Solving the Zero-Sum Control Problem for Tidal Turbine System: An Online Reinforcement Learning Approach

Haiyang Fang, Maoguang Zhang, Shuping He, *Senior Member, IEEE*, Xiaoli Luan, *Member, IEEE*, Fei Liu, *Member, IEEE*, and Zhengtao Ding, *Senior Member, IEEE*

***Abstract*—A novel completely mode-free integral reinforcement learning (CMFIRL)-based iteration algorithm is proposed in this article to compute the two-player zero-sum games and the Nash equilibrium problems, that is, the optimal control policy pairs, for tidal turbine system based on continuous-time Markov jump linear model with exact transition probability and completely unknown dynamics. First, the tidal turbine system is modeled into Markov jump linear systems, followed by a designed subsystem transformation technique to decouple the jumping modes. Then, a completely mode-free reinforcement learning algorithm is employed to address the game-coupled algebraic Riccati equations without using the information of the system dynamics, in order to reach the Nash equilibrium. The learning algorithm includes one iteration loop by updating the control policy and the disturbance policy simultaneously. Also, the exploration signal is added for motivating the system, and the convergence of the CMFIRL iteration algorithm is rigorously proved. Finally, a simulation example is given to illustrate the effectiveness and applicability of the control design approach.**

***Index Terms*—Game-coupled algebraic Riccati equations, integral reinforcement learning, Markov jump linear systems (MJLSs), tidal turbine, zero-sum games.**

Manuscript received 27 April 2022; revised 6 June 2022; accepted 18 June 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62073001; in part by the Anhui Provincial Key Research and Development Project under Grant 2022i01020013; in part by the State Key Program of National Natural Science Foundation of China under Grant 61833007; and in part by the University Synergy Innovation Program of Anhui Province under Grant GXXT-2021-010. This article was recommended by Associate Editor P. Shi. *(Corresponding author: Shuping He.)*

Haiyang Fang is with the Anhui Engineering Laboratory of Human-Robot Integration System and Intelligent Equipment, School of Electrical Engineering and Automation, Anhui University, Hefei 230601, China, and also with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: fangocean1996@gmail.com).

Maoguang Zhang and Shuping He are with the Anhui Engineering Laboratory of Human-Robot Integration System and Intelligent Equipment, School of Electrical Engineering and Automation, Anhui University, Hefei 230601, China (e-mail: 1057200891@qq.com; shuping.he@ahu.edu.cn).

Xiaoli Luan and Fei Liu are with the Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Jiangnan University, Wuxi 214122, China (e-mail: xiaoli_luan@126.com; fliu@jiangnan.edu.cn).

Zhengtao Ding is with the Department of Electrical and Electronic Engineering, The University of Manchester, Manchester M13 9PL, U.K. (e-mail: zhengtao.ding@manchester.ac.uk).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2022.3186886.

Digital Object Identifier 10.1109/TCYB.2022.3186886

NOMENCLATURE

T_r	Rotor torque of tidal turbine.
C_p	Tidal energy conversion efficient.
ρ	Density of the fluid.
R	Rotor radius.
v	Actual water speed.
B_s	Torsional stiffness of shaft.
K_d	Damping factor of shaft.
N_g	Gear ratio.
J_r	Rotational inertia of tidal turbine rotor.
w_r	Turbine Rotor speed.
J_g	Rotational inertia of generator rotor.
w_g	Generator rotor speed.
T_g	Generator torque.
B_g	Slope of torque-speed curve for tidal induction generators.
θ	Torsional angle of shaft.
β	Pitch angle.
β_r	Referenced pitch angle.
τ	Time constant of actuator.
$\mathbf{E}\{\cdot\}$	Expectation of stochastic processes.
$L_2^n[0,\infty)$	n-dimensional square integrable function vector over $[0,\infty]$.
$\Im(\cdot)$	Weak infinitesimal operator.
A^{T}	Matrix transpose.
A^{-1}	Matrix inverse.
$\mathrm{rank}(A)$	Rank of A.
I_n	n-dimensional identity matrix.
$\mathbb{R}^n$	n-dimensional Euclidean space.
$\|\cdot\|$	Euclidean vector norm.
$\mathbb{Z}_+$	Set of non-negative real numbers.
$\otimes$	Kronecker product.
$\mathrm{vec}(\cdot)$	If $Z = [z_1, z_2, \ldots, z_n] \in \mathbb{R}^{m\times n}$, where z_i $(i = 1, 2, \ldots, n)$ are the column vectors of z, $\mathrm{vec}(Z) = [z_1^{\mathrm{T}}, z_2^{\mathrm{T}}, \ldots, z_n^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{mn}$.

I. INTRODUCTION

IN RECENT years, tidal turbine has been widely used in coastal regions for electricity supply. The increasing capacity of the tidal turbine system has posed higher demand on the control strategies. Many researchers have been concerned with this study in terms of some potential issues [1]–[3]. At present, the mainstream control schemes of tidal turbine works to

achieve the maximum tidal energy conversion efficiency, especially in fluctuant water current [4]. This can amount to the optimal control problems which have been extensively studied in theory and many published results [5]–[9] are available. For linear plants, the optimal control problems are equivalent to computing the Riccati equations correlated with the quadratic cost function of the system state and control input [10]–[16]. In this way, we obtain a positive-definite matrix P and the corresponding optimal control policy that minimizes the performance index. For nonlinear dynamic systems, we can study the optimal control problems through the solutions of the Hamilton–Jacobi–Bellman (HJB) equations [17]–[20]. For more related work, refer to [21]–[23].

On the other hand, the stochastic property of water speed tends to bring more challenges for the design of the aforementioned control strategies. Despite that Markov jump systems have been thoroughly investigated in virtue of the probabilistic description of model parameters switchings induced by external factors [24]–[26], there is no up-to-date research that combines the stochastic characteristic of water speed with the control strategy for tidal turbine. Furthermore, even if the tidal turbine has been modeled into the Markov process, the conventional techniques are no longer suitable to handle the stochasticity of the tidal turbine system, which is the initiative of this study. To that end, we are about to design the optimal control law for the tidal turbine system with stochastic characteristic to achieve the desired performance.

For jumping parameters, the offline methods would be constrained and the efficiency cannot be always guaranteed. Recently, adaptive dynamic programming or reinforcement learning is widely used to study the adaptive optimal control problems [27]–[32]. In reinforcement learning, there is an agent interacting with an environment which provides numeric reward signals for its action with respect to the current state. The objective of reinforcement learning is to learn how to take actions to maximize the expected returns in specific time sequences. Inspired by the important idea of reinforcement learning in [33], if we set "minimizing the control cost" as the control objective and force the dynamical systems to learn from the environment, it can be applied in the adaptive optimal control problems for various systems. Such kinds of methods are classified as online algorithms, and in the process, the system states are collected in real time and the control policy would be upgraded based on iteration. In [11] and [34], the infinite horizon optimal control problems were studied by the online partial mode-free reinforcement learning algorithm. In [35], the online and offline methods were compared and the efficient online reinforcement learning methods were applied to solve the H_∞ control problems. Also, the relevant results were extended to the nonlinear control systems [36]–[39]. More results on this topic can be found in [42]–[46].

The zero-sum games are a class of *min–max* optimization problems that are designed to find the two-player control strategy of the control system. Specifically, the controller is a minimizing player and the disturbance is a maximizing one [47]. By reinforcement learning knowledge, the zero-sum control problems were widely studied via some related methods [48]–[51]. However, the interests of research rarely involve the complex jumping parameters and processes, and the above results cannot be directly applied to jump systems. In the existing literature, Song *et al.* [52] studied a partially model-free adaptive H_∞ control problem by using the policy iteration algorithm. Afterward, some relevant approaches were extended to LDI-represented neural networks [53] and to Itô stochastic systems [54]. Note that some system dynamics are still essential in those works, the completely model-free adaptive optimal control design subject to Markov jump linear systems (MJLSs) has yet been achieved, not to mention more complicated games problems.

In the body part, an online completely mode-free integral reinforcement learning (CMFIRL) algorithm is presented to solve the two-player zero-sum games for the tidal turbine system. The information of system dynamics is not needed here, that is, the solutions can be computed without the identification of system dynamics. Due to the complexity of MJLSs, the subsystem transformation technique is first used to decompose the jumping model into a set of decoupling subsystems. To motivate the system, the exploration noise is also introduced as the actual input and disturbance of the system. The iteration process includes the simultaneous update on the control policy and disturbance policy. Finally, we employ a simulation example to verify the effectiveness and feasibility of this novel algorithm.

The contributions of this work are summarized as follows.

1) We use the Markov process to model the tidal turbine system into MJLSs to deal with the stochasticity while conventional control methods are not applicable.
2) A novel CMFIRL algorithm is implemented for the first time for the two-player zero-sum games and the Nash equilibrium problems for the modeled MJLSs.
3) The subsystems transformation technique is utilized to decompose the MJLSs into N decoupled continuous-time linear subsystems, and with respect to each subsystem, the data collection of states and inputs is used to learn the optimal solution pairs in integral reinforcement learning framework.
4) By the designed CMFIRL algorithm, a new insight is offered into the optimization problems of Markov jump systems. Based on this, our approach has potential to be extended to nonlinear models to solve more intractable HJB equations.

The remainder of this article is organized as follows. First, the physical plant of tidal turbine system, the two-player zero-sum games problem of MJLSs, as well as the corresponding decoupling technique are introduced in Section II. After that, an online CMFIRL algorithm with convergence proof and its online implementation are given in Section III. Finally, a simulation on the tidal turbine system is shown in Section IV, followed by a conclusion in Section V.

II. Backgrounds and Preliminaries

A. Tidal Turbine With Markovian Jump Model

A tidal turbine is an equipment to exploit energy from marine currents, and works in a manner similar to a wind turbine. It is usually placed on the sea bed where there is strong

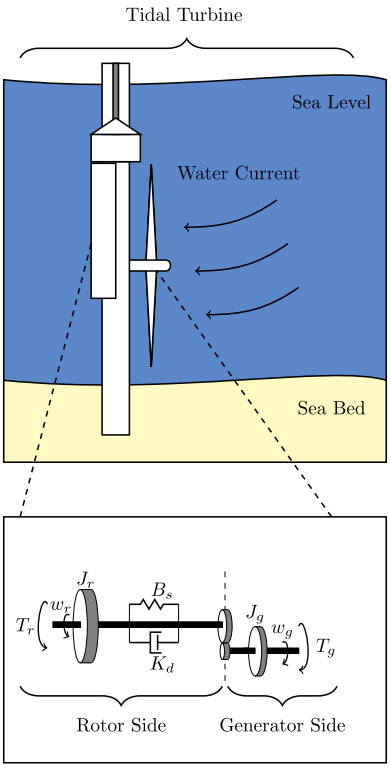


Fig. 1. Schematic of the tidal turbine.

tidal flow. The water current drives the blades of the turbine connected to the gearbox that can create electricity. Consider a tidal turbine system operating in flowing water as illustrated in Fig. 1, which is made up of three parts: 1) the tidal turbine; 2) the motor rotor; and 3) the generator. Basically, the rotor torque that power the entire system is expressed as

$$T_r = \frac{1}{2} C_p \rho \pi R^2 \frac{v^3}{\omega_r}. \tag{1}$$

Because of the stochastic characteristic of the water speed, $v(t)$ can be modeled with the Markov process $r(t)$

$$v(t) = v_a(r(t)) + v_w(t) \tag{2}$$

where v_a is the low-frequency average water speed and v_w is the high-frequency disturbance water speed.

Then, the dynamic of the tidal turbine is described as

$$\begin{cases} J_r \dot{\omega}_r = T_r - B_s\theta - K_d\dot{\theta} \\ \dot{\theta} = \omega_r - \frac{\omega_g}{N_g}. \end{cases} \tag{3}$$

The actuator of pitch angle control is expressed as

$$\dot{\beta} = \frac{1}{\tau}(\beta_r - \beta). \tag{4}$$

Next, defining ω_o as the control output, the dynamic of the generator is represented as

$$\begin{cases} J_g \dot{\omega}_g = \frac{B_s\theta}{N_g} + \frac{K_d\dot{\theta}}{N_g} - T_g \\ T_g = B_g\big(\omega_g - \omega_o\big). \end{cases} \tag{5}$$

Define the state variable as $x = [\omega_r, \omega_g, \theta, \beta]^{\mathrm{T}}$ and the control input as $u = [\beta_r, \omega_o]^{\mathrm{T}}$. By (1)–(5), we can formulate the tidal turbine system as

$$\dot{x} = f(x,t) + g(x,t)u \tag{6}$$

where

$$f(x,t) = \begin{bmatrix} \frac{T_r}{J_r} - \frac{B_s\theta}{J_r} - \frac{K_d}{J_r}(\omega_r - \frac{\omega_g}{N_g}) \\ \frac{K_d\omega_r}{J_gN_g} - \frac{K_d\omega_g}{J_gN_g^2} - \frac{B_g\omega_g}{J_g} + \frac{B_s\theta}{J_gN_g} \\ \omega_r - \frac{\omega_g}{N_g} \\ -\frac{\beta}{\tau} \end{bmatrix}$$

$$g(x,t) = \begin{bmatrix} 0 & 0 \\ 0 & \frac{B_g}{J_g} \\ 0 & 0 \\ \frac{1}{\tau} & 0 \end{bmatrix}.$$

Taking (1), (2), and (6) into account, and by applying the Taylor expansion on v_a and ignoring the high-order terms, T_r can be rewritten as

$$T_r = T_{r1}(v_a) + T_{r2}(v_a)v_w \tag{7}$$

where $T_{r1}(v_a) = T_r|_{v=v_a}$ and $T_{r2}(v_a) = (\partial T_r/\partial v)|_{v=v_a}$.

Substituting (7) into (6), it leads to the following Markov jump nonlinear model with external disturbance:

$$\dot{x} = \tilde{f}(x,t) + g_u(x,t)u + g_w(x,t)v_w \tag{8}$$

where

$$\tilde{f}(x,t) = \begin{bmatrix} \frac{T_{r1}}{J_r} - \frac{B_s\theta}{J_r} - \frac{K_d}{J_r}\left(\omega_r - \frac{\omega_g}{N_g}\right) \\ \frac{K_d\omega_r}{J_gN_g} - \frac{K_d\omega_g}{J_gN_g^2} - \frac{B_g\omega_g}{J_g} + \frac{B_s\theta}{J_gN_g} \\ \omega_r - \frac{\omega_g}{N_g} \\ -\frac{\beta}{\tau} \end{bmatrix}$$

$$g(x,t) = \begin{bmatrix} 0 & 0 \\ 0 & \frac{B_g}{J_g} \\ 0 & 0 \\ \frac{1}{\tau} & 0 \end{bmatrix}, g_w(x,t) = \begin{bmatrix} \frac{T_{r2}}{J_r} \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Applying the Taylor expansion in the state space and omitting the high-order terms, system (8) can be linearized into the continuous-time MJLSs as

$$\dot{x} = A(r(t))x + B_1(r(t))u + B_2(r(t))w \tag{9}$$

where

$$A(r(t)) = \begin{bmatrix} \frac{1}{J_r}\frac{\partial T_{r1}}{\partial \omega_r} - \frac{K_d}{J_r} & \frac{K_d}{J_rN_g} & -\frac{B_s}{J_r} & \frac{1}{J_r}\frac{\partial T_{r1}}{\partial \beta} \\ \frac{K_d}{J_gN_g} & -\frac{B_g}{J_g} - \frac{K_d}{J_gN_g^2} & \frac{B_s}{J_gN_g} & 0 \\ 1 & -\frac{1}{N_g} & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{\tau} \end{bmatrix}$$

$$B_1(r(t)) = \begin{bmatrix} 0 & 0 \\ 0 & \frac{B_g}{J_g} \\ 0 & 0 \\ \frac{1}{\tau} & 0 \end{bmatrix}, B_2(r(t)) = \begin{bmatrix} \frac{T_{r2}}{J_r} \\ 0 \\ 0 \\ 0 \end{bmatrix}, w = v_w.$$

Based on the above, the stochastic characteristic of water speed has been involved in linear model (9) based on the Markov process, which is able to represent the dynamics of tidal turbine at an accurate level. It is worth noting that tradition methods in [55] and [56] are not available for such

systems. Also, the dynamical parameters of (9) are usually difficult or impossible to identify due to variable environmental factors. In what follows, we will develop a CMFIRL algorithm free of the full knowledge of $A(r(t)), B_1(r(t))$, and $B_2(r(t))$ in (9).

B. Two-Player Zero-Sum Games of (9)

Assume that the initial condition of the tidal turbine system is available, we rewrite (9) as a class of continuous-time MJLSs described by

$$\begin{cases} \dot{x}(t) = A(r(t))x(t) + B_1(r(t))u(t) + B_2(r(t))w(t) \\ x(t_0) = x_0, r(t_0) = r_0, t_0 = 0 \end{cases} \tag{10}$$

with the transition probabilities

$$P_r\{r(t+\Delta t) = j | r(t) = i\} = \begin{cases} \pi_{ij}\Delta t + o(\Delta t), & j \neq i \\ 1 + \pi_{ii}\Delta t + o(\Delta t), & j = i \end{cases} \tag{11}$$

where $x(t) \in \mathbb{R}^n$ is the system state, $u(t) \in \mathbb{R}^m$ is the control input, and $w(t) \in \mathbb{R}^q$ is the external disturbance input with $w(t) \in L_2^q[0,\infty)$. $A(r(t)), B_1(r(t))$, and $B_2(r(t))$ are unknown mode-dependent coefficient matrices with appropriate dimensions. $r(t)$ is a continuous-time right continuous Markov random process that represents the system mode and values in a discrete set $M = \{1, 2, \ldots, N\}$. x_0 is the initial state and r_0 is the initial mode. In (11), we have $i, j \in M$, $\Delta t > 0$, and $\lim_{\Delta t \to 0}([o(\Delta t)]/\Delta t) = 0$. $\pi_{ij} \geq 0 (i \neq j)$ represents the transition rate from mode i to mode j at time $t \to t + \Delta t$ and we have $\pi_{ii} = -\sum_{i \neq j} \pi_{ij}$.

For given feedback control policy $u(t) = -K(r(t))x(t)$ and disturbance policy $w(t) = L(r(t))x(t)$, the performance index can be defined as

$$V(x(t)) = \mathbf{E}\left\{\int_0^\infty \Big[x^{\mathrm{T}}(t)Q(r(t))x(t) + u^{\mathrm{T}}(t)R(r(t))u(t) - \gamma^2 w^{\mathrm{T}}(t)w(t)\Big]\mathrm{d}t\right\} \tag{12}$$

where $Q(r(t))$ and $R(r(t))$ are the mode-dependent positive-definite weight matrices, $K(r(t))$ and $L(r(t))$ are the mode-dependent controllers to design, and γ is a positive prescribed scalar. For convenience, when $r(t) = i$, we denote $A(r(t)), B_1(r(t)), B_2(r(t)), Q(r(t)), R(r(t)), K(r(t))$, and $L(r(t))$ as $A_i, B_{1i}, B_{2i}, Q_i, R_i, K_i$, and L_i, respectively. In addition, we assume that the MJLSs (10) are stochastically stabilizable.

Remark 1: In this article, the system parameters and the controllers are mode-dependent, that is, depending on the mode. The reason for us to choose mode-dependent methods is that we can design different controllers for different subsystems to reduce conservativeness.

Definition 1: Subject to MJLSs (10) which are stochastically stable, for any initial conditions x_0 and r_0, we have

$$\lim_{t \to \infty} \mathbf{E}\Big\{\|x(t)\|^2 | x_0, r_0\Big\} = 0. \tag{13}$$

Definition 2: Given a disturbance attenuation level $\gamma \geq 0$, the L_2 gain of MJLSs (10) is no greater than γ if the following relation is established:

$$\int_0^\infty \big[x^{\mathrm{T}}(t)Q_i x(t) + u^{\mathrm{T}}(t)R_i u(t)\big]\mathrm{d}t \leq \int_0^\infty \gamma^2 w^{\mathrm{T}}(t)w(t)\mathrm{d}t. \tag{14}$$

Definition 3: For $\forall i \in M$, the mode-dependent solution $(u(t), w(t))$ is admissible if $(u(t), w(t))$ stochastically stabilizes MJLSs (10) and yields a finite $V(x(t))$.

The two-player zero-sum differential games for MJLSs (10) are to find a mode-dependent admissible saddle-point solution $(u^*(t), w^*(t))$, which minimizes and maximizes the performance index in (12)

$$(u^*(t), w^*(t)) = \arg\min_{u(t)} \max_{w(t)} \mathbf{E}\left\{\int_0^\infty \Big[x^{\mathrm{T}}(t)Q_i x(t) + u^{\mathrm{T}}(t)R_i u(t) - \gamma^2 w^{\mathrm{T}}(t)w(t)\Big]\mathrm{d}t\right\}. \tag{15}$$

Lemma 1 [47]: The mode-dependent unique positive-definite matrix P_i^* can be obtained from the following CGAREs:

$$A_i^{\mathrm{T}}P_i + P_iA_i + Q_i + \sum_{j=1}^N \pi_{ij}P_j - P_iB_{1i}R_i^{-1}B_{1i}^{\mathrm{T}}P_i + \gamma^{-2}P_iB_{2i}B_{2i}^{\mathrm{T}}P = 0 \tag{16}$$

where $P_i = P_i^{\mathrm{T}} > 0$. Then, the feedback control policy and the disturbance policy are determined by

$$\begin{cases} u^*(t) = -K_i^*x(t) = -R_i^{-1}B_{1i}^{\mathrm{T}}P_i^*x(t) \\ w^*(t) = L_i^*x(t) = \gamma^{-2}B_{2i}^{\mathrm{T}}P_i^*x(t). \end{cases} \tag{17}$$

Assumption 1: The triples $(A_i, B_{1i}, Q_i^{1/2})$ are stabizable-detectable and

$$\max_{i \in M}\left\{\inf_{\Gamma_i} |\lambda_{\max}\left[\int_0^\infty e^{\Lambda^{\mathrm{T}}t} \times e^{\Lambda t}\mathrm{d}t\right]\right\} < 1 \tag{18}$$

where $\Lambda = A_i + (\pi_{ii}/2)I + \gamma^{-2}B_{2i}B_{2i}^{\mathrm{T}}\Gamma_i - B_{1i}R_i^{-1}B_{1i}^{\mathrm{T}}\Gamma_i$.

Assumption 1 can ensure that CGAREs (16) have unique positive-definition solutions [14]. To solve (16), we introduce an offline algorithm (Algorithm 1) based on the quadratic Lyapunov equation to solve the two-player games. To analyze the convergence of Algorithm 1, two lemmas are given as follows.

Lemma 2 [57]: An operator $\mathcal{T}$ between two ordered matrix spaces is positive if $\mathcal{T}X_1 > 0$ for $\forall X_1 > 0$.

Lemma 3 [58]: Letting $\mathcal{T}$ be a positive operator and $\rho(\mathcal{T})$ be the spectral radius of $\mathcal{T}$, the following two statements are equivalent: 1) there exists an $X_2 > 0$ such that $\mathcal{T}X_2 - X_2 < 0$ and 2) $\rho(\mathcal{T}) < 1$.

Theorem 1: The solution matrix sequences $\{P_{i(k)}\}$ generated by Algorithm 1 can monotonically converge to the unique positive definition solution of (16), that is, $\lim_{k \to \infty} P_{i(k)} = P_i^*$.

Proof: To begin with, we define a new operator

$$\mathcal{F}(P_{i(k)}) = \Big(A_i + \frac{\pi_{ii}}{2}I\Big)^{\mathrm{T}}P_{i(k)} + P_{i(k)}\Big(A_i + \frac{\pi_{ii}}{2}I\Big) - P_{i(k)}B_{1i}R_i^{-1}B_{1i}^{\mathrm{T}}P_{i(k)}P_{i(k)} + \gamma^{-2}P_{i(k)}B_{2i}B_{2i}^{\mathrm{T}}P_{i(k)} + \sum_{j=1, j \neq i}^N \pi_{ij}P_{j(k)} + Q_i. \tag{22}$$

Algorithm 1: Offline Parallel Iterative Algorithm to Solve the Zero-Sum Games of MJLSs (10) and (11)

Input: Initial stabilizing matrices $\{P_{i(0)}\}$, sufficiently small positive scalar ϵ_1.

Output: Optimal solution $\{P_i^*\}$ of CGAREs (16).

for $i \leftarrow 1$ ***to*** N **do**

  $k \leftarrow 0$;

  while $k \geq 0$ **do**

$$A_{i(k)} = A_i + \frac{\pi_{ii}}{2} I + \gamma^{-2} B_{2i} B_{2i}^{\mathrm{T}} P_{i(k)} - B_{1i} R^{-1} B_{1i}^{\mathrm{T}} P_{i(k)}; \tag{19}$$

$$Q_{i(k)} = \sum_{j=1, i\neq j}^{N} \pi_{ij} P_{j(k)} + Q_i; \tag{20}$$

Solve:

$$A_{i(k)}^{\mathrm{T}} P_{i(k+1)} + P_{i(k+1)} A_{i(k)} + Q_{i(k)} - \gamma^{-2} P_{i(k)} B_{2i} B_{2i}^{\mathrm{T}} P_{i(k)} + P_{i(k)} B_{1i} R^{-1} B_{1i}^{\mathrm{T}} P_{i(k)} = 0; \tag{21}$$

  if $\{\|P_{i(k+1)} - P_{i(k)}\| \geq \epsilon_1$ **then**

    $k \leftarrow k + 1$;

  else

    $P_i^* = P_{i(k+1)}$;

    break;

Return $\{P_i^*\}$.

Then, we have the Fréchet differential of $\mathcal{F}(P_{i(k)})$ with $P_{i(k)}$

$$\mathcal{F}'_{P_{i(k)}}(M) = A_{i(k)}^{\mathrm{T}} M + M A_{i(k)} \tag{23}$$

where $M \in \mathbb{R}^{n\times n}$. By means of (17), we have

$$\begin{cases} \mathcal{F}'_{P_{i(k)}}\big(P_{i(k+1)}\big) = A_{i(k)}^{\mathrm{T}} P_{i(k+1)} + P_{i(k+1)} A_{i(k)} \\ \mathcal{F}'_{P_{i(k)}}\big(P_{i(k)}\big) = A_{i(k)}^{\mathrm{T}} P_{i(k)} + P_{i(k)} A_{i(k)}. \end{cases} \tag{24}$$

On the other side, the computation between (19)–(21) is equivalent to

$$\begin{aligned} &\Big[A_i + \frac{\pi_{ii}}{2} I + \gamma^{-2} B_{2i} B_{2i}^{\mathrm{T}} P_{i(k)} - B_{1i} R_i^{-1} B_{1i}^{\mathrm{T}} P_{i(k)}\Big]^{\mathrm{T}} P_{i(k+1)} \\ &\quad + P_{i(k+1)} \Big[A_i + \frac{\pi_{ii}}{2} + \gamma^{-2} B_{2i} B_{2i}^{\mathrm{T}} P_{i(k)} - B_{1i} R_i^{-1} B_{1i}^{\mathrm{T}} P_{i(k)}\Big] \\ &\quad - \gamma^{-2} P_{i(k)} B_{2i} B_{2i}^{\mathrm{T}} P_{i(k)} + P_{i(k)} B_{1i} R_i^{-1} B_{1i}^{\mathrm{T}} P_{i(k)} + Q_i \\ &\quad + \sum_{j=1, j\neq i}^{N} \pi_{ij} P_{j(k)} = 0. \end{aligned} \tag{25}$$

which equals the following equation by operators $\mathcal{F}$ and $\mathcal{F}'_{P_i(k)}$:

$$\mathcal{F}'_{P_{i(k)}}\big(P_{i(k+1)}\big) = \mathcal{F}'_{P_{i(k)}}\big(P_{i(k)}\big) - \mathcal{F}\big(P_{i(k)}\big). \tag{26}$$

Then, we define another operator with the following form:

$$\mathcal{L} = I - \Big(\mathcal{F}'_{P_i(k)}\Big)^{-1} \mathcal{F}. \tag{27}$$

Using $\mathcal{L}$, (26) can be rewritten as

$$P_{i(k+1)} = \mathcal{L}\big(P_{i(k)}\big). \tag{28}$$

Considering that $P_{i(k+1)}$ and $P_{i(k)}$ are positive definite, it follows $\mathcal{L}(P_{i(k)}) > 0$ and $\mathcal{L}$ is a positive operator according to Lemma 1.

Similar to [59], we give an initial stabilizing sequence $\{P_{i(0)}\}$ such that it satisfies $P_{i(0)} - P_{i(1)} > 0$. Because $P_{i(0)}$ is positive, we have $\mathcal{L}(P_{i(0)}) - P_{i(0)}) = P_{i(1)} - P_{i(0)} < 0$. By Lemma 2, $\rho(\mathcal{L}) < 1$; thus, the output of Algorithm 1 converges to nonincreasing sequences monotonically by the convergence theorem of positive operators [60].

Taking limit on both sides of (25), letting $\lim_{k\to\infty} P_{i(k+1)} = \lim_{k\to\infty} P_{i(k)} = P_i^{\infty}$ and considering that $\pi_{ii} = -\sum_{i\neq j} \pi_{ij}$, we can obtain

$$\begin{aligned} &A_i P_i^{\infty} + P_i^{\infty} A_i + Q_i + \sum_{j=1}^{N} \pi_{ij} P_j^{\infty} + \gamma^{-2} P_i^{\infty} B_{2i} B_{2i}^{\mathrm{T}} P_i^{\infty} \\ &\quad - P_i^{\infty} B_{1i} R_i^{-1} B_{1i}^{\mathrm{T}} P_i^{\infty} = 0. \end{aligned} \tag{29}$$

Obviously, (29) has the same fashion as (16). Considering that P_i^* is the unique solution of (16), it follows $P_i^{\infty} = P_i^*$ and the proof is completed. ■

Although Algorithm 1 is an effective avenue to solve CGAREs (16), it requires the full information of system dynamics. Thus, it is infeasible when A_i, B_{1i}, and B_{2i} are unknown. For this reason, we will go for an online CMFIRL algorithm in Section III.

C. Subsystems Transformation

It is seen from [47] that for continuous-time linear dynamic systems, we can directly solve the two-player zero-sum differential games via an offline algorithm or a partial mode-free online integral reinforcement learning algorithm, but those methods can not be directly applied to MJLSs (10). To that end, we would present the subsystems transformation technique for MJLSs (10) in this section. Before that, we recall the relevant lemmas in the following.

Lemma 4 [61]: For continuous-time linear systems with input disturbance

$$\dot{x}(t) = Ax(t) + B_1 u(t) + B_2 w(t) \tag{30}$$

the optimal control pairs for (30) can be computed by solving P^* from the following game algebraic Riccati equation (GARE):

$$A^{\mathrm{T}} P + PA + Q - PB_1 R^{-1} B_1^{\mathrm{T}} P + \gamma^{-2} PB_2 B_2^{\mathrm{T}} P = 0 \tag{31}$$

with the optimal feedback control policy and disturbance policy

$$\begin{cases} u^*(t) = -K^* x(t) = -R^{-1} B_1^{\mathrm{T}} P^* x(t) \\ w^*(t) = L^* x(t) = \gamma^{-2} B_2^{\mathrm{T}} P^* x(t). \end{cases} \tag{32}$$

Lemma 5 [62]: Giving an initial stabilizing P_0 to solve $P_{(k+1)} (k = 0, 1, \ldots)$ from Lyapunov equations

$$\begin{aligned} &\Big[A + \gamma^{-2} B_2 B_2^{\mathrm{T}} P_{(k)} - B_1 R^{-1} B_1^{\mathrm{T}} P_{(k)}\Big]^{\mathrm{T}} P_{(k+1)} \\ &\quad + P_{(k+1)} \Big[A + \gamma^{-2} B_2 B_2^{\mathrm{T}} P_{(k)} - B_1 R^{-1} B_1^{\mathrm{T}} P_{(k)}\Big] \\ &\quad + Q - \gamma^{-2} P_{(k)} B_2 B_2^{\mathrm{T}} P_{(k)} + P_{(k)} B_1 R^{-1} B_1^{\mathrm{T}} P_{(k)} = 0 \end{aligned} \tag{33}$$

the optimal solution of the GARE (31) can be numerically approximated.

To take a further step, we define the following N continuous-time linear subsystems from MJLSs (10):

$$\begin{cases} \dot{x}_{i,t} = \left(A_i + \frac{\pi_{ii}}{2} I\right) x_{i,t} + B_{1i} u_{i,t} + B_{2i} w_{i,t} \\ x_i(0) = x_{i,0}, t_0 = 0 \end{cases} \tag{34}$$

where $x_{i,t}$, $u_{i,t}$, and $w_{i,t}$ are the system state, control input, and input disturbance of the ith subsystem, respectively.

Theorem 2: For any subsystem (34), if we solve the corresponding two-player zero-sum games by (31) with the same disturbance suppression rate γ and recursively update the corresponding parameter Q_i with

$$Q_{i(k)} = \sum_{j=1, j\neq i}^{N} \pi_{ij} P_{j(k)} + Q_i \tag{35}$$

then we have

$$\lim_{k\to\infty} P_{i(k)} = P_i^* \tag{36}$$

where P_i^* is the optimal solution of CGAREs (16).

Proof: In terms of the ith subsystem, (33) is applicable to solve the two-player zero-sum games problem. It is reasonable to substitute $A_i + (\pi_{ii}/2)I, B_{1i}, B_{2i}, Q_i, R_i, P_{i(k)}$, and $P_{i(k+1)}$ to (33) and it yields

$$\begin{aligned} &\left[A_i + \frac{\pi_{ii}}{2} I + \gamma^{-2} B_{2i} B_{2i}^{\mathrm{T}} P_{i(k)} - B_{1i} R_i^{-1} B_{1i}^{\mathrm{T}} P_{i(k)}\right]^{\mathrm{T}} P_{i(k+1)} \\ &\quad + P_{i(k+1)} \left[A_i + \frac{\pi_{ii}}{2} + \gamma^{-2} B_{2i} B_{2i}^{\mathrm{T}} P_{i(k)} - B_{1i} R_i^{-1} B_{1i}^{\mathrm{T}} P_{i(k)}\right] \\ &\quad - \gamma^{-2} P_{i(k)} B_{2i} B_{2i}^{\mathrm{T}} P_{i(k)} + P_{i(k)} B_{1i} R_i^{-1} B_{1i}^{\mathrm{T}} P_{i(k)} + Q_i = 0. \end{aligned} \tag{37}$$

Considering that Q_i is a given weight matrix, it is reasonable for us to replace Q_i by $Q_{i(k)}$ in (35). In this case, we can obtain (25). Then, according to Theorem 1, we have (16). The proof is completed. ∎

Theorem 2 indicates that the solution of CGAREs (16) equals addressing the two-player zero-sum games problem for N continuous-time linear subsystems. In Section III, the decoupled MJLSs will be studied to establish a completely model-free adaptive optimal control algorithm in the framework of online reinforcement learning.

III. Main Results

In this section, an optimal zero-sum games control law for MJLSs (10) is designed with an online CMFIRL algorithm. In this way, we do not need any knowledge of the origin system. In addition, the corresponding algorithmic implementation and analysis on convergence are provided at the end.

A. CMFIRL Algorithm for Two-Player Zero-Sum Games of MJLSs (10)

In Section II, we have developed Algorithm 1 to solve GCAREs (16). However, it would be difficult even impossible to implement the offline algorithm when the explicit knowledge of system dynamics is not precedently known. To overcome the limitation, we need to design a model-free algorithm. Inspired by the work in [40], we transform (25) into Kleinman's iteration form

$$\begin{aligned} &A_{i(k)}^{\mathrm{T}} P_{i(k)} + P_{i(k)} A_{i(k)} + Q_{i(k)} \\ &\quad + K_{i(k)}^{\mathrm{T}} R_i K_{i(k)} - \gamma^2 L_{i(k)}^{\mathrm{T}} L_{i(k)} = 0 \end{aligned} \tag{38}$$

with

$$\begin{cases} K_{i(k)} = R_i^{-1} B_{1i}^{\mathrm{T}} P_{i(k-1)} \\ L_{i(k)} = \gamma^{-2} B_{2i}^{\mathrm{T}} P_{i(k-1)} \end{cases} \tag{39}$$

where $A_{i(k)} = A_i + (\pi_{ii}/2)I - B_{1i}K_{i(k)} + B_{1i}L_{i(k)}$. $A_{i(k)}$ is Hurwitz with the initial stablizing $K_{i(0)}$ and $L_{i(0)}$ are given [13]. Then, we can rewrite (34) as

$$\begin{cases} \dot{x}_{i,t} = A_{i(k)} x_{i,t} + B_{1i}\left(u_{i,t} + K_{i(k)} x_{i,t}\right) \\ + B_{2i}\left(w_{i,t} - L_{i(k)} x_{i,t}\right) \\ x_i(0) = x_{i,0}, t_0 = 0. \end{cases} \tag{40}$$

Now, we select the value function as $V(x_{i,t}) = x_{i,t}^{\mathrm{T}} P_{i(k)} x_{i,t}$ and calculate its weak infinitesimal operator along the state of subsystem (40)

$$\begin{aligned} \Im V(x_{i,t}) &= x_{i,t}^{\mathrm{T}} (A_{i(k)}^{\mathrm{T}} P_{i(k)} + P_{i(k)} A_{i(k)}) x_{i,t} \\ &\quad + 2\left(u_{i,t} + K_{i(k)} x_{i,t}\right)^{\mathrm{T}} B_{1i}^{\mathrm{T}} P_{i(k)} x_{i,t} \\ &\quad - 2\left(L_{i(k)x_{i,t}} - w_{i,t}\right)^{\mathrm{T}} B_{2i}^{\mathrm{T}} P_{i(k)} x_{i,t} \\ &= -x_{i,t}^{\mathrm{T}} \overline{Q}_{i(k)} x_{i,t} + 2\left(u_{i,t} + K_{i(k)} x_{i,t}\right)^{\mathrm{T}} R_i K_{i(k+1)} x_{i,t} \\ &\quad - 2\gamma^2 \left(L_{i(k)} x_{i,t} - w_{it}\right)^{\mathrm{T}} L_{i(k+1)} x_{i,t} \end{aligned} \tag{41}$$

where $\overline{Q}_{i(k)} = Q_{i(k)} + K_{i(k)}^{\mathrm{T}} R_i K_{i(k)} - \gamma^2 L_{i(k)}^{\mathrm{T}} L_{i(k)}$. For $T > 0$, integrating (41) in time interval $[t, t+T]$, we have

$$\begin{aligned} x_{i,t+T}^{\mathrm{T}} P_{i(k)} x_{i,t+T} - x_{i,t}^{\mathrm{T}} P_{i(k)} x_{i,t} &= -\int_t^{t+T} x_{i,\tau}^{\mathrm{T}} \overline{Q}_{i(k)} x_{i,\tau} \mathrm{d}\tau \\ &\quad + 2\int_t^{t+T} \left[\left(u_{i,\tau} + K_{i(k)} x_{i,\tau}\right)^{\mathrm{T}} R_i K_{i(k+1)} x_{i,\tau}\right] \mathrm{d}\tau \\ &\quad - 2\gamma^2 \int_t^{t+T} \left[\left(L_{i(k)} x_{i,\tau} - w_{i,\tau}\right)^{\mathrm{T}} L_{i(k+1)} x_{i,\tau}\right] \mathrm{d}\tau. \end{aligned} \tag{42}$$

Note that in the above equation, $x_{i,t}, u_{i,t}$, and $w_{i,t}$ are required to learn the optimal solution and to avoid the dynamics knowledge of the subsystem. The convergence analysis on the reinforcement learning algorithm-based iteration (42) is included in the following theorem.

Theorem 3: The solution sequences $\{P_{i(k)}\}, \{K_{i(k)}\}$ and $\{L_{i(k)}\}$ generated by (42) finally converge to the solution of (16) and (17), that is, $\lim_{k\to\infty} P_{i(k)} = P_i^*$, $\lim_{k\to\infty} K_{i(k)} = K_i^*$, and $\lim_{k\to\infty} L_{i(k)} = L_i^*$.

Proof: On the one side, since (42) is derived from Algorithm 1, Algorithm 1 is the sufficient condition of (42). To complete the proof, we still need to prove that Algorithm 1 is the necessary condition. This can be achieved by showing that $P_{i(k)}, K_{i(k+1)}$, and $L_{i(k+1)}$ are uniquely determined by (42).

Letting $V(x_{i,t}) = x_{i,t}^{\mathrm{T}} P_{i(k)} x_{i,t}$ as a Lyapunov candidate function for the subsystem (40), we have $V(x_{i,t}) \geq 0$ and $V(x_{i,t}) = 0$ if and only if $x_{i,t} = 0$. Suppose that (P_i, K_i, L_i) is another solution of (42) and define the Lyapunov function

as $\hat{V}(x_{i,t}) = x_{i,t}^{\mathrm{T}} P x_{i,t}$ with the boundary condition $\hat{V}(x_{i,t}) = 0$ such that

$$\begin{aligned}\Im\hat{V}(x_{i,t}) = {} & 2\big(u_{i,t} + K_{i(k)}x_{i,t}\big)^{\mathrm{T}} R_i K_i x_{i,t} \\ & - 2\gamma^2\big(L_{i(k)}x_{i,t} - w_{i,t}\big)^{\mathrm{T}} L_i x_{i,t} - x_{i,t}^{\mathrm{T}}\overline{Q}_{i(k)}x_{i,t}. \quad (43)\end{aligned}$$

Subtracting (43) from (42), it holds for $\forall u_{i,t} \in \mathbb{R}^m \forall w_{i,t} \in \mathbb{R}^q$ that

$$\begin{aligned}&\Im\Big(V(x_{i,t}) - \hat{V}(x_{i,t})\Big) \\ &\quad = 2\big(u_{i,t} + K_{i(k)}x_{i,t}\big)^{\mathrm{T}} R_i\big(K_{i(k+1)} - K_i\big)x_{i,t} \\ &\qquad - 2\gamma^2\big(L_{i(k)}x_{i,t} - w_{i,t}\big)^{\mathrm{T}}\big(L_{i(k+1)} - L_i\big)x_{i,t}. \quad (44)\end{aligned}$$

Letting $u_{i,t} = -K_{i(k)}x_{i,t}$ and $w_{i,t} = L_{i(k)}x_{i,t}$, one can obtain

$$\Im\Big(V(x_{i,t}) - \hat{V}(x_{i,t})\Big) = 0 \quad (45)$$

that is $V(x_{i,t}) - \hat{V}(x_{i,t}) = x_{i,t}^{\mathrm{T}}(P_{i(k)} - P_i)x_{i,t} = c$ holds for $\forall x \in \mathbb{R}^n$. Because $V(0) - \hat{V}(0) = 0$, so $c = 0$. Immediately, we have $P_{i(k)} = P_i$ and

$$\begin{aligned}&2\big(u_{i,t} + K_{i(k)}x_{i,t}\big)^{\mathrm{T}} R_i\big(K_{i(k+1)} - K_i\big)x_{i,t} \\ &\quad - 2\gamma^2\big(L_{i(k)}x_{i,t} - w_{i,t}\big)^{\mathrm{T}}\big(L_{i(k+1)} - L_i\big)x_{i,t} = 0 \quad (46)\end{aligned}$$

for $\forall u_{i,t} \in \mathbb{R}^m$ and $\forall w_{i,t} \in \mathbb{R}^q$; thus, $K_{k+1} = K_i, L_{i(k+1)} = L_i$. This can sufficiently interpret that $(P_{i(k)}, K_{i(k+1)}, L_{i(k+1)})$ are the unique solutions of (42). This completes the proof. ■

Equation (42) formulates a novel CMFIRL algorithm to solve the addressed problem for MJLSs (10). Obviously, the solution of the N parallel integration equations (42) is completely model free, which does not involve the matrices A_i, B_{1i}, and B_{2i}.

B. Online Implementation of the CMFIRL Algorithm

The online implementation of the CMFIRL algorithm will be deduced by some mathematical operation in this section. For this aim, the left-hand side of (42) is rewritten as

$$x_{i,t+T}^{\mathrm{T}} P_{i(k)} x_{i,t+T} - x_{i,t}^{\mathrm{T}} P_{i(k)} x_{i,t} = \tilde{P}_{i(k)}^{\mathrm{T}}\big(\bar{x}_{i,t+T} - \bar{x}_{i,t}\big) \quad (47)$$

where

$$\tilde{P}_i = \big[p_{i11}, 2p_{i12}, \ldots, 2p_{i1n}, p_{i22}, 2p_{i23}, \ldots, p_{inn}\big]^{\mathrm{T}} \quad (48)$$

and

$$\bar{x}_i = \Big[x_{i1}^2, x_{i1}x_{i2}, \ldots, x_{i1}x_{in}, x_{i2}^2, x_{i2}x_{i3}, \ldots, x_{i2}x_{in}, \ldots, x_{in}^2\Big]^{\mathrm{T}}. \quad (49)$$

Introducing the Kronecker product representation, it has

$$x_{i,t}^{\mathrm{T}}\overline{Q}_{i(k)}x_{i,t} = (x_{i,t}^{\mathrm{T}} \otimes x_{i,t}^{\mathrm{T}})vec(\overline{Q}_{i(k)}). \quad (50)$$

In addition, the two parts of the right-hand side of (42) can be, respectively, represented as

$$\begin{aligned}&\big(u_{i,\tau} + K_{i(k)}x_{i,\tau}\big)^{\mathrm{T}} R_i K_{i(k+1)} x_{i,\tau} \\ &\quad = \Big[\big(x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}}\big)\Big(I_n \otimes K_{i(k)}^{\mathrm{T}} R_i\Big) + \big(x_{i,\tau}^{\mathrm{T}} \otimes u_{i,\tau}^{\mathrm{T}}\big)(I_n \otimes R_i)\Big] \\ &\qquad \cdot \mathrm{vec}\big(K_{i(k+1)}\big) \quad (51) \\ &\big(L_{i(k)}x_{i,\tau} - w_{i,\tau}\big)^{\mathrm{T}} L_{i(k+1)} x_{i,\tau} \\ &\quad = \Big[\big(x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}}\big)\Big(I_n \otimes L_{i(k)}^{\mathrm{T}}\Big) - \big(x_{i,\tau}^{\mathrm{T}} \otimes w_{i,\tau}^{\mathrm{T}}\big)\Big]\mathrm{vec}\big(L_{i(k+1)}\big). \quad (52)\end{aligned}$$

Applying the expressions above to rewrite (42) into

$$\begin{aligned}\tilde{P}_{i(k)}^{\mathrm{T}}\big(\bar{x}_{i,t+T} - \bar{x}_{i,t}\big) = {} & -\int_t^{t+T}\big(x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}}\big)\mathrm{vec}\big(\overline{Q}_{i(k)}\big)\mathrm{d}\tau \\ & + 2\int_t^{t+T}\Big[\big(x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}}\big)\Big(I_n \otimes K_{i(k)}^{\mathrm{T}} R_i\Big) \\ & \quad + \big(x_{i,\tau}^{\mathrm{T}} \otimes u_{i,\tau}^{\mathrm{T}}\big)(I_n \otimes R_i)\Big]\mathrm{vec}(K_{i(k+1)})\mathrm{d}\tau \\ & - 2\gamma^2\int_t^{t+T}\Big[\big(x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}}\big)\Big(I_n \otimes L_{i(k)}^{\mathrm{T}}\Big) \\ & \quad - \big(x_{i,\tau}^{\mathrm{T}} \otimes w_{i,\tau}^{\mathrm{T}}\big)\Big]\mathrm{vec}(L_{i(k+1)})\mathrm{d}\tau. \quad (53)\end{aligned}$$

The compact matrix form is denoted as

$$\Theta_{i(k)}\begin{bmatrix}\tilde{P}_{i(k)} \\ \tilde{K}_{i(k+1)} \\ \tilde{L}_{i(k+1)}\end{bmatrix} = \Xi_{i(k)} \quad (54)$$

where $\Theta_{i(k)} = [\bar{x}_{i,t+T} - \bar{x}_{i,t}, -2\int_t^{t+T}(x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}})(I_n \otimes K_{i(k)}^{\mathrm{T}} R_i)\mathrm{d}\tau - 2\int_t^{t+T}(x_{i,\tau}^{\mathrm{T}} \otimes u_{i,\tau}^{\mathrm{T}})(I_n \otimes R_i)\mathrm{d}\tau, 2\gamma^2\int_t^{t+T}(x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}})(I_n \otimes L_{i(k)}^{\mathrm{T}})\mathrm{d}\tau - 2\gamma^2\int_t^{t+T}(x_{i,\tau}^{\mathrm{T}} \otimes w_{i,\tau}^{\mathrm{T}})\mathrm{d}\tau]$, $\Xi_{i(k)} = -\int_t^{t+T}(x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}})vec(\overline{Q}_{i(k)})\mathrm{d}\tau$, $\tilde{K}_{i(k+1)} = \mathrm{vec}(K_{i(k+1)})$, and $\tilde{L}_{i(k+1)} = vec(L_{i(k+1)})$.

Theorem 3 explains that one can always obtain the unique $(P_{i(k)}, K_{i(k+1)}, L_{i(k+1)})$ with the persistent excitation condition. There are $n(n+1)/2 + nm + nq$ unknown variables in the left-hand side of (54). By the least-square sense, (54) can be accurately solved after sufficient data have been collected.

Unfortunately, persistent excitation condition requires us to reset the system state after each iteration, which can induce some unstable factors to the subsystems. Considering that $x_{i,t}$ is the solution of (40) for the arbitrary control input $u_{i,t}$, the alternative is to utilize the random explosion signal $u_{i,t} = -K_{i(k)}x_{i,t} + e_{1i}$ and $w_{i,t} = L_{i(k)}x_{i,t} + e_{2i}$ on the time interval $[t_0, t_s]$. The explosion signal can replace the persistent excitation without affecting the stability of the systems in the solution process.

Furthermore, we define the following matrices ($s \in \mathbb{Z}_+$):

$$\delta_{ixx} = \begin{bmatrix}\bar{x}_{i,t_1} - \bar{x}_{i,t_0} \\ \bar{x}_{i,t_2} - \bar{x}_{i,t_1} \\ \vdots \\ \bar{x}_{i,t_s} - \bar{x}_{i,t_{s-1}}\end{bmatrix} \quad (55)$$

$$I_{ixx} = \begin{bmatrix}\int_{t_0}^{t_1} x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}}\mathrm{d}\tau \\ \int_{t_1}^{t_2} x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}}\mathrm{d}\tau \\ \vdots \\ \int_{t_{s-1}}^{t_s} x_{i,\tau}^{\mathrm{T}} \otimes x_{i,\tau}^{\mathrm{T}}\mathrm{d}\tau\end{bmatrix} \quad (56)$$

$$I_{ixu} = \begin{bmatrix}\int_{t_0}^{t_1} x_{i,\tau}^{\mathrm{T}} \otimes u_{i,\tau}^{\mathrm{T}}\mathrm{d}\tau \\ \int_{t_1}^{t_2} x_{i,\tau}^{\mathrm{T}} \otimes u_{i,\tau}^{\mathrm{T}}\mathrm{d}\tau \\ \vdots \\ \int_{t_{s-1}}^{t_s} x_{i,\tau}^{\mathrm{T}} \otimes u_{i,\tau}^{\mathrm{T}}\mathrm{d}\tau\end{bmatrix} \quad (57)$$

$$I_{ixw} = \begin{bmatrix} \int_{t_0}^{t_1} x_{i,\tau}^{\mathrm{T}} \otimes w_{i,\tau}^{\mathrm{T}} \mathrm{d}\tau \\ \int_{t_1}^{t_2} x_{i,\tau}^{\mathrm{T}} \otimes w_{i,\tau}^{\mathrm{T}} \mathrm{d}\tau \\ \vdots \\ \int_{t_{s-1}}^{t_s} x_{i,\tau}^{\mathrm{T}} \otimes w_{i,\tau}^{\mathrm{T}} \mathrm{d}\tau \end{bmatrix}. \tag{58}$$

Then, the online implementation form of (54) is obtained as

$$X_{i(k)} \begin{bmatrix} \tilde{P}_{i(k)} \\ \tilde{K}_{i(k+1)} \\ \tilde{L}_{i(k+1)} \end{bmatrix} = Y_{i(k)} \tag{59}$$

where

$$X_{i(k)} = \begin{bmatrix} \delta_{ixx} \\ -2\Big[I_{ixx}(I_n \otimes K_{i(k)}^{\mathrm{T}} R_i) + I_{ixu}(I_n \otimes R_i)\Big] \\ 2\gamma^2\Big[I_{ixx}\Big(I_n \otimes L_{i(k)}^{\mathrm{T}}\Big) - I_{ixw})\Big] \end{bmatrix}^{\mathrm{T}} \tag{60}$$

$$Y_{i(k)} = -I_{ixx} vec(\overline{Q}_{i(k)}). \tag{61}$$

Note that $X_{i(k)}$ have to be with full-column rank to assure that the CMFIRL algorithm can be implemented at each iteration. Then, it yields the following least square matrix equation:

$$\begin{bmatrix} \tilde{P}_{i(k)} \\ \tilde{K}_{i(k+1)} \\ \tilde{L}_{i(k+1)} \end{bmatrix} = \Big(X_{i(k)} X_{i(k)}^{\mathrm{T}}\Big) X_{i(k)} Y_{i(k)}. \tag{62}$$

Ultimately, the solution triples $(P_{i(k)}, K_{i(k+1)}, L_{i(k+1)})$ can be defined by $(\tilde{P}_{i(k)}, \tilde{K}_{i(k+1)}, \tilde{L}_{i(k+1)})$ uniquely.

Theorem 4: If the following rank condition holds for $\forall s > s_0 > 0$:

$$\mathrm{rank}([I_{ixx}, I_{ixu}, I_{ixw}]) = \frac{n(n+1)}{2} + nm + nq \tag{63}$$

$X_{i(k)}$ is of full rank for $\forall k \in \mathbb{Z}_+$.

Proof: Construct a linear matrix equation

$$X_{i(k)} W = 0. \tag{64}$$

Suppose that $W = [E_v^{\mathrm{T}}, F_v^{\mathrm{T}}, G_v^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{([n(n+1)]/2)+mn+nq}$ is one nonzero solution of (64), where $E_v \in \mathbb{R}^{([n(n+1)]/2)}$, $F_v \in \mathbb{R}^{mn}$, and $G_v \in \mathbb{R}^{nq}$. A symmetric matrix $E \in \mathbb{R}^{n\times n}$ and matrices F and G correspond to E_v, F_v, and G_v, respectively, which satisfy $\tilde{E} = E_v$, $\mathrm{vec}(F) = F_v$, and $vec(G) = G_v$.

Recalling to (60), we have

$$X_{i(k)} W = I_{ixx}\mathrm{vec}(S) + 2I_{ixu}\mathrm{vec}(T) + 2I_{ixw}\mathrm{vec}(U) \tag{65}$$

where

$$\begin{aligned} S = {} & A_{i(k)}^{\mathrm{T}} E + E A_{i(k)} + K_{i(k)}^{\mathrm{T}}\big(B_1^{\mathrm{T}} E - RF\big) \\ & + \big(EB_1 - F^{\mathrm{T}} E\big) K_{i(k)} + L_{i(k)}^{\mathrm{T}}\Big(\gamma^2 G - B_{2i}^{\mathrm{T}} E\Big) \\ & + \Big(\gamma^2 G^{\mathrm{T}} - EB_{2i}\Big) \end{aligned} \tag{66}$$

$$T = B_1^{\mathrm{T}} E - RF \tag{67}$$

$$U = \gamma^2 G - B_{2i}^{\mathrm{T}} E. \tag{68}$$

Equations (64) and (65) amount to the linear matrix equation

$$\big[I_{ixx}, 2I_{ixu}, 2I_{ixw}\big] \begin{bmatrix} \mathrm{vec}(S) \\ \mathrm{vec}(T) \\ \mathrm{vec}(U) \end{bmatrix} = 0. \tag{69}$$

Algorithm 2: Online CMFIRL Algorithm to Solve the Two-Player Zero-Sum Games of MJLSs (10) and (11).

Input: Initial stabilizing matrices $\{P_{i(0)}\}$,$\{K_{i(0)}\}$,$\{L_{i(0)}\}$, sufficiently small positive real number ϵ_2.

Output: Optimal controller gain $\{K_i^*\}$, optimal disturbance controller gain $\{L_i^*\}$.

for $i \leftarrow 1$ ***to*** N **do**
  $k \leftarrow -1$;
  while $k < 0$ **do**
    Employ $u_{i,t} = -K_{i(0)}x_{i,t} + e_{1i}$ and $w_{i,t} = -L_{i(0)}x_{i,t} + e_{2i}$ on the time interval $[t_0, t_l]$;
    Compute $\delta_{ixx}, I_{ixx}, I_{ixu}, I_{ixw}$;
    if *(63) is true* **then**
      $k \leftarrow k + 1$;
  while $k \geq 0$ **do**
    Solve $P_{i(k)}, K_{i(k+1)}, L_{i(k+1)}$ from (62);
    if $\{\|P_{i(k)} - P_{i(k-1)}\|\} \geq \epsilon_2$ **then**
      $k \leftarrow k + 1$;
    else
      $P_i^* = P_{i(k)}$;
      $K_i^* = K_{i(k+1)}$;
      $L_i^* = L_{i(k+1)}$;
      break;
Return $\{P_i^*\}, \{K_i^*\}, \{L_i^*\}$.

Since $[I_{ixx}, I_{ixu}, I_{ixw}]$ has full-column rank with the rank condition of (63), the only solution of (69) is $\mathrm{vec}(S) = 0$, $\mathrm{vec}(T) = 0$ and $\mathrm{vec}(U) = 0$. Hence, we have $S = 0$, $T = 0$, and $U = 0$. Furthermore, it follows $F = R^{-1}B_1^{\mathrm{T}}E$ and $G = \gamma^{-2}B_{2i}E$ by (67) and (68). Finally, (66) is simplified as

$$A_{i(k)}^{\mathrm{T}} E + E A_{i(k)} = 0. \tag{70}$$

For $\forall k \in \mathbb{Z}_+$, $A_{i(k)}$ is Hurwitz and (70) only has a trivial solution $E = 0$. Clearly, $F = 0$ and $G = 0$. Hence, $W = 0$. This contradicts the presumption $W \neq 0$, which can complete the proof. ■

From all the above, a novel mode-free online CMFIRL algorithm to solve the two-player zero-sum games of MJLSs (10) is given in Algorithm 2.

Theorem 5: Give initial stabilizing values $K_{i(0)}$ and $L_{i(0)}$, where $\forall i \in M$ in Algorithm 2 and once the rank condition (63) holds, the solutions of (62) equal to the optimal solutions of (16) and (17).

Proof: Given two stabilizing initial value $K_{i(0)}$ and $L_{i(0)}$, if there has a positive solution $P_{i(k)} = P_{i(k)}^{\mathrm{T}}$ satisfying (16), $K_{i(k)}$ and $L_{i(k)}$ can be iteratively obtained by (17). According to Theorem 1, we know the solutions of linear Kleinman iteration equations (38) and (39) satisfy the nonlinear equations (16) and (17). Then, (42) is formulated if the solution of (40) is achieved by (38) and (39). Also, (42) accords with (38) and (39) based on Theorem 3. Therefore, the CMFIRL algorithm for the two-player zero-sum games based on (62) amounts to the solution of (16) and (17). The convergence proof is thus completed. ■

Remark 2: Initializing k to -1 is an algorithm design technique. The loop from procedures 3–8 is broken when $k = 0$, then the algorithm enters into the loop from procedures 9–16.

Remark 3: As illustrated in Algorithm 2, the proposed CMFIRL algorithm to solve the two-player zero-sum games of MJLSs is a completely model-free algorithm, which does not involve any information of matrices A_i, B_{1i}, and B_{2i}. Comparing it with the model-based and the partially model-free method in [63] and [64], we have developed some improved results to the control community.

IV. Numerical Simulation

In order to show the applicability and effectiveness of the proposed online CMFIRL algorithm to solve the two-player zero-sum games problem for the tidal turbine system, we consider the following four-order MJLSs that can represent the tidal turbine system in Section II with two jumping modes:

$$A_1 = \begin{bmatrix} -0.2361 & 1.1358 & -0.5458 & 0.6324 \\ 0.1024 & -3.0213 & 0.3835 & 0 \\ 1 & -0.1123 & 0 & 0 \\ 0 & 0 & 0 & -12.1801 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} -1.5326 & 1.1358 & -0.5458 & 3.7136 \\ 0.1024 & -3.0213 & 0.3835 & 0 \\ 1 & -0.1123 & 0 & 0 \\ 0 & 0 & 0 & -12.1801 \end{bmatrix}$$

$$B_{11} = B_{12} = \begin{bmatrix} 0 & 0 \\ 0 & 10.65 \\ 0 & 0 \\ 12.1801 & 0 \end{bmatrix}, B_{21} = \begin{bmatrix} 1.55 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$B_{22} = \begin{bmatrix} 0.28 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \Pi = \begin{bmatrix} -3 & 3 \\ 2.5 & -2.5 \end{bmatrix}$$

$$Q_1 = Q_2 = I_4, R_1 = R_2 = 1.$$

The disturbance attenuation level $\gamma = 1.5$ for this two-player zero-sum games problem. For the sake of comparison, first we compute the exact solutions by using Algorithm 1. The following solutions with $\epsilon_1 = 10^{-15}$ are obtained after 29 iterations:

$$P_1^* = \begin{bmatrix} 1.1875 & 0.1021 & 0.7223 & 0.0518 \\ 0.1021 & 0.0802 & 0.0501 & 0.0046 \\ 0.7223 & 0.0501 & 1.5433 & 0.0320 \\ 0.0518 & 0.0046 & 0.0320 & 0.0372 \end{bmatrix}$$

$$P_2^* = \begin{bmatrix} 0.5775 & 0.0494 & 0.4360 & 0.0961 \\ 0.0494 & 0.0757 & 0.0262 & 0.0087 \\ 0.4360 & 0.0262 & 1.4198 & 0.0726 \\ 0.0961 & 0.0087 & 0.0726 & 0.0520 \end{bmatrix}$$

$$K_1^* = \begin{bmatrix} 0.6308 & 0.0563 & 0.3895 & 0.4528 \\ 1.0875 & 0.8545 & 0.5336 & 0.0492 \end{bmatrix}$$

$$K_2^* = \begin{bmatrix} 1.1710 & 0.1062 & 0.8840 & 0.6336 \\ 0.5265 & 0.8059 & 0.2794 & 0.0928 \end{bmatrix}$$

$$L_1^* = [0.8181 \quad 0.0703 \quad 0.4976 \quad 0.0357]$$

$$L_2^* = [0.0719 \quad 0.0062 \quad 0.0543 \quad 0.0120].$$

Next, we calculate the optimal solutions by Algorithm 2 with the initial stabilizing feedback gain $K_{i(0)} = 0$ and disturbance gain $L_{i(0)} = 0$, and the initial state is set as $x_{i,0} = [10\ 5\ 20\ 10]^{\mathrm{T}}$. At the learning stage, the control policy and the disturbance policy are set as $u_{i(k)} = -K_{i(k)}x_i + e_{1i}$ and $w_{i(k)} = L_{i(k)}x_i + e_{2i}$ between 0 and 2 s. The exploration noises are chosen as low-frequency signal $e_{1i} = [\exp(-0.01t)\sin(30t) + 0.01\sin(100t), \exp(-0.01t)\sin(15t) + 0.01\sin(100t)]^{\mathrm{T}}$ and high-frequency signal $e_{2i} = \exp(-0.01t)\sin(12t) + 0.01\sin(1500t)$. The simulation is performed by using the data collected at every 0.01 s. By the given iteration criterion, once the rank condition (63) is satisfied, the least squares form (62) can be solved very quickly. Consequently, $P_{i(k)}$ are obtained after 12 iterations with $\epsilon_2 = 10^{-15}$, $K_{i(k)}$ and $L_{i(k)}$ are obtained, respectively, after 13 iterations

$$P_{1(12)} = \begin{bmatrix} 1.1875 & 0.1021 & 0.7223 & 0.0518 \\ 0.1021 & 0.0802 & 0.0501 & 0.0046 \\ 0.7223 & 0.0501 & 1.5433 & 0.0320 \\ 0.0518 & 0.0046 & 0.0320 & 0.0372 \end{bmatrix}$$

$$P_{2(12)} = \begin{bmatrix} 0.5775 & 0.0494 & 0.4360 & 0.0961 \\ 0.0494 & 0.0757 & 0.0262 & 0.0087 \\ 0.4360 & 0.0262 & 1.4198 & 0.0726 \\ 0.0961 & 0.0087 & 0.0726 & 0.0520 \end{bmatrix}$$

$$K_{1(13)} = \begin{bmatrix} 0.6308 & 0.0563 & 0.3895 & 0.4528 \\ 1.0875 & 0.8545 & 0.5336 & 0.0492 \end{bmatrix}$$

$$K_{2(13)} = \begin{bmatrix} 1.1710 & 0.1062 & 0.8840 & 0.6336 \\ 0.5265 & 0.8059 & 0.2794 & 0.0928 \end{bmatrix}$$

$$L_{1(13)} = [0.8181 \quad 0.0703 \quad 0.4976 \quad 0.0357]$$

$$L_{2(13)} = [0.0719 \quad 0.0062 \quad 0.0543 \quad 0.0120].$$

Figs. 2–4 show the simulation results of the proposed Algorithm 2. The values of $P_{1(k)}$ and $P_{2(k)}$ which converge to the optimal values can be seen in Figs. 2 and 3. There are both ten different elements in $P_{1(k)}$ and $P_{2(k)}$; thus, we use ten graphs to illustrate the convergence, respectively. In each graph, the red line shows each element of P_1^* or P_2^* solved by Algorithm 1, while the blue line shows each element of $P_{1(k)}$ or $P_{2(k)}$ after each iteration in Algorithm 2. We can see from Figs. 2 and 3 that $P_{1(k)}$ and $P_{2(k)}$ converge to the optimal solution P_1^* and P_2^* after 12 iterations. The system states and jumping modes are shown in Fig. 4. All states are shown to be stabilized by the designed controller. The control results can effectively evaluate the correctness and feasibility of the proposed approach.

Note that we no more need any knowledge about A_i, B_{1i}, and B_{2i}. The solution accuracy of the results is $\|P_i^* - P_{i(12)}\| < 0.0001$. Therefore, the novel CMFIRL algorithm is feasible and applicable. Moreover, it requires 29 iterations to reach the Nash equilibrium for Algorithm 1, in contrast to only 12 iterations for Algorithm 2. It can be concluded that the proposed method has met all the design specifications addressed in this article. For comparison, we also establish a traditional linear quadratic Gaussian (LQG) controller to solve the two-player zero-sum games from the perspective of H_∞ control. Although the LQG technique cannot directly deal with the Markov jump parameters, we can design the LQG controller

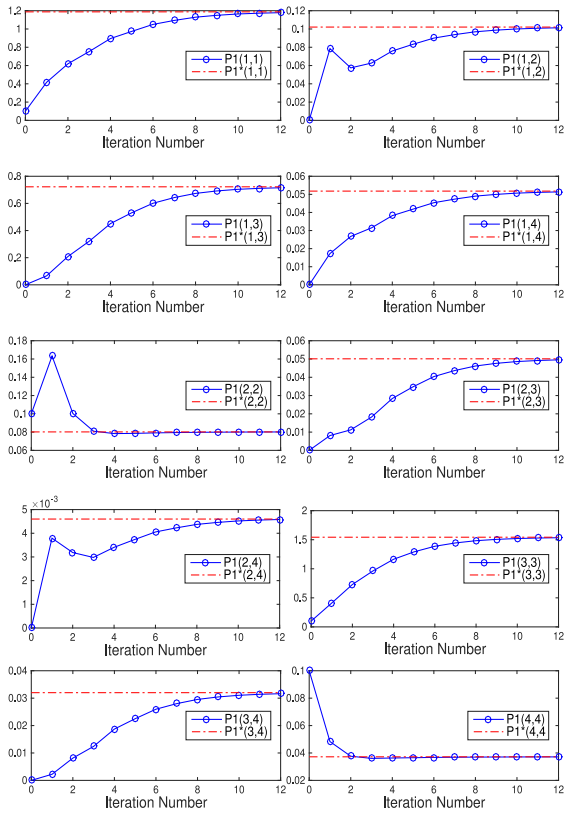


Fig. 2. Convergence of $P_{1(k)}$ to the optimal value P_1^*.

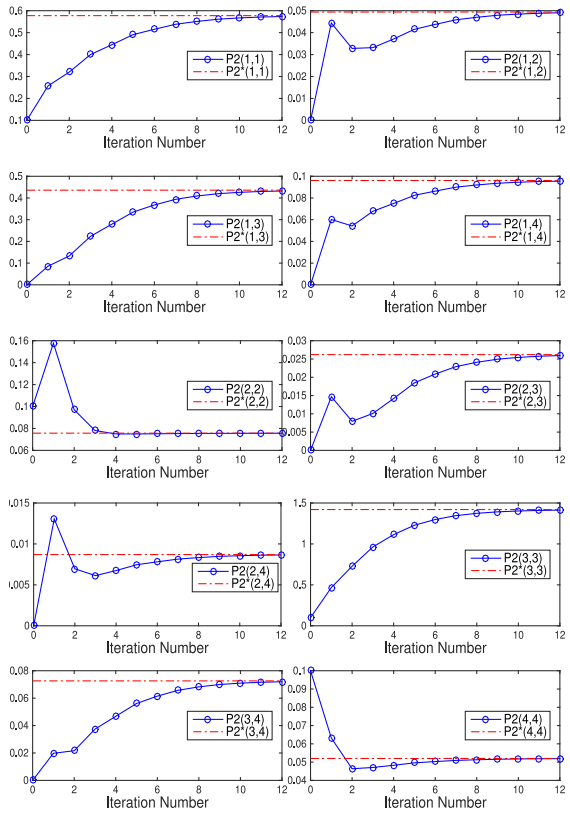


Fig. 3. Convergence of $P_{2(k)}$ to the optimal value P_2^*.

for each decoupled subsystem via the subsystem transformation technique. The control results are shown in Fig. 5, which are basically the same as our method. However, our approach is completely model free, that is, it avoids the system identification process before the controller design. Considering the LQG technique is a mature approach for optimal control, this comparative experiment can fully demonstrate the correctness of the proposed CMFIRL algorithm. In addition, for addressing the influence of different noises on the simulation results, we simulate another case where the exploration noises are given as $e_{1i} = [10\sin(10t)+5\sin(5t), \cos(10t)^2+0.1\sin(t)]^{\mathrm{T}}$, and $e_{2i} = \sin(5000t)+0.5\sin(10000t)$. The results are all the same except the iteration number

$$P_{1(23)} = \begin{bmatrix} 1.1875 & 0.1021 & 0.7223 & 0.0518 \\ 0.1021 & 0.0802 & 0.0501 & 0.0046 \\ 0.7223 & 0.0501 & 1.5433 & 0.0320 \\ 0.0518 & 0.0046 & 0.0320 & 0.0372 \end{bmatrix}$$

$$P_{2(23)} = \begin{bmatrix} 0.5775 & 0.0494 & 0.4360 & 0.0961 \\ 0.0494 & 0.0757 & 0.0262 & 0.0087 \\ 0.4360 & 0.0262 & 1.4198 & 0.0726 \\ 0.0961 & 0.0087 & 0.0726 & 0.0520 \end{bmatrix}$$

$$K_{1(24)} = \begin{bmatrix} 0.6308 & 0.0563 & 0.3895 & 0.4528 \\ 1.0875 & 0.8545 & 0.5336 & 0.0492 \end{bmatrix}$$

$$K_{2(24)} = \begin{bmatrix} 1.1710 & 0.1062 & 0.8840 & 0.6336 \\ 0.5265 & 0.8059 & 0.2794 & 0.0928 \end{bmatrix}$$

$$L_{1(24)} = [0.8181 \quad 0.0703 \quad 0.4976 \quad 0.0357]$$

$$L_{2(24)} = [0.0719 \quad 0.0062 \quad 0.0543 \quad 0.0120].$$

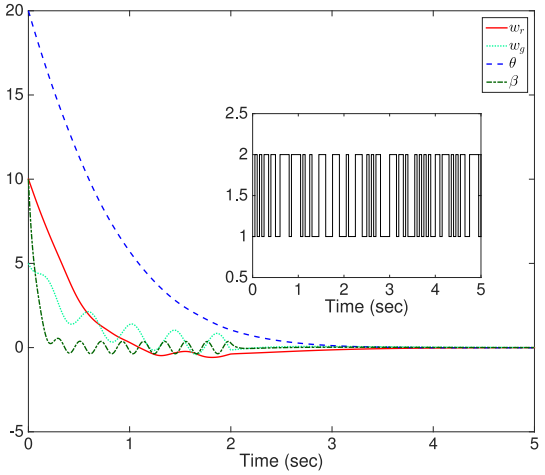


Fig. 4. State trajectories $x(t)$.

The system states and jumping modes in this case are shown in Fig. 6. It is clear that higher frequency disturbance and different jumping mode do not affect the stability and convergence of the control system, which can sufficiently demonstrate the robustness of our algorithm.

Remark 4: In view of the complexity and stochasticity of MJLSs, conventional control methods are not applicable here because they cannot handle the Markov jumping parameters. In addition, the conventional control methods need the prior knowledge of the system dynamics, sometimes, such a system identification process is not available in practical application.

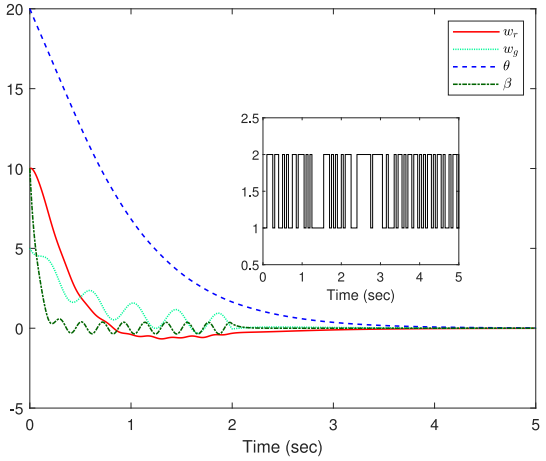


Fig. 5. State trajectories $x(t)$ with the LQG controller.

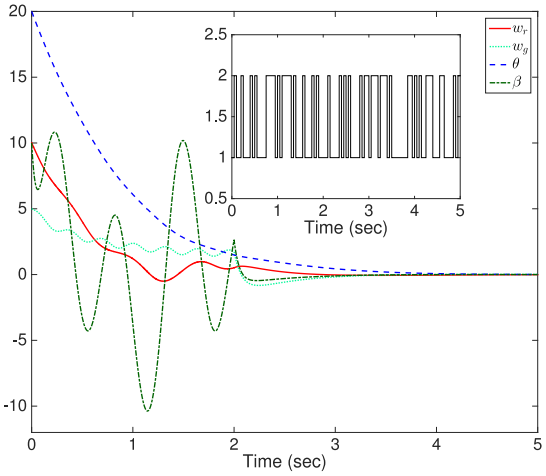


Fig. 6. State trajectories $x(t)$ with different explosion noises.

This is also our motivation to develop the model-free method in this article.

V. Conclusion

In this study, an online CMFIRL algorithm has been proposed with completely unknown dynamics to solve the two-player zero-sum games for the tidal turbine system. It can solve the CGAREs by measuring the data of each subsystem-state trajectories. The CMFIRL algorithm is online and parallelly computes the corresponding N CGAREs without using the knowledge of system dynamics. The convergence proof and a simulation result that demonstrates the effectiveness and robustness of the proposed algorithm were also provided. In the near future, we will extend the proposed method to handle large-scale control problems such as energy-efficient climate control [65].

References

[1] X. Yin and X. Zhao, "Composite hierarchical pitch angle control for a tidal turbine based on the uncertainty and disturbance estimator," *IEEE Trans. Ind. Electron.*, vol. 67, no. 1, pp. 329–339, Jan. 2020.

[2] B. Whitby and C. E. Ugalde-Loo, "Performance of pitch and stall regulated tidal stream turbines," *IEEE Trans. Sustain. Energy*, vol. 5, no. 1, pp. 64–72, Jan. 2014.

[3] R. Genest and J. V. Ringwood, "Receding horizon pseudospectral control for energy maximization with application to wave energy devices," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 1, pp. 29–38, Jan. 2017.

[4] X. Yin and X. Zhao, "Sensorless maximum power extraction control of a hydrostatic tidal turbine based on adaptive extreme learning machine," *IEEE Trans. Sustain. Energy*, vol. 11, no. 1, pp. 426–435, Jan. 2020.

[5] K. J. Astrom and B. Wittenmark, *Adaptive Control*. Chelmsford, MA, USA: Courier Corp., 2013.

[6] C. Mu, D. Wang, and H. He, "Data-driven finite-horizon approximate optimal control for discrete-time nonlinear systems using iterative HDP approach," *IEEE Trans. Cybern.*, vol. 48, no. 10, pp. 2948–2961, Oct. 2018.

[7] Y. Li, K. Sun, and S. Tong, "Observer-based adaptive fuzzy fault-tolerant optimal control for SISO nonlinear systems," *IEEE Trans. Cybern.*, vol. 49, no. 2, pp. 649–661, Feb. 2019.

[8] K. S. Narendra and J. Balakrishnan, "Adaptive control using multiple models," *IEEE Trans. Autom. Control*, vol. 42, no. 2, pp. 171–187, Feb. 1997.

[9] M. Radenkovic and A. N. Michel, "Stochastic adaptive control of non-minimum phase systems in the presence of unmodelled dynamics," *Circuits Syst. Signal Process.*, vol. 14, no. 3, pp. 317–349, 1995.

[10] D. Liu, Q. Wei, and P. Yan, "Generalized policy iteration adaptive dynamic programming for discrete-time nonlinear systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 45, no. 12, pp. 1577–1591, Dec. 2015.

[11] K. G. Vamvoudakis and F. L. Lewis, "Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, 2010.

[12] D. Liu and Q. Wei, "Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 3, pp. 621–634, Mar. 2014.

[13] D. Kleinman, "On an iterative technique for Riccati equation computations," *IEEE Trans. Autom. Control*, vol. 13, no. 1, pp. 114–115, Feb. 1968.

[14] Z. Gajic and I. Borno, "Lyapunov iterations for optimal control of jump linear systems at steady state," *IEEE Trans. Autom. Control*, vol. 40, no. 11, pp. 1971–1975, Nov. 1995.

[15] L.-Z. Lu and W.-W. Lin, "An iterative algorithm for the solution of the discrete-time algebraic Riccati equation," *Linear Algebra Appl.*, vols. 188–189, pp. 465–488, Jul./Aug. 1993.

[16] H. Zhang, H. Jiang, C. Luo, and G. Xiao, "Discrete-time nonzero-sum games for multiplayer using policy-iteration-based adaptive dynamic programming algorithms," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3331–3340, Oct. 2017.

[17] H. Zhang, H. Liang, Z. Wang, and T. Feng, "Optimal output regulation for heterogeneous multiagent systems via adaptive dynamic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 1, pp. 18–29, Jan. 2017.

[18] R. Song and H. Zhang, "The finite-horizon optimal control for a class of time-delay affine nonlinear system," *Neural Comput. Appl.*, vol. 22, no. 2, pp. 229–235, 2013.

[19] V. Narayanan and S. Jagannathan, "Event-triggered distributed control of nonlinear interconnected systems using online reinforcement learning with exploration," *IEEE Trans. Cybern.*, vol. 48, no. 9, pp. 2510–2519, Sep. 2018.

[20] H. Zhang, D. Yue, W. Zhao, S. Hu, and C. Dou, "Distributed optimal consensus control for multiagent systems with input delay," *IEEE Trans. Cybern.*, vol. 48, no. 6, pp. 1747–1759, Jun. 2018.

[21] H. Pan and M. Xin, "Nonlinear robust and optimal control of robot manipulators," *Nonlinear Dyn.*, vol. 76, no. 1, pp. 237–254, 2014.

[22] Q. Wei, F.-Y. Wang, D. Liu, and X. Yang, "Finite-approximation-error-based discrete-time iterative adaptive dynamic programming," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2820–2833, Dec. 2014.

[23] R. Goebel, "Stabilizing a linear system with saturation through optimal control," *IEEE Trans. Autom. Control*, vol. 50, no. 5, pp. 650–655, May 2005.

[24] H. Yang, Y. Jiang, and S. Yin, "Fault-tolerant control of time-delay Markov jump systems with Itô stochastic process and output disturbance based on sliding mode observer," *IEEE Trans. Ind. Informat.*, vol. 14, no. 12, pp. 5299–5307, Dec. 2018.

[25] Z.-G. Wu, Y. Shen, P. Shi, Z. Shu, and H. Su, "H_∞ control for 2-D Markov jump systems in Roesser model," *IEEE Trans. Autom. Control*, vol. 64, no. 1, pp. 427–432, Jan. 2019.

[26] H. Yang and S. Yin, "Actuator and sensor fault estimation for time-delay Markov jump systems with application to wheeled mobile manipulators," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3222–3232, May 2020.

[27] H. Wang, T. Huang, X. Liao, H. Abu-Rub, and G. Chen, "Reinforcement learning for constrained energy trading games with incomplete information," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3404–3416, Oct. 2017.

[28] J. Li, T. Chai, F. L. Lewis, Z. Ding, and Y. Jiang, "Off-policy interleaved Q-learning: Optimal control for affine nonlinear discrete-time systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 5, pp. 1308–1320, May 2019.

[29] J. Li, B. Kiumarsi, T. Chai, F. L. Lewis, and J. Fan, "Off-policy reinforcement learning: Optimal operational control for two-time-scale industrial processes," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4547–4558, Dec. 2017.

[30] J. Hao and H.-F. Leung, "Achieving socially optimal outcomes in multiagent systems with reinforcement social learning," *ACM Trans. Auton. Adapt. Syst.*, vol. 8, no. 3, pp. 1–23, 2013.

[31] J. Hao *et al.*, "An adaptive Markov strategy for defending smart grid false data injection from malicious attackers," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 2398–2408, Jul. 2018.

[32] J. Li, J. Ding, T. Chai, and F. L. Lewis, "Nonzero-sum game reinforcement learning for performance optimization in large-scale industrial processes," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 4132–4145, Sep. 2020.

[33] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 1994, pp. 157–163.

[34] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits Syst. Mag.*, vol. 9, no. 3, pp. 32–50, 3rd Quart., 2009.

[35] H.-N. Wu and B. Luo, "Simultaneous policy update algorithms for learning the solution of linear continuous-time H_∞ state feedback control," *Inf. Sci.*, vol. 222, pp. 472–485, Feb. 2013.

[36] D. Zhai, L. An, D. Ye, and Q. Zhang, "Adaptive reliable H_∞ static output feedback control against Markovian jumping sensor failures," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 3, pp. 631–644, Mar. 2018.

[37] Y. Lv, J. Na, Q. Yang, X. Wu, and Y. Guo, "Online adaptive optimal control for continuous-time nonlinear systems with completely unknown dynamics," *Int. J. Control*, vol. 89, no. 1, pp. 99–112, 2016.

[38] F.-Y. Wang, N. Jin, D. Liu, and Q. Wei, "Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ε-error bound," *IEEE Trans. Neural Netw.*, vol. 22, no. 1, pp. 24–36, Jan. 2011.

[39] X. Yang, H. He, D. Liu, and Y. Zhu, "Adaptive dynamic programming for robust neural control of unknown continuous-time non-linear systems," *IET Control Theory Appl.*, vol. 11, no. 14, pp. 2307–2316, 2017.

[40] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, 2012.

[41] B. Kiumarsi, F. L. Lewis, H. Modares, A. Karimpour, and M.-B. Naghibi-Sistani, "Reinforcement Q-learning for optimal tracking control of linear discrete-time systems with unknown dynamics," *Automatica*, vol. 50, no. 4, pp. 1167–1175, 2014.

[42] S. Zuo, Y. Song, F. L. Lewis, and A. Davoudi, "Optimal robust output containment of unknown heterogeneous multiagent system using off-policy reinforcement learning," *IEEE Trans. Cybern.*, vol. 48, no. 11, pp. 3197–3207, Nov. 2018.

[43] H. Fu, H. Tang, J. Hao, Z. Lei, Y. Chen, and C. Fan, "Deep multi-agent reinforcement learning with discrete-continuous hybrid action spaces," in *Proc. Int. Joint Conf. Artif. Intell.*, 2019, pp. 2329–2335.

[44] C. Zhang *et al.*, "SA-IGA: A multiagent reinforcement learning method towards socially optimal outcomes," *Auton. Agents Multi-Agent Syst.*, vol. 33, no. 4, pp. 403–429, 2019.

[45] J. Qin, Q. Ma, W. X. Zheng, H. Gao, and Y. Kang, "Robust H_∞ group consensus for interacting clusters of integrator agents," *IEEE Trans. Autom. Control*, vol. 62, no. 7, pp. 3559–3566, Jul. 2017.

[46] J. Qin, Q. Ma, H. Gao, Y. Shi, and Y. Kang, "On group synchronization for interacting clusters of heterogeneous systems," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4122–4133, Dec. 2017.

[47] T.-Y. Li and Z. Gajic, "Lyapunov iterations for solving coupled algebraic Riccati equations of Nash differential games and algebraic Riccati equations of zero-sum games," in *New Trends in Dynamic Games and Applications*, vol. 3. Boston, MA, USA: Birkhäuser, 1995, pp. 333–351.

[48] D. Vrabie and F. Lewis, "Adaptive dynamic programming for online solution of a zero-sum differential game," *J. Control Theory Appl.*, vol. 9, no. 3, pp. 353–360, 2011.

[49] J.-S. Wang and G.-H. Yang, "Output-feedback control of unknown linear discrete-time systems with stochastic measurement and process noise via approximate dynamic programming," *IEEE Trans. Cybern.*, vol. 48, no. 7, pp. 1977–1988, Jul. 2018.

[50] X. Zhong, Z. Ni, and H. He, "Gr-GDHP: A new architecture for globalized dual heuristic dynamic programming," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3318–3330, Oct. 2017.

[51] R. A. Howard, *Dynamic Programming and Markov Processes*. Cambridge, MA, USA: MIT Press, 1960.

[52] J. Song, S. He, Z. Ding, and F. Liu, "A new iterative algorithm for solving H_∞ control problem of continuous-time Markovian jumping linear systems based on online implementation," *Int. J. Robust Nonlinear Control*, vol. 26, no. 17, pp. 3737–3754, 2016.

[53] S. He, H. Fang, M. Zhang, F. Liu, and Z. Ding, "Adaptive optimal control for a class of nonlinear systems: The online policy iteration approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 2, pp. 549–558, Feb. 2020.

[54] J. Song, S. He, F. Liu, Y. Niu, and Z. Ding, "Data-driven policy iteration algorithm for optimal control of continuous-time Itô stochastic systems with Markovian jumps," *IET Control Theory Appl.*, vol. 10, no. 12, pp. 1431–1439, 2016.

[55] K. Ghefiri, S. Bouallègue, J. Haggège, I. Garrido, and A. J. Garrido, "Modeling and MPPT control of a tidal stream generator," in *Proc. 4th Int. Conf. Control Decis. Inf. Technol. (CoDIT)*, 2017, pp. 1003–1008.

[56] Z. Lin, J. Liu, Q. Wu, and Y. Niu, "Mixed H_2/H_∞ pitch control of wind turbine with a Markovian jump model," *Int. J. Control*, vol. 91, no. 1, pp. 156–169, 2018.

[57] C. D. Aliprantis and O. Burkinshaw, *Positive Operators*. Dordrecht, The Netherlands: Springer, 2006.

[58] H. Schneider, "Positive operators and an inertia theorem," *Numerische Mathematik*, vol. 7, no. 1, pp. 11–17, 1965.

[59] Z. Gajic and R. Losada, "Monotonicity of algebraic Lyapunov iterations for optimal control of jump parameter linear systems," *Syst. Control Lett.*, vol. 41, no. 3, pp. 175–181, 2000.

[60] J. J. Sopka, "Functional analysis in normed spaces," *SIAM Rev.*, vol. 11, no. 3, p. 412, 1969.

[61] M. Green and D. J. Limebeer, *Linear Robust Control*. Chelmsford, MA, USA: Courier Corp., 2012.

[62] Y. Fu and T. Chai, "Online solution of two-player zero-sum games for continuous-time nonlinear systems with completely unknown dynamics," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 12, pp. 2577–2587, Dec. 2016.

[63] J. C. Geromel and G. W. Gabriel, "Optimal H_2 state feedback sampled-data control design of Markov jump linear systems," *Automatica*, vol. 54, pp. 182–188, Apr. 2015.

[64] S. He, J. Song, Z. Ding, and F. Liu, "Online adaptive optimal control for continuous-time Markov jump linear systems using a novel policy iteration algorithm," *IET Control Theory Appl.*, vol. 9, no. 10, pp. 1536–1543, 2015.

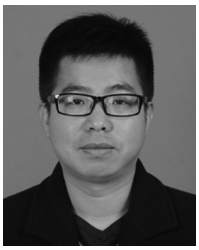
[65] I. Michailidis, S. Baldi, E. B. Kosmatopoulos, and P. A. Ioannou, "Adaptive optimal control for large-scale nonlinear systems," *IEEE Trans. Autom. Control*, vol. 62, no. 11, pp. 5567–5577, Nov. 2017.

Haiyang Fang received the B.Eng. degree in automation from Anhui University, Hefei, China. He is currently pursuing the Ph.D. degree with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong, China.

He has been serving as a Research Fellow with the Anhui Engineering Laboratory of Human–Robot Integration System and Intelligent Equipment, Anhui University, since 2018. His current research interests include optimal control, machine learning, and soft robotics.

Maoguang Zhang received the B.S. degree in automation from Huainan Normal University, Huainan, China, in 2016, and the M.Sc. degree in control theory and control engineering from Anhui University, Hefei, China, in 2019.

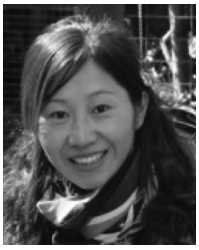
His current research interests include adaptive control, nonlinear systems, reinforcement learning algorithm, and their applications.



Shuping He (Senior Member, IEEE) received the B.S. degree in automation and the Ph.D. degree in control theory and control engineering from Jiangnan University, Wuxi, China, in 2005 and 2011, respectively.

From 2010 to 2011, he was a Visiting Scholar with the Control Systems Centre, School of Electrical and Electronic Engineering, The University of Manchester, Manchester, U.K. He is currently a Professor with the School of Electrical Engineering and Automation, Anhui University, Hefei, China. He has authored or coauthored more than 100 papers in professional journals, conference proceedings, and technical reports in his research areas and published two books about stochastic systems. His current research interests include stochastic systems control, reinforcement learning, system modeling with applications, signal processing, and artificial intelligence methods.

Dr. He is the Associate Editor or Youth Editor for many professional journals, such as the IEEE/CAA JOURNAL OF AUTOMATICA SINICA and *Journal of Central South University*.



Xiaoli Luan (Member, IEEE) received the B.Sc. degree in industrial automation, the M.Sc. degree in control theory and control engineering, and the Ph.D. degree in control theory and control engineering from Jiangnan University, Wuxi, China, in 2002, 2006, and 2010, respectively.

She is currently a Professor with the Institute of Automation, Jiangnan University. In 2016, she was a Visiting Professor with the University of Alberta, Edmonton, AB, Canada. She has authored or coauthored more than 80 articles in professional journals, conference proceedings, and technical reports in these related areas. Her research interests include robust control and optimization of complex industrial process.

Prof. Luan hosted and participated several research programs funded by the National Natural Science Foundation, and served as a reviewer for a number of international journals.

Fei Liu (Member, IEEE) received the Ph.D. degree in control science and control engineering from Zhejiang University, Hangzhou, China, in 2002.

He is currently a Professor with the Institute of Automation, Jiangnan University, Wuxi, China. His current research interests include advanced control theory and applications, batch process control engineering, statistical monitoring and diagnosis in industrial process, and intelligent equipment.

Zhengtao Ding (Senior Member, IEEE) received the B.Eng. degree in thermal energy from Tsinghua University, Beijing, China, 1984, and the M.Sc. degree in systems and control and the Ph.D. degree in control systems from the University of Manchester Institute of Science and Technology, Manchester, U.K., in 1986 and 1989, respectively.

After working as a Lecturer with Ngee Ann Polytechnic, Singapore, for ten years, in 2003, he joined The University of Manchester, Manchester, where he is currently a Senior Lecturer of Control Engineering with the School of Electrical and Electronic Engineering. He has authored the book *Nonlinear and Adaptive Control Systems* (IET, 2013) and a number of journal papers. His research interests include nonlinear and adaptive control theory and their applications.

Dr. Ding serves or served as an Associate Editor for the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, *Transactions of the Institute of Measurement and Control*, *Control Theory and Technology*, *Mathematical Problems in Engineering*, *Unmanned Systems*, and *International Journal of Automation and Computing*.