

$$\dot{u} = K_t(x, u) \tag{6}$$

The basic structure of traditional parallel controllers is shown in Fig. 1. For comparisons, the basic structure of the inverse system based parallel controllers is shown in Fig. 2.

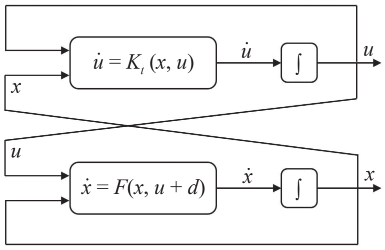


Figure 1 Structure of parallel controller.

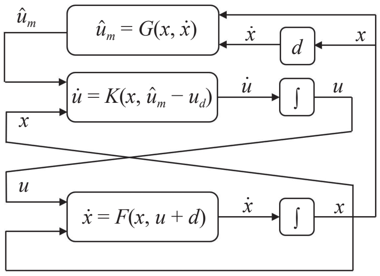


Figure 2 Structure of inverse system based parallel controller.

The difference between ISPC method and the traditional parallel control methods lies in the application of inverse systems. We use Eq. (2) to obtain the control law with regard to the total input of the real system. At the same time, inverse systems are used to estimate the total input based on real-time data and historical data. Then, the inputs of the proposed ISPC law include state measurement values and virtual estimates of control inputs.

B. Implementation via ADP

Let the augmented state $z_m = [x_m, u_m - u_d]^{\mathrm{T}}$. The parallel control law is designed based on an augmented system in Eq. (1) as

$$\begin{aligned}\dot{z}_m = &\begin{bmatrix} F(x_m, u_m) \\ 0_{k\times 1}\end{bmatrix} + \begin{bmatrix} 0_{n\times k} \\ I_{k\times k}\end{bmatrix}\dot{u}_m = \\ &f(z_m) + B\dot{u}_m\end{aligned} \tag{7}$$

where I is the identity matrix. The objective of the parallel control is to minimize the augmented performance index

$$J(z_m, \dot{u}_m) = \int_0^{\infty} (z_m^{\mathrm{T}} Q z_m + \dot{u}_m^{\mathrm{T}} R \dot{u}_m) \mathrm{d}\tau \tag{8}$$

where $Q \geqslant 0$ and $R > 0$.

Based on ADP methods, the optimal control law $\dot{u}_m^*$ and the optimal value function $V^*(z_m)$ satisfy

$$\dot{u}_m^* = -\frac{1}{2} R^{-1} B^{\mathrm{T}} \nabla V^*(z_m) \tag{9}$$

The polynomial function is commonly applied to express the approximation of the optimal value function $V^*(z_m)$, and is listed as

$$\hat{V}(z_m, w) = w^{\mathrm{T}} \sigma(z_m) \tag{10}$$

where $\sigma(\cdot)$ is the activation function vector. Thus, the approximate optimal control law is formulated as

$$\hat{\dot{u}}_m^* = -\frac{1}{2} R^{-1} B^{\mathrm{T}} \nabla^{\mathrm{T}} \sigma(z_m) w \tag{11}$$

Based on the approximate value function, the residual of the HJB equation is constructed as

$$\begin{aligned}\varepsilon_{\mathrm{H}}(z_m, w) = &w^{\mathrm{T}} \nabla \sigma(z_m)(f(z_m) + B\dot{u}_m) + z_m^{\mathrm{T}} Q z_m + \\ &\frac{1}{4} w^{\mathrm{T}} \nabla \sigma(z_m) B R^{-1} B^{\mathrm{T}} \nabla^{\mathrm{T}} \sigma(z_m) w\end{aligned} \tag{12}$$

According to the principle of the optimality, Eq. (12) is the optimal control law when the residual $\varepsilon_{\mathrm{H}}(z_m, w) = 0$. Therefore, the residual of the HJB equation can be selected to construct the loss function to train the weight vector w. Thus, the optimization problem is given as

$$\min_{w} L(z_m) = \frac{1}{2}\varepsilon_{\mathrm{H}}^2 \tag{13}$$

where $L(\cdot)$ represents the loss function.

The weight vector w is trained based on the gradient of $L(z_m)$ as

$$\dot{w} = -\alpha\varepsilon_{\mathrm{H}}\frac{\partial\varepsilon_{\mathrm{H}}}{\partial w} \tag{14}$$

where $\alpha > 0$ is the learning rate. Equation (14) can be used to improve the weight vector w. After a sufficient amount of time, the appropriate weight parameters can be obtained. The process of optimizing the weight vector w using ADP methods is summarized in Algorithm 1.

Define the augmented state $z = [x, \hat{u}_m - u_d]^{\mathrm{T}}$ and replace z_m of the virtual system by measurements z of the real system. Then, we can obtain the parallel control law as

$$\dot{u} = -\frac{1}{2}R^{-1}B^{\mathrm{T}}\nabla^{\mathrm{T}}\sigma(z)w \tag{15}$$

Algorithm 1 Optimizing weight vector

Require:

Choose the initial weight w_0;

Choose the suitable learning rate α;

Choose the suitable stop time t_s;

Ensure: w;

1: Obtain an augmented state z_m randomly;

2: Compute the residual of the HJB equationn $\varepsilon_{\mathrm{H}}(z_m, w)$ according to Eq. (12);

3: Compute the gradient of the loss function $L(z_m)$ according to Eq. (14);

4: Update the weight w of the value function;

5: If the time $t < t_s$, go to step 1. Else go to step 6;

6: **Return** w.

III. Numerical Example

The developed ISPC method is applied to the boiler-turbine system (BTS) to verify its effectiveness. The task of the coordinated control system (CCS) of the BTS is to regulate the output power and achieve the stable power generation. There is a significant inertia in the dynamics of the boiler. The coupling between turbine dynamics and boiler dynamics is the difficulty for coordinated control. The dynamics [20–21] of the BTS are modeled in the following differential equations

$$\begin{aligned} \dot{x}_1 &= 0.9u_1 - 0.0018x_1^{9/8}u_2 - 0.15u_3, \\ \dot{x}_2 &= -0.016x_1^{9/8} - 0.1x_2 + 0.073x_1^{9/8}u_2, \\ \dot{x}_3 &= (0.19x_1 - 1.1x_1u_2 + 141u_3)/85 \end{aligned} \tag{16}$$

where x_1, x_2, and x_3 represent the drum pressure (kg/cm^2), the electric power (MW), and the steam water density (kg/cm^3), respectively. u_1, u_2, and u_3 represent the control valves of the fuel flow, the steam flow, and the feedwater flow, respectively. Thus, the state vector $x_m = [x_1, x_2, x_3]^{\mathrm{T}}$ and the control vector $u_m = [u_1, u_2, u_3]^{\mathrm{T}}$. The output equations of BTS are listed as

$$\begin{aligned} y_1 &= x_1, \\ y_2 &= x_2, \\ y_3 &= 0.05(0.13073x_3 + 100\alpha_{\mathrm{s}} + q_{\mathrm{e}}/9 - 67.975) \end{aligned} \tag{17}$$

where y_1 and y_2 represent the drum pressure (kg/cm^2) and the electric power (MW). y_3 represents the drum water level (m). α_{s} and q_{e} are intermediate variables calculated by

$$\begin{aligned} q_{\mathrm{e}} &= (0.85u_2 - 0.147)x_1 + 45.59u_1 - 2.51u_3 - 2.096, \\ \alpha_{\mathrm{s}} &= \frac{(1 - 0.001538x_3)(0.8x_1 - 25.6)}{x_3(1.0394 - 0.0012304x_1)} \end{aligned} \tag{18}$$

The input variables u_1, u_2, and u_3 must satisfy the following incremental and amplitude constraints

$$\begin{aligned} -0.007 \leqslant \dot{u}_1 \leqslant 0.007,\ 0 \leqslant u_1 \leqslant 1; \\ -2 \leqslant \dot{u}_2 \leqslant 0.02,\ 0 \leqslant u_2 \leqslant 1; \\ -0.05 \leqslant \dot{u}_3 \leqslant 0.05,\ 0 \leqslant u_3 \leqslant 1 \end{aligned} \tag{19}$$

Set the desired state $x_d = [135, 100, 337]^{\mathrm{T}}$ and the augmented state vector $z_m = [x_m^{\mathrm{T}}, u_m^{\mathrm{T}}]^{\mathrm{T}}$. The activation function vector $\sigma(\cdot)$ is selected as

$$\begin{aligned} \sigma(z_m) = [&x_1^2, x_1x_2, x_1x_3, x_1u_1, x_1u_2, x_1u_3, x_2^2, \\ &x_2x_3, x_2u_1, x_2u_2, x_2u_3, x_3^2, x_3u_1, x_3u_2, \\ &x_3u_3, u_1^2, u_1u_2, u_1u_3, u_2^2, u_2u_3, u_3^2]^{\mathrm{T}} \end{aligned} \tag{20}$$

The initial weight vector w_0 is set to be [3.0, 1.0, 1.0, 1.0, 1.0, 1.0, 3.0, 1.0, 1.0, 1.0, 1.0, 3.0, 1.0, 1.0, 1.0, 3.0, 1.0, 1.0, 3.0, 1.0, 3.0]$^{\mathrm{T}}$. The response speed of the output power is the important indicator for power generation systems. Therefore, the weight of state x_2 is set to be larger. Q = diag([0.05, 1.00, 0.05, 1.00, 1.00, 1.00]) and R = diag([1.0, 1.0, 1.0]). The learning rate $\alpha = 0.0005$. Then, on the basis of implementation method of Section II, the weight vector w is trained via ADP. The motion trajectories of weight parameters are drawn in Fig. 3. It can be seen that after about 300 s, the weight parameters converge. The weight vector w at 500 s is [0.3569, 0.0331, 0.0404, 0.4615, 0.0469, −0.0262, 0.3183, 0.0334, 0.0151, 1.9074, 0.0385, 0.2073, 0.0304, −0.0147, 0.4557, 1.3903, 0.0046, −0.0163, 5.9181, −0.0138, 1.4854]$^{\mathrm{T}}$.

Moreover, based on Eq. (16) of BTS, the inverse system can be obtained as

$$\begin{aligned} u_2 &= (\dot{x}_2 + 0.016x_1^{9/8} + 0.1x_2)/(0.073x_1^{9/8}), \\ u_3 &= (85\dot{x}_3 - 0.19x_1 + 1.1x_1u_2)/141, \\ u_1 &= (\dot{x}_1 + 0.0018x_1^{9/8}u_2 + 0.15u_3)/0.9 \end{aligned} \tag{21}$$

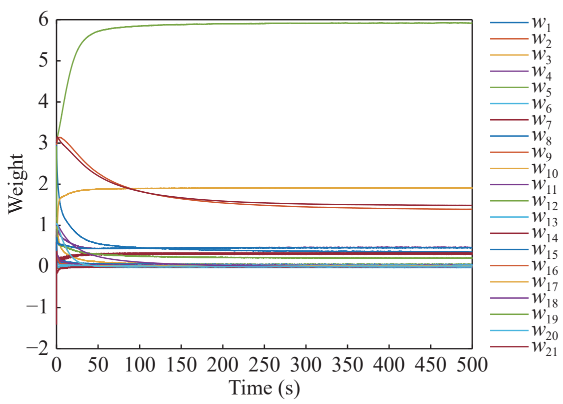


Figure 3 Trajectories of weight parameters.

Based on Eq. (21) and the weight vector w, the parallel control law proposed in this paper can be obtained.

To test the effectiveness of the ISPC method, we consider that the real dynamics of BTS satisfy the following equations

$$\begin{aligned}
\dot{q}_1 &= 0.9u_1/1.05 - 0.0018q_1^{9/8}u_2 - 0.15u_3,\\
\dot{q}_2 &= -0.016q_1^{9/8} - 0.1q_2 + 0.073q_1^{9/8}u_2/1.05,\\
\dot{q}_3 &= (0.19q_1 - 1.1q_1u_2 + 141u_3)/90
\end{aligned} \tag{22}$$

where q_1, q_2, and q_3 represent the measurements of the drum pressure (kg/cm^2), the electric power (MW), and the steam water density (kg/cm^3), respectively. The output equations are the same as in Eqs. (17) and (18).

Next, we apply the parallel control law to the above real system. In other words, the virtual state variables are replaced by their measured values. The simulation results are plotted in Figs. 4–8. The real state variables of BTS have been controlled to the equilibrium point. The mismatches between real dynamics of BTS and virtual dynamics of the model do not result in steady-state errors. The simulation results demonstrate the feasibility of the proposed ISPC method.

IV. Conclusion

The inverse system based parallel control method for nonlinear systems with unknown disturbances is proposed in this paper. Inverse systems are used to compute the posteriori estimate of total inputs based on the real-time data and historical data. ISPC method no longer requires controller output values, but uses virtual total inputs as augmented states. Actuator faults, external disturbances, and so on can be estimated and compensated in parallel control laws. The advantages of ISPC method stem from the use of state derivative measurements. Therefore, the ISPC method assists in reducing the dependence on model accuracy and is easy to implement. The adaptive dynamic programming method based on the approximate value function is used to solve the parallel control law for optimal control.

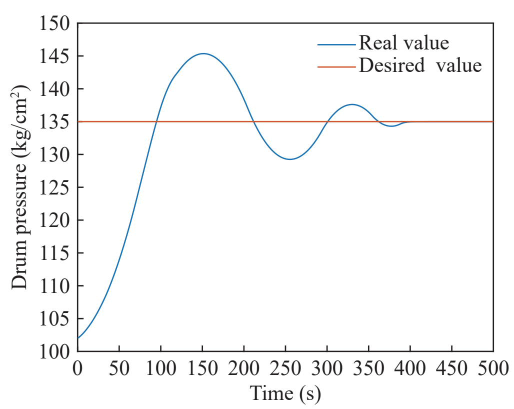


Figure 4 Trajectories of the drum pressure and its desired value.

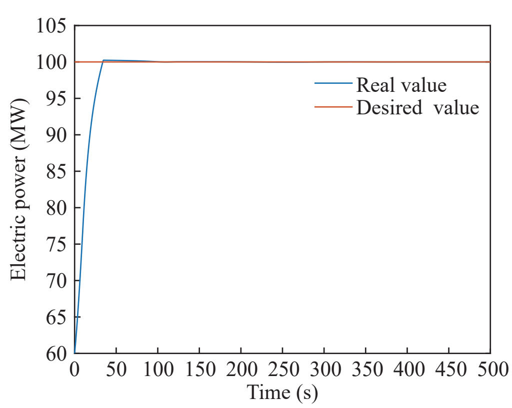


Figure 5 Trajectories of the electric power and its desired value.

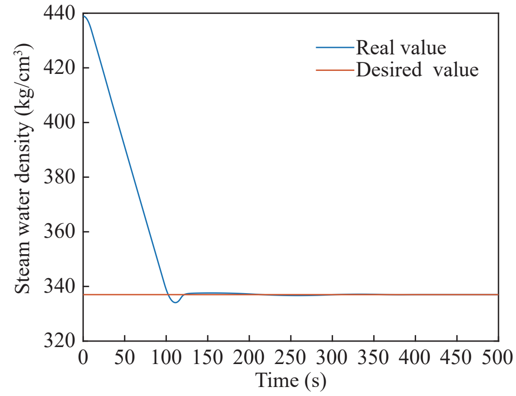


Figure 6 Trajectories of the steam water density and its desired value.

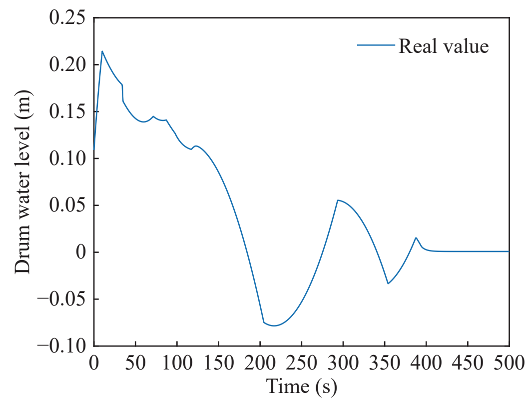


Figure 7 Trajectory of the drum water level.

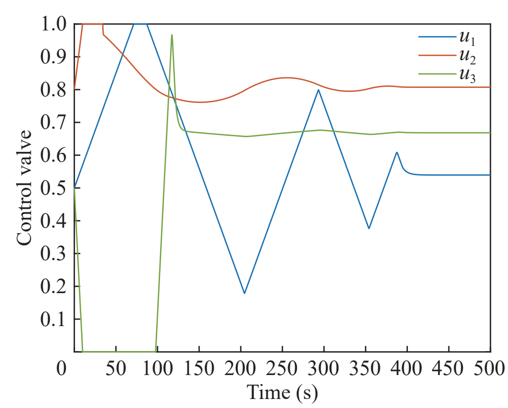


Figure 8 Trajectories of u_1, u_2, and u_3.

REFERENCES

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

[2] F.-Y. Wang, N.-N. Zheng, D. Cao, C. M. Martinez, L. Li, and T. Liu, Parallel driving in CPSS: A unified approach for transport automation and vehicle intelligence, *IEEE/CAA J. Autom. Sin.*, 2017, 4(4), 577–587.

[3] Q. Wei, H. Li, and F.-Y. Wang, A novel parallel control method for continuous-time linear output regulation with disturbances, *IEEE Trans. Cybern.*, 2023, 53(6), 3760–3770.

[4] S. Jiao, Q. Wei, and J. Xiao, Parallel control with event-based adaptive critic implementation for robust optimal tracking of uncertain nonlinear systems, *IEEE Trans. Automat. Sci. Eng.*, 2025, 22, 5673–5686.

[5] P. J. Werbos, Approximate dynamic programming for real-time control and neural modeling, in *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches*. New York, NY, USA: Van Nostrand Reinhold, 1992.

[6] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, Discrete-time nonlinear HJB solution using approximate dynamic programming: Convergence proof, *IEEE Trans. Syst., Man, Cybern., Part B* (*Cybern.*), 2008, 38(4), 943–949.

[7] D. P. Bertsekas, Value and policy iterations in optimal control and adaptive dynamic programming, *IEEE Trans. Neural Netw. Learn. Syst.*, 2017, 28(3), 500–509.

[8] D. Wang, H. He, and D. Liu, Intelligent optimal control with critic learning for a nonlinear overhead crane system, *IEEE Trans. Ind. Inf.*, 2018, 14(7), 2932–2940.

[9] W. Gao and Z.-P. Jiang, Adaptive dynamic programming and adaptive optimal output regulation of linear systems, *IEEE Trans. Automat. Control*, 2016, 61(12), 4164–4169.

[10] T. Sun and X.-M. Sun, An adaptive dynamic programming scheme for nonlinear optimal control with unknown dynamics and its application to turbofan engines, *IEEE Trans. Ind. Inf.*, 2021, 17(1), 367–376.

[11] W. Cao, Q. Yang, W. Meng, and S. Xie, Data-based robust adaptive dynamic programming for balancing control performance and energy consumption in wastewater treatment process, *IEEE Trans. Ind. Inf.*, 2024, 20(4), 6622–6630.

[12] K. Zhang, Z.-X. Zhang, X. Xie, and J. de Jesus Rubio, An unknown multiplayer nonzero-sum game: Prescribed-time dynamic event-triggered control via adaptive dynamic programming, *IEEE Trans. Automat. Sci. Eng.*, 2025, 22, 8317–8328.

[13] S. Xue, N. Zhao, W. Zhang, B. Luo, and D. Liu, A hybrid adaptive dynamic programming for optimal tracking control of USVs, *IEEE Trans. Neural Netw. Learn. Syst.*, 2025, 36(6), 9961–9969.

[14] Z.-X. Fan, S. Li, and J. Su, Adaptive dynamic programming for PMSM control under safety, robustness, and optimality constraints, *IEEE Trans. Syst., Man, Cybern.: Syst.*, 2025, 55(4), 2724–2733.

[15] C. Mu, K. Wang, X. Xu, and C. Sun, Safe adaptive dynamic programming for multiplayer systems with static and moving no-entry regions, *IEEE Trans. Artif. Intell.*, 2024, 5(5), 2079–2092.

[16] S. Salahudden, Decoupled incremental nonlinear dynamic inversion control for aircraft spin recovery, *IEEE Trans. Aerosp. Electron. Syst.*, 2025, 61(2), 3336–3345.

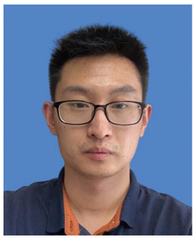
[17] A. Mancinelli, B. D. W. Remes, G. C. H. E. de Croon, and E. J. J. Smeur, Unified incremental nonlinear controller for the transition control of a hybrid dual-axis tilting rotor quad-plane, *IEEE Trans. Robot.*, 2025, 41, 306–325.

[18] Y. Li, X. Liu, P. Lu, Q. He, R. Ming, and W. Zhang, Angular acceleration estimation-based incremental nonlinear dynamic inversion for robust flight control, *Control Eng. Pract.*, 2021, 117, 104938.

[19] T. S. C. Pollack, S. T. Theodoulis, and E. van Kampen, Commonalities between robust hybrid incremental nonlinear dynamic inversion and proportional-integral-derivative flight control law design, *Aerosp. Sci. Technol.*, 2024, 152, 109377.

[20] Q. Wei, J. Lu, T. Zhou, X. Cheng, and F.-Y. Wang, Event-triggered near-optimal control of discrete-time constrained nonlinear systems with application to a boiler-turbine system, *IEEE Trans. Ind. Inf.*, 2022, 186(6), 3926–3935.

[21] K. J. Åström and R. D. Bell, Drum-boiler dynamics, *Automatica*, 2000, 36(3), 363–378.



Xiang Cheng received the BS degree in electrical engineering and the MA degree in control science and engineering from China University of Mining and Technology, Xuzhou, China, in 2016 and 2019, respectively. He is currently pursuing the PhD degree in pattern recognition and intelligent system at School of Artificial Intelligence, University of Chinese Academy of Sciences, China, and at State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, China. He is also an engineer at Institute of Automation, Chinese Academy of Sciences, China. His research interests include parallel control, artificial intelligence, and modeling and optimization of complex industrial processes.

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 and the associate director at State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, China. He is the secretary of IEEE Computational Intelligence Society Beijing Chapter since 2015. He was the 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 Scientist 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 Chinese Association of Automation. He has authored 4 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.