mathematics

MDPI

Review

On Model Identification Based Optimal Control and It's Applications to Multi-Agent Learning and Control

Rui Luo [1,2], Zhinan Peng [1] and Jiangping Hu [1,2,*]

[1] School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
[2] Yangtze Delta Region Institute (Huzhou), University of Electronic Science and Technology of China, Huzhou 313001, China
* Correspondence: hujp@uestc.edu.cn

Abstract: This paper reviews recent progress in model identification-based learning and optimal control and its applications to multi-agent systems (MASs). First, a class of learning-based optimal control method, namely adaptive dynamic programming (ADP), is introduced, and the existing results using ADP methods to solve optimal control problems are reviewed. Then, this paper investigates various kinds of model identification methods and analyzes the feasibility of combining the model identification method with the ADP method to solve optimal control of unknown systems. In addition, this paper expounds the current applications of model identification-based ADP methods in the fields of single-agent systems (SASs) and MASs. Finally, some conclusions and some future directions are presented.

Keywords: model identification; optimal control; multi-agent systems; adaptive dynamic programming; reinforcement learning

MSC: 49L20; 93B30; 68T07

check for updates

Citation: Luo, R.; Peng, Z.; Hu, J. On Model Identification Based Optimal Control and It's Applications to Multi-Agent Learning and Control. *Mathematics* **2023**, *11*, 906. https://doi.org/10.3390/math11040906

Academic Editors: Aydin Azizi and Irina Bashkirtseva

Received: 20 December 2022
Revised: 17 January 2023
Accepted: 9 February 2023
Published: 10 February 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In recent years, with the rapid development of communication and network technology, MASs have been deeply applied in many fields, such as transportation, industrial production, etc. Facing increasingly large-scale and complex systems, the integration solutions to single-agent systems (SASs) are often limited by various resources and conditions. The MASs can effectively improve the robustness, reliability, and flexibility of large-scale complex systems [1,2].

MASs are composed of multiple agents with particular capabilities of sensing, computation, communication and control, and agents can coordinate to complete some common tasks through local interactions among agents [3,4]. Compared with traditional SASs, MASs involve relatively simple agents and thus reduce costs while improving robustness. Meanwhile, distributed coordination mechanisms exerted on multiple agents can improve the operation efficiency and reduce resource consumption. MASs have been widely used in real applications, such as resource detection, safety monitoring, natural disaster preparedness, etc. In some scenarios, agents can replace humans to guarantee the safety of military or agricultural production. In industrial applications, using multiple agents instead of single-agent can reduce production costs. Especially via coordination, such as mobile multi-unmanned aerial vehicles (Multi-UAV) systems, multi-robot systems, and multi-agent supporting systems, agents can complete more complex and challenging tasks while safety and reliability can be guaranteed [5–7].

The concerns in system control have gradually shifted from stabilization and stability to high steady-state accuracy, rapidity, strong robustness, and anti-interference performances. In many engineering application fields, scientists and engineers usually not

only want to ensure the stability of controllable systems, but also aim to optimize certain performances (energy consumption and cost) at the same time. In this way, considering optimization is a key topic with greater practical implications for MASs. That is, a group of autonomous agents set out to complete some difficult tasks while also optimizing their performance indices.

Recently, optimization and optimal control employing a preset performance criterion have become increasingly hot research topics in the system and control fields. By interacting with an environment, an agent or decision maker develops a strategy to maximize a long-term reward using reinforcement learning (RL), a goal-oriented learning technology, which has achieved great success in the field of artificial intelligence (AI) [8–10]. In this context, the ADP method with strong self-learning ability has become a promising intelligent optimization technology. At present, in the field of multi-agent optimal control, most existing ADP methods are partially model-dependent or completely model-dependent. Unfortunately, model uncertainties exist in most of actual control systems, which leads to inaccurate modeling. In order to solve this problem, model identification-based ADP methods have been developed to solve MAS optimal control problems.

Motivated by the observations mentioned above, this paper aims at giving a brief survey for important developments in model identification based optimal control and its applications to multi-agent learning and control. In particular, we mainly focus on adaptive dynamic programming based optimal control method, model identification method, and the combination of ADP and model identification for dealing with the kinds of control problems of unknown system dynamics.

2. Adaptive Dynamic Programming-Based Optimal Control Method

Adaptive Dynamic Programming (ADP) is a learning-based intelligent control method with capabilities of adaption and optimization, which has great potential in solving optimal control problems. This section mainly introduces the origin of ADP, its basic structures and the development in the field of optimal control of dynamical systems, respectively.

2.1. Basic Structures of ADP

ADP, as a fusion technology of AI and control theory, is based on the traditional optimal control theory and RL principle. ADP can effectively solve a series of complex optimal control problems by learning through the continuous interactions between the agent and the environment. It is noted that there are some synonyms for ADP, such as Approximate Dynamic Programming [11], Neuro-Dynamic Programming [12], Adaptive Critic Design [13].

In the early stage, ADP was mainly used in the fields of computer science and operational research [14] and then gradually integrated with RL technology to solve optimal control problems later. Theoretically, ADP borrows from the basic principle of RL. That is, an agent interacts with the environment and constantly adjusts its strategy to achieve the optimal cumulative feedback (return) to solve an optimal decision problem. In 1977, Werbos proposed four basic ADP structures [11,15]: Heuristic Dynamic Programming (HDP), Dual Heuristic Programming (DHP), Action Dependent HDP (ADHDP), and Action Dependent DHP (ADDHP). Generally speaking, these ADP structures mainly include an actor-critic framework with the use of neural network approximation structure, which significantly improves the online learning and adaptive abilities of ADP. The basic structure of ADP is given in Figure 1. The ADP method not only avoids the "dimensional disaster" problem in dynamic programming (DP) methods, but also provides an effective way to solve the decision control problem of complex nonlinear systems, which makes it become an important research direction in the fields of artificial intelligence and control theory [9,16].

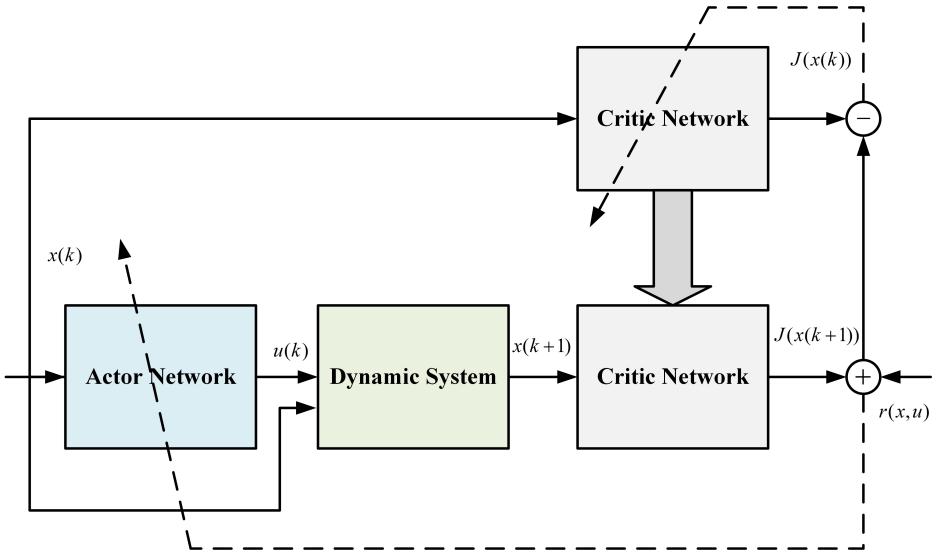

Figure 1. The basic structure of ADP.

2.2. Developments of ADP-Based Optimal Control

As an important optimal control method, ADP has been widely used in the field of optimal control. Particularly, many significant scientific research achievements have been made in early theoretical studies (including algorithm and convergence) [17]. In 2002, under the ADP framework proposed by Werbos, Murray et al. [8] firstly proposed an off-line iterative algorithm of the ADP strategy to solve an optimal control problem for nonlinear systems. At the same time, the authors offered rigorous proofs for the convergence of the iterative technique and the stability of the closed-loop system with an initial admissible control. This important theoretical result laid a solid theoretical foundation for the subsequent studies related to ADP.

The early groups engaged in ADP-related research mainly included Professor Frank L. Lewis' Team from the University of Texas at Arlington, Professor Zhongping Jiang's Team from New York University, Professor Huaguang Zhang's Team from Northeastern University, Professor Derong Liu's Team, etc. They have done much pioneering research in the field of optimal control based on ADP in the early stage. Frank L. Lewis [18] designed an ADP policy iterative algorithm to solve an input-constrained control problem for nonlinear systems. In [18], they introduced a special non-quadratic performance index function for the first time and proposed a Hamilton–Jacobi–Isaac (HJI) equation simultaneously. However, the limitation of this algorithm is that the controller design depended on the complete dynamics information of the system. To overcome this limitation, Vrabie [19] proposed a partially dynamics-dependent online optimal control algorithm based on a policy iteration, namely Integral Reinforcement Learning (IRL), for nonlinear systems with partially unknown dynamics. This algorithm parametrically represents the system's control strategy and performance using an actor-critic neural network framework, which makes the algorithm converge to the optimal control solution without requiring the system's internal dynamics, and guarantees the stability of the closed-loop system as well. After that, in order to solve a tracking control problem for partially unknown nonlinear systems, Hamidreza Modares [20] developed an IRL-based control method. The authors proposed an augmented system containing both error states and desired states, and used the augmented system to define a new non-quadratic discount performance index function.

In recent years, in order to improve the parameter updating efficiency of the actor-critic structure, Vamvoudakis [21] proposed an online policy iteration algorithm. In this algorithm, new parameter update laws were designed for the actor and critic networks, respectively, so that the two networks can realize online updates synchronously. In addition, Zhang [22] proposed a Greedy HDP iterative algorithm to solve a tracking control problem for discrete-time nonlinear systems by introducing a new tracking error performance

index function. The above research results provided an essential theoretical basis for the developments of ADP methods.

In the following, we will describe the formulation of optimal control problems for two class of nonlinear dynamical systems, that is, discrete-time system and continuous-time system, respectively.

(1) For a continuous-time nonlinear system whose dynamics are modeled as follows

$$\dot{x}(t) = f(x) + g(x)u(t), \tag{1}$$

where $f(x)$ and $g(x)$ are the system matrices. $x(t) = [x_1(t), x_2(t), \cdots, x_n(t)] \in R^n$ denotes the system state, and $u(t) = [u_1(t), u_2(t), \cdots, u_m(t)] \in R^m$ is the control input. The objective is to find an optimal controller to stabilize the system (1) as well as minimize a pre-defined performance index function, which is given by

$$V(x(t), u(t)) = \int_t^\infty r(x(\tau), u(\tau))d\tau, \tag{2}$$

where $r(x(t), u(t)) = x^\top(t)Qx(t) + u^\top(t)Ru(t)$ represents the utility function, and Q and R are symmetric positive definite matrices with appropriate dimension. It is important to assume that the control input must be admissible such that a finite performance index function can be ensured.

The Hamiltonian of the system (1) is defined as

$$H(x(t), V_x(t), u(t)) = r(x(t), u(t)) + V_x^T(f(x(t)) + g(t)u(t)), \tag{3}$$

where $V_x = \partial V/\partial x$ is a partial derivative of x.

The optimal performance index function satisfies the continuous-time HJB (CT-HJB), i.e.,

$$0 = \min_{u(t)}\{H(x(t), V_x^*(t), u(t))\}. \tag{4}$$

By applying the stationarity condition, the ideal optimal control is then given by

$$u^*(t) = -\frac{1}{2}R^{-1}g(t)^\top \frac{\partial V^*(x(t))}{\partial x(t)}. \tag{5}$$

In order to obtain the optimal controller, it is necessary to solve the CT-HJB Equation (4). However, it is very difficult to solve (4) because it contains nonlinear and partial differential items, and requires knowledge of system dynamics model $g(x)$ (that is, it needs to be known in advance). Therefore, the CT-HJB is difficult to be solved directly.

(2) For a discrete-time nonlinear system, whose dynamics is given as follows

$$x(k+1) = f(x(k), u(k)), \tag{6}$$

where $x(k)$ is system state, $u(k)$ is control input, and $k = 0, 1, 2, \ldots$ denotes the sampling index. The goal is to design a controller $u(k)$ to minimize the following performance index function

$$J(x(k), u(k)) = \sum_{j=k}^{\infty} r(x(j), u(j)), \tag{7}$$

where $r(x(j), u(j))$ denotes the utility function. By using the performance index (7), the following Bellman Equation (nonlinear Lyapunov equation) can be obtained

$$J(x(k)) = r(x(k), u(k)) + J(x(k+1), u(k+1)). \tag{8}$$

According to the Bellman's principle of optimality, the optimal performance index function satisfies the following discrete-time Hamilton–Jacobi–Bellman (DT-HJB) equation

$$J^*(x(k)) = \min_{u(k)}\{r(x(k), u(k)) + J^*(x(k+1), u(k+1))\}. \tag{9}$$

Then, we can obtain the optimal controller as

$$u^*(k) = \arg\min_{u(k)}\{r(x(k), u(k)) + J^*(x(k+1), u(k+1))\}. \tag{10}$$

It is noted from above process that the optimal controller relies on the performance index at next time step $J^*(x(k+1), u(k+1))$. No matter how, the HJB equation is the key part for computing optimal control for both discrete-time and continuous-time nonlinear systems. Thus, it is important to obtain the approximate solution to the HJB equation. In the past decades, many researchers have made great efforts to propose all kinds of iterative algorithms to deal with this issue.

2.3. ADP-Based Approximate Solution to HJB Equations

In fact, most of the research results discussed above are mainly obtained for optimal control of nonlinear systems. Theoretically, the solutions to optimal control problems for nonlinear systems usually rely on Hamilton Jacobi Bellman (HJB) Equations [18]. However, it is very difficult to compute the analytical solutions to HJB equations in general, and thus numerous researches are essentially dedicated to approximate HJB equations. Till now, from the perspective of approximate solution methods, ADP-based algorithms can be divided into two categories: Value Iteration (VI) [23,24] and Policy Iteration (PI) [18,25].

Policy Iteration (PI):

Step 1: Initialization: Initial an admissible control $u^0(t)$;

Step 2: Policy evaluation: For a given iterative control strategy $u^k(t)$, the cost function can be updated according to the following rules:

$$0 = \min_{u(k)}\{H(x(t), V_x^k(t), u^k(t))\};$$

Step 3: Strategy improvement: the iterative control strategy is updated as follows:

$$u^{k+1}(t) = -\frac{\alpha}{2}R^{-1}g(t)^\top \frac{\partial V^k(x(t))}{\partial x(t)},$$

where k is the iterative index, the policy evaluation and policy improvement are updated alternately until the performance function and control policy converge to the optimal value. In addition, for the above PI iterative algorithm, the convergence of the algorithm has been proved.

Value Iteration (VI):

Step 1: Initialization: given an any control $u^0(t)$ and $V^0(t)$;

Step 2: Policy evaluation: the control policy can be updated according to the following rule:

$$u^k(t) = \min_{u(k)}\{H(x(t), V_x^k(t), u^k(t))\};$$

Step 3: Value improvement: the index function is updated according to the following Bellman equation:

$$V^{k=1}(x(t)) = r(x(t), u^k(t)) + V^{k+1}(x(t+1)),$$

where k is the iterative index and the policy evaluation and value improvement are updated alternately until the performance function and control converge to the optimal value.

The PI algorithm starts from an initial admissible control strategy and solves a series of HJB equations to obtain the optimal control strategy. In contrast, PI has a faster convergence

rate than VI. The advantage of VI algorithm is that it does not require an initial admissible control. However, the iterative control during the iterative processing may not guarantee the stability of the closed-loop system. Al-Tamimi [23] presented a VI method (also known as greedy iterative ADP algorithm) for a discrete-time system and studied its convergence and stability under the approximation optimum controller. In [25], Liu et al. proposed a PI algorithm. Compared with other ADP algorithms, this paper presented a complete convergence analysis of the proposed PI algorithm for discrete-time nonlinear systems for the first time.

In recent decades, ADP methods have been widely concerned by academia and industry because of their theoretical research and practical application values. However, most ADP methods are partially model dependent or completely model dependent [26,27], so it is difficult to deal with the situation that accurate system information cannot be obtained. In most practical cases, the system model structure of the controlled object is unknown, or the model structure is known but the model parameters are unknown. Actually, the first consideration for the unknown model in the engineering field is to identify the model. Because accurate system models can reflect the system structure information, corresponding control strategies can then be better formulated.

From another perspective, in order to address the issue of unknown system dynamics, ADP can be divided into two main types: the indirect method and the direct method. In the direct method, the optimal control law is directly designed based on the measurable system data including the state information or input/output information without system identification process [28–30]. The indirect technique might be a significant new trend in the development of model-free optimal control, where the reconstructed system model is firstly established by approximate approaches such as neural networks (NNs) based identifiers. Then, an ADP algorithm is introduced to design an optimal controller for the approximate model. However, Modares et al. [31] have shown that the error of model identification directly affects the convergence effect of NN weights in the ADP algorithm. Therefore, the synthesis of model identification and ADP is an important trend and also a challenging issue, which has been widely attracted in this field very recently.

3. Model Identification

From the perspective of model structure, model identification methods can be divided into parametric model identification and non-parametric model identification, which will be introduced in the following, respectively.

3.1. Parametric Model Identification Method

A parametric model identification method needs to determine the model structure and order of the system in advance, and then estimates the unknown parameters of the system model. This method mainly includes the least squares method, the gradient method, the maximum likelihood estimation method, and expectation maximization method. The overview of the parametric model identification methods is illustrated in Figure 2.

Least squares methods have formed a complete theoretical system architecture and been widely applied in many model identification problems till now. Aiming at a parameter identification problem of linear-in-parameter systems with missing data, Ding et al. [32] developed an interval-varying auxiliary model based on the recursive least squares (AM-RLS) algorithm with the help of the auxiliary model identification idea. By introducing the forgetting factors, the parameter estimation accuracy and convergence rates can be improved. For the multivariable pseudo-linear autoregressive moving average (ARMA) systems, Ding et al. [33] proposed a decomposition-based least squares iterative identification algorithm. The key in the proposed algorithm is to transform the original system to a hierarchical identification model using a data filtering technique. The model was then divided into three subsystems, with each subsystem being identified separately. The proposed approach involves less processing effort than least squares-based iterative techniques. For the identification of bilinear forms, Camelia [34] proposed a recursive least-squares for

bilinear forms (RLS-BF) algorithm. Two variations of the RLS-BF algorithm based on the dichotomous coordinate descent (DCD) approach were presented to lower the computing complexity of the process. Meanwhile, a regularized version of the RLS-BF method was created to increase the resilience of the RLS-BF technique in noisy situations.

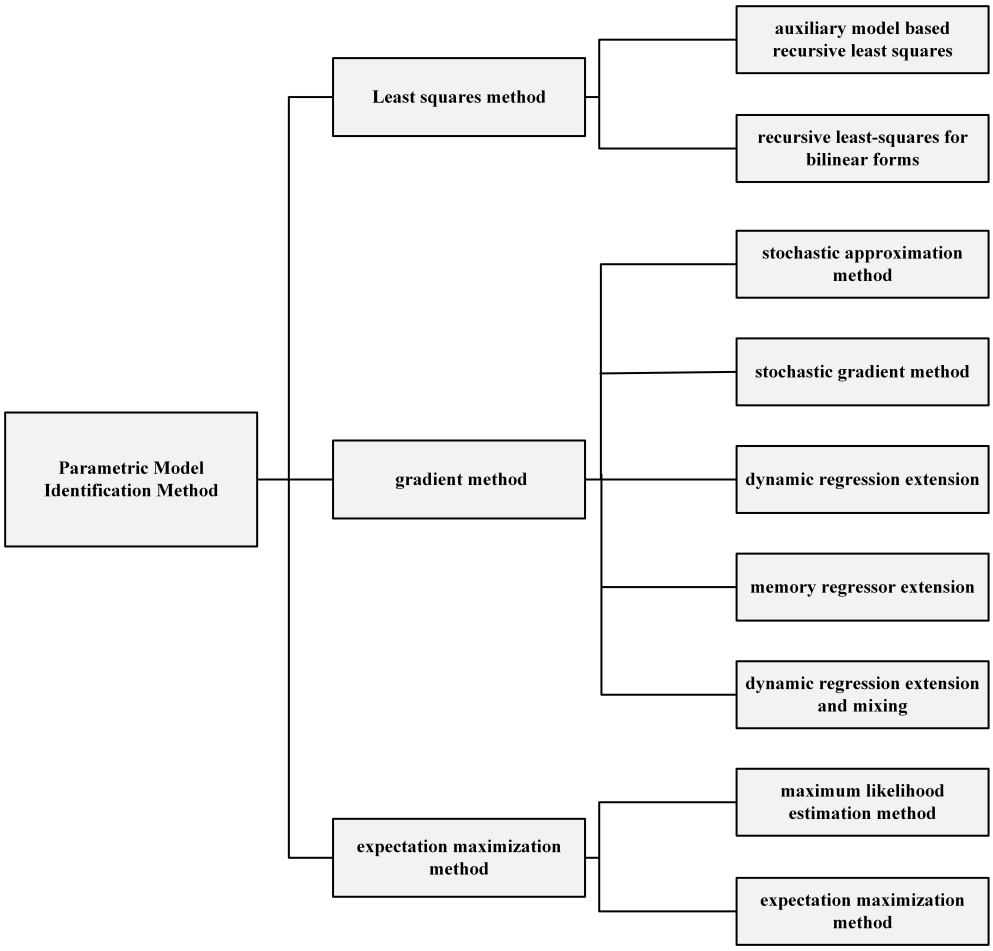

Figure 2. Overview of Parametric Model Identification Methods.

Essentially, a gradient method is an iterative algorithm. Compared with the recursive least squares, it has a slower convergence rate and lager error variance of parameter estimation. However, the computation of each step in the recursive process of gradient methods is smaller. According to the different search steps, the gradient method can be divided into the stochastic approximation method and the stochastic gradient method. There are two commonly used stochastic approximation methods, the Robbins-Monro algorithm, and the Kiefer-Wolfpwitz algorithm. However, because of the slow convergence rates of these two algorithms near the extreme points, they have not received widespread attention.

On the basis of stochastic approximation method, the stochastic gradient method adjusts the search step and accelerates the convergence rate. Recently, this method is widely used in the identification of various systems. For multivariate output-error systems, Wu [35] developed an auxiliary model based stochastic gradient (AM-SG) method and a coupled AM-SG algorithm, which ensured the parameter estimation error converged to zero under the persistence excitation (PE) condition. For the bilinear system with white noise, Ding [36] introduced a stochastic gradient (SG) technique and a gradient-based iterative approach for estimating system parameters with known input-output data using an auxiliary model. Experimental results show that the proposed gradient-based iterative algorithm has higher estimation accuracy than the auxiliary model based stochastic gradient.

In recent years, a new class of algorithms has been derived in the field of adaptive control based on the gradient method. An important concern in developing parameter identification and adaptive control schemes is transforming the original system model to a linear regressor Equation (LRE), in which the unknown parameters are linearly related to the measurable data. Then the unknown parameter estimation problem of the original system is transformed to solving the LRE, which derives a series of parameter identification methods based on the LRE of the original system.

The classical LRE can be expressed as

$$y = \phi^T \theta,$$

where $y \in R$ and $\phi(t) \in R^q$ are measurable signals. $\theta \in R^q$ is an unknown constant signal. Herein, $\phi(t)$ is also called the regression vector. Generally, we can use the least square method [37] or the gradient method [38] to solve the unknown parameters of the original system LRE. The gradient-descent based adaptive law is designed as

$$\dot{\hat{\theta}}(t) = \alpha \phi(t)[y(t) - \phi^T(t)\hat{\theta}(t)],$$

where $\hat{\theta}$ is the estimation of θ, $\alpha > 0$ presents adaptive learning gain. The idea of these two methods is to generate a linear time-varying (LTV) dynamic equation, known as the parameter error Equation (PEE) that can describe the estimation error, and then design the parameter estimator based on the PEE. However, the fundamental disadvantage of these techniques is that parameter estimation convergence is dependent on the PE condition of the regression vector.

Mathematically, the PE condition means that there exist some constants $\mathfrak{t} > 0$ and $\Delta > 0$ such that

$$\int_t^{t+\mathfrak{t}} \phi(s)\phi^T(s)ds \geq \Delta I$$

for any time t. That is, the input signal should excite all kinds of system modality so that the measurable signal contains enough information about the system, and then the convergence of parameter estimation can be guaranteed. In practice, input signals need to be designed to satisfy the PE condition. However, this is seldom practicable and difficult to verify online. Even if the input signal meets the PE criteria, the adaptive control's parameter convergence is largely reliant on the PE intensity, which leads to a slower convergence rate.

Moreover, the transient performance of these two methods is highly unpredictable and can only guarantee weak (vector norm) monotonicity of the estimation errors. Unfortunately, poor transient estimation error performance (such as significant overshoot and slow convergence in the first few seconds) may severely degrade the estimation response, resulting in identification and adaptive control instability. Therefore, engineering applications increasingly need fast, accurate, and robust parameter estimation method to maintain the security and reliability of control systems.

To improve the parameter convergence of the gradient method, most ideas are to convert the LRE of the original system into an alternative LRE to generate a new PEE with stronger convergence properties. By introducing multiple linear filter operators to apply on the LRE of the original system, Lion [39] piled up the filtered signals to generate an extended LRE. Then, a gradient estimator based on the extended LRE was proposed. The way of developing the extended LRE is called dynamic regression extension (DRE). Compared with the classical gradient estimation method, the parameter convergence rate of the DRE-based gradient estimator can be made arbitrarily fast by increasing the adaptive gain. Kreisselmeier [40] also proposed a filter method, namely memory regressor extension (MRE), to design new LREs. Unlike DRE, Kreisselmeier only applied one linear filter operator to $\phi(s)\phi^T(s)$. In fact, DRE can be transformed into MRE by rationally choosing the filter operator in the DRE algorithm. That is, MRE is a particular case of DRE. Except the advantages of the DRE-based gradient estimator, the MRE-based gradient estimator

has better estimation performance than traditional gradient estimators for systems which do not satisfy the PE conditions.

To improve the transient performance of parameter identification, some researchers advocated combining the tracking error in direct adaptive control and the identification error in indirect control to form a new PEE. Then, parameter estimation algorithms based on tracking and identification errors were successively proposed [41–43]. Duarte et al. [41] used such an approach for model reference adaptive control (MRAC) of linear time-invariant (LTI) systems and gave the name composite adaptive control. In [42], position tracking control of robot manipulators was considered with composite adaptive control. Panteley [43] applied the composite adaptive control algorithm to the adaptive control of a class of nonlinear systems with measurable states, and relaxed a rather restrictive–detectability assumption in the stability proof. Later, Lavretsky [44] applied the work of Panteley to linear systems.

The above two types of parameter estimation frameworks lay the foundation for LRE parameter estimation. Five new adaptive control methods have gradually evolved in recent years based on these two types of original system LRE parameter estimation frameworks.

For the adaptive control of linear LTI systems, Chowdhary [45] used recorded and current data concurrently to estimate unknown parameters when designing composite adaptive law. This technique is named concurrent learning. Notably, the technique does not rely on the PE condition but guarantees the global exponential stability (GES) of the closed-loop system under an interval excitation (IE) condition. Compared with the traditional PE condition, the IE condition focuses on the evolution of integrals within an interval which is strictly weaker than the PE condition.

Cho [46] and Roy [47] designed a new composite estimator by constructing residual signals. Smilar with Chowdhary, the proposed algorithm used an "offline data selection method". That is, the incoming data are first accumulated to build the information matrix. A composite estimator is designed by the full rank information matrix after sufficient but not persistent excitation.

In [48–50], a variant algorithm of MRE is proposed, which selects the filter operator as a pure integral form. Actually, this improvement leads to a positive semi-definite open-loop integral in the parameter estimator, which affects the noise sensitivity and high-gain adaptive alertness of the parameter estimator. It will make the algorithm difficult to apply in practical engineering.

Adetola [51] proposed a finite-time parameter estimation algorithm for nonlinear systems. This algorithm combines the pure integrator based MRE technique with the "initialization" process proposed in [48], and the unknown parameters of the original system can be estimated in finite time under the condition that the regression vector satisfies IE.

Aranovskiy [52] proposed a modified algorithm for DRE and named it "DRE and mixing" (DREM). The DREM algorithm adds a key mixing step to DRE and decouples vector PEEs into scalar PEEs. The scalar PEE ensures the monotonicity of each element in the parameter estimation error, which is stronger than the norm monotonicity of the traditional parameter error vector. It means the parameter estimator designed based on the scalar PEE has stronger transient stability. At the same time, the algorithm guarantees the parameter convergence and proposes a new parameter convergence condition that does not depend on the PE condition.

The least squares method and the gradient method have been developed very well, but it is difficult to address the data with missing information. Since the maximum likelihood estimation method and the expectation maximization algorithm can deal with the problem of missing information, these two algorithms have received more and more attention. The maximum likelihood estimation method proposed by Panuskal [53] is the initial probabilistic model identification method, but it did not consider the situation of missing information at that time. To deal with the parameter estimation problem in the absence of data, Dempster [54] proposed the expectation maximization algorithm. This algorithm has been used for parameter estimation of the Gaussian mixture model [55], linear variable

parameter model [56], and state space model [57], and a series of expectation maximization variants algorithms have been developed.

Notably, the parametric model identification method can describe the controlled object analytically and achieve better identification results. In the development of recent decades, a fairly complete theoretical system has been formed. However, these methods are mainly for the identification of linear systems. However, most of the controlled objects often contain many complex nonlinear uncertain items in the actual system, and their model structure parameters also show time-varying characteristics, making it impossible to obtain the accurate system dynamic model. Recently, since the non-parametric models can approximate the dynamics of arbitrary complex processes in infinite dimensions, the nonparametric identification methods have begun to become the focus of scholars.

3.2. Non-Parametric Model Identification Method

The model reconstructed by the non-parametric model identification method is called a non-parametric model. It does not mean that there are no parameters in the model but that it does not need to determine the structure and order of the model in advance, which is the advantage of the non-parametric model identification method. Non-parametric model identification methods include some classic identification methods, such as correlation analysis and spectral analysis, etc. It also includes neural network (NN) models which have been developed rapidly in recent years. A neural network has been widely used in nonlinear system control because of its high nonlinearity, approximation ability, and strong self-learning ability. At present, non-parametric model identification methods mainly include: Back-Propagation (BP) neural network non-parametric model identification and Radial Basis Function (RBF) neural network non-parametric model identification. The overview of the non-parametric model identification methods is illustrated in Figure 3.

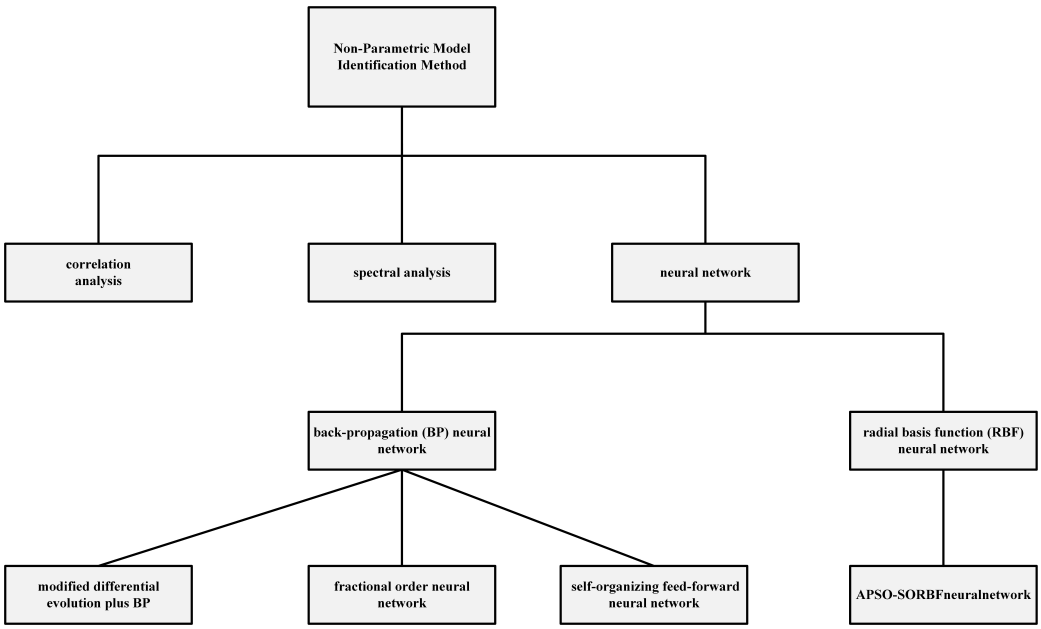

Figure 3. Overview of Non-Parametric Model Identification Methods.

For the non-parametric model identification method using the BP neural network, since the BP neural network can approximate any nonlinear mapping relationship, and the BP algorithm belongs to the global approximation algorithm, it has better generalization ability. Generally speaking, when using a neural network to identify nonlinear systems, it is often combined with classical parameter identification methods to optimize the weights of NN.

Coban [58] proposed a new recurrent neural network, the context layered locally recurrent neural network (CLLRNN), which is effective in the identification of input-output

relationships in both linear and nonlinear dynamic systems. To maximize the weights of the neural network model, Nguyen [59] proposed a hybrid modified differential evolution plus back-propagation (MDE-BP) approach. The suggested training method was evaluated in comparison to existing algorithms, including the classic DE and BP algorithms. As a result, the proposed strategy can improve the identification process's accuracy. In [60], Aguilar proposed a fractional order neural network (FONN) for system identification by combining neural network and fractional order calculus methodologies. When compared to existing techniques, the suggested FONN model achieved higher accuracy with fewer parameters. Li [61] developed a new bilevel learning paradigm for self-organizing feed-forward neural networks (FFNN). The hybrid binary particle swarm optimization (BPSO) algorithm is used as an upper level optimizer in this interactive learning algorithm to self-organize network architecture, while the Levenberg–Marquardt (LM) algorithm is used as a lower level optimizer to optimize the connection weights of an FFNN. When compared to conventional learning algorithms, experimental results show that the bilevel learning algorithm produces much more compact FFNNs with superior generalization capabilities. Singh [62] developed a gradient evolution-based counter propagation network (GE-CPN) for approximating the noncanonical form of a nonlinear system. Learning from nonlinear systems with parametric uncertainty is a key characteristic of GE-CPN networks. Furthermore, this demonstrated that reparameterization of neural network models is required and beneficial for approximation of noncanonical systems.

As a feedforward network, RBF neural network has attracted extensive attention recently because of its fixed basis function and linear parameter network structure, which can approximate any continuous function with arbitrary precision. For the identification and modeling of nonlinear dynamic systems, Qiao [63] designed a novel self-organizing radial basis function (SORBF) neural network. Based on the neuron activity and mutual information (MI), the SORBF neural network's hidden neurons can be added or removed to reach the desired network complexity while maintaining overall computing efficiency for identification and modeling. Meanwhile, parameter adjustment can considerably increase model performance. Slimani [64] utilized the descent gradient and the genetic algorithm technique to developed an optimization technique of neural networks radial basis function multi-model identification of nonlinear system. Errachdi [65] developed a no-preprocessing online radial basis function (RBF) neural network technique. The suggested online RBF neural network approach is then combined with a kernel principal component analysis (KPCA), which made RBF neural network training efficient and fast by reducing memory requirements of the models. In [66], with the use of adaptive particle swarm optimization, a self-organizing radial basis function (SORBF) neural network was constructed to increase both accuracy and parsimony (APSO). The presented APSO-SORBF neural network is capable of producing a network model with a compact structure and outstanding accuracy. In [67], to self-organize the structure and parameters of the RBFNN, a distance concentration immune algorithm (DCIA) was devised. A sample with the most frequent occurrence of maximum error was constructed to govern the parameters of the new neuron in order to increase forecasting accuracy and reduce computation time.

The above studies have introduced many mature identification algorithms from linear system identification into the RBF network framework. At the same time, many scholars have extended the RBF network framework to solve the problem of parameter model identification, which makes up for the limitations of traditional parametric model identification methods for nonlinear system identification. When the structure and order of the system model are known, even if the controlled object contains many complex nonlinear uncertain terms, or its model parameters show time-varying characteristics, the RBF neural network framework can identify it accurately. This not only makes full use of the available system information but also maximizes the accurate feature description of the original system.

4. Model Identification-Based Optimal Control for SAS

In the optimal control community, researchers are trying to introduce the model identification methods to the classical optimal control for a single agent system (SAS) with an unknown system model.

Bhasin [68] proposed an actor-critic-identifier (ACI) ADP framework, in which an identifier NN is utilized to approximate the unknown system information and then embedded into the actor-critic NN architectures. The ACI ADP framework is shown in Figure 4. However, the input system dynamics are still assumed to be known. To further remove this assumption, Modares [31] designed an new ACI algorithm for the unknown constrained-input nonlinear systems. The proposed ACI algorithm contains an identifier NN with an experience replay technique (ERT) to fully approximate the unknown system information (including system dynamics and input dynamics). Then, a gradient method was used to estimate the weights of critic-actor NNs. Actually, the idea of ERT is very similar to concurrent learning, both of which use recorded historical data and current data to estimate the unknown information of the system. Although this technique can relax the PE condition for parameter convergence during the online learning, it requires more computation time and computer memory to store historical data. The algorithm was then generalized to solve many control problems, such as the IRL algorithm for constrained input systems [69], the H_∞ tracking control problem [70], and so on.

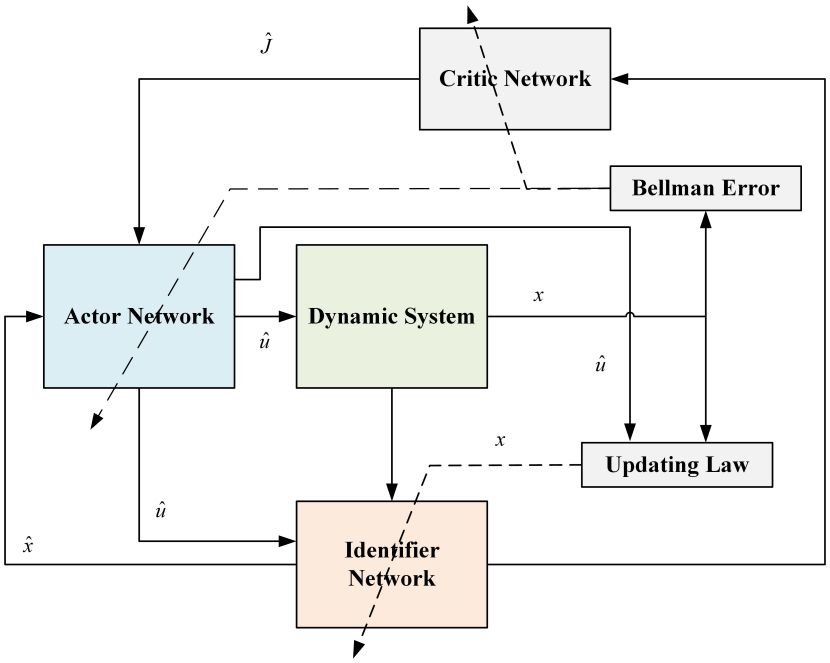

Figure 4. The basic framework of actor-critic-identifier ADP.

To relax the PE condition for parameter convergence during the online learning, Zhao [71] used the ERT to estimate the unknown weights of the identifier NN and critic NN simultaneously, so that the conventional PE condition could be relaxed to a simplified condition on recorded data. However, the proposed algorithm also has the same drawbacks in [31]. Based on the ERT, Yang [72] proposed an event-triggered robust control policy for unknown continuous-time nonlinear systems. To improve the convergence rate of the ERT, a data-based feedback relearning (FR) algorithm for uncertain nonlinear systems with control channel disturbances and actuator faults was developed [73]. Furthermore, a data processing method based on experience replay technology is designed to improve data utilization efficiency and algorithm convergence. To achieve model-free fault compensation, a neural network (NN)-based fault observer is used. To reduce the difficulty of designing NNs for an unknown nonlinear system and improve generalization, the poly-

nomial activation function is redesigned using the sigmoid function/hyperbolic tangent activation function.

To avoid excessive use of NNs and achieve faster convergence, Lv [74] proposed a new identifier-critic (IC) ADP structure with the MRE method. Since the algorithm did not use an actor NN, and it did not need to record historical data, the convergence rate is greatly improved. Later, this IC algorithm was used to solve a series of other control problems [75–77].

5. Model Identification-Based Optimal Control for MASs

More recently, few works on the model identification-based optimal control has been studied for MASs. Based on the work of Modares [31], Tatari [78] proposed an online optimal distributed learning algorithm to find the game theoretic solution of systems on graphs with completely unknown dynamics. In [79], Tatari introduced an online distributed optimal adaptive algorithm for continuous-time nonlinear differential graphical games with unknown systems subject to external disturbances. Shi [80] utilized the MRE filtering technique and designed an adaptive disturbance observer for a class of nonlinear systems with unknown disturbances where the disturbance is assumed to be generated by some unknown dynamics. Tan [81] proposed a novel event-triggered, model-free structure to address the optimal consensus control problem for MASs with unknown dynamics and input constraints.

In the following, as an example, we give the model identification-based optimal control of MASs with unknown dynamics.

Algebraic graph theory: The communication topology between agents in a MAS is described by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ where $\mathcal{V} = \{1, 2, \cdots, N\}$ is a nonempty set of vertices and $\mathcal{E} = \{(i, j) \mid i, j \in \mathcal{V}\} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. Define $\mathcal{A} = \{e_{ij}\} \in \mathbb{R}^{N\times N}$ as a weighted adjacency matrix, where $e_{ij} = 1$ if and only if $(i, j) \in \mathcal{E}$, and $e_{ij} = 0$, otherwise. The neighbor set of the agent i is denoted by $\mathcal{N}_i = \{j \mid (i, j) \in \mathcal{E}\}$. Define a diagonal matrix $\mathcal{D} = diag\{d_i\}$ as the in-degree matrix, where $d_i = \sum_{j\in\mathcal{N}_i} e_{ij}$. The Laplacian matrix $\mathcal{L}$ is defined by $\mathcal{L} = \mathcal{D} - \mathcal{A}$.

In order to take a single leader into account, we introduce an augmented graph $\overline{\mathcal{G}} = (\overline{\mathcal{V}}, \overline{\mathcal{E}})$, where $\overline{\mathcal{V}} = \{0, 1, \cdots, N\}$ and $\overline{\mathcal{E}} \subseteq \overline{V} \times \overline{V}$. A nonnegative number e_{i0} is used to describe the interaction relationship between the leader and agent i. Specifically, $e_{i0} > 0$ if agent i can receive information from the leader; otherwise, $e_{i0} = 0$. A leader adjacency matrix $\mathcal{B}$ is defined by $\mathcal{B} = \text{diag}(e_{10}, \cdots, e_{N0}) \in \mathbb{R}^{N\times N}$.

Assumption 1. *The communication interaction network $\overline{\mathcal{G}}$ has a spanning tree with the root vertex 0.*

Problem formulation: Consider heterogeneous MASs described by a linear time-invariant system as follows

$$\dot{x}_i(t) = A_i x_i(t) + B_i u_i(t),\ i = 1, 2, \ldots, N, \tag{11}$$

where $x_i(t) \in \mathbb{R}^n$ and $u_i(t) \in \mathbb{R}^m$ are the state vector and the control input vector, respectively. The system matrices $A_i \in \mathbb{R}^{n\times n}$ and input matrices $B_i \in \mathbb{R}^{n\times m}$ are assumed be unknown in this paper. Furthermore, we assume that the pairs (A_i, B_i) $(\forall i = 1, \ldots, N)$ are controllable, and the state and the control input of each agent are available.

The dynamics of the leader agent is described by

$$\dot{x}_0 = A_0 x_0, \tag{12}$$

where $x_0 \in \mathbb{R}^n$.

The local tracking error $\delta_i \in \mathbb{R}^n, i = 1, \ldots, N$ can be defined as

$$\delta_i(t) = \sum_{j\in\mathcal{N}_i} e_{ij}(x_i - x_j) + e_{i0}(x_i - x_0), \tag{13}$$

where the pinning gain $e_{i0} \geq 0$. Then, the dynamics of the local tracking error are written by

$$\begin{aligned}\dot{\delta}_i(t) &= \sum_{j\in\mathcal{N}_i} e_{ij}(\dot{x}_i - \dot{x}_j) + e_{i0}(\dot{x}_i - \dot{x}_0) \\ &= \sum_{j\in\mathcal{N}_i} e_{ij}(A_i x_i - A_j x_j) + e_{i0}(A_i x_i - A_0 x_0) + (d_i + e_{i0})B_i u_i - \sum_{j\in\mathcal{N}_i} e_{ij} B_j u_j.\end{aligned} \tag{14}$$

The overall tracking error vector is given by

$$\begin{aligned}\delta(t) =&((\mathcal{L}+\mathcal{B})\otimes I_n)(x - \hat{x}_0) \\ =&((\mathcal{L}+\mathcal{B})\otimes I_n)\zeta,\end{aligned} \tag{15}$$

where $\delta = (\delta_1^T, \delta_2^T, \cdots, \delta_N^T)^T$, $x = (x_1^T, x_2^T, \cdots, x_N^T)^T \in \mathbb{R}^{nN}$, $\hat{x}_0 = (x_0^T, x_0^T, \cdots, x_0^T)^T \in \mathbb{R}^{nN}$, $\zeta = x - \hat{x}_0$ is the global synchronization error.

One of the objectives in this paper is to design a tracking strategy to ensure that all follower agents can follow the leader, that is, $\lim_{t\to\infty} \|x_i(t) - x_0(t)\| = 0$. The second objective is to design a distributed controller that can minimize the performance index function.

In fact, under Assumption 1, $\mathcal{L} + \mathcal{B}$ is invertible. From (15), one can obtain that $\lim_{t\to\infty} \zeta(t) = 0$ if and only if $\lim_{t\to\infty} \|\delta(t)\| = 0$. Thus, once the local neighbor error approaches to zero, we can say that the tracking control problem is solved.

We define the local performance index (value function) for the agent i as follows

$$V_i(\delta_i(t)) = \frac{1}{2}\int_0^\infty (\delta_i^T Q_{ii}\delta_i + U(u_i) + \sum_{j\in\mathcal{N}_i} U(u_j))d\tau, \tag{16}$$

where $Q_{ii} \geqslant 0$ is a symmetric weight matrix, $U(\cdot) = u_i R_{ii} u_i$ is a positive definite integrand function. We assume that (16) satisfies zero-state observability.

The tracking problem is aimed at finding the Nash equilibrium policies u_i^* for the N player game. That is, for all agent i, there have $V_i^* = V_i(\delta_i(0), u_i^*, u_{\mathcal{N}_i}^*) \leq V_i(\delta_i(0), u_i, u_{\mathcal{N}_i}^*)$, $\forall u_i$, $(i = 1, \ldots, N)$. Therefore, the tracking problem of MASs with input constraint in this paper can be transformed to solving the N coupled optimization problems, that is

$$V_i^*(\delta_i(t)) = \min_{u_i} \frac{1}{2}\int_0^\infty (\delta_i^T Q_{ii}\delta_i + U(u_i) + \sum_{j\in\mathcal{N}_i} U(u_j))d\tau, \tag{17}$$

with given (14) while the dynamic informations A_i and B_i, $i = 1, \ldots, N$ are considered completely unknown.

By differentiating each value function V_i, and using (16), the following Lyapunov equation is obtained

$$\begin{aligned}\nabla V_i^T(\sum_{j\in\mathcal{N}_i} e_{ij}(A_i x_i - A_j x_j) + e_{i0}(A_i x_i - A_0 x_0) + (d_i + e_{i0})B_i u_i - \sum_{j\in\mathcal{N}_i} e_{ij} B_j u_j) + \frac{1}{2}\delta_i^T Q_{ii}\delta_i \\ + \frac{1}{2}U(u_i) + \frac{1}{2}\sum_{j\in\mathcal{N}_i} U(u_j) = 0,\end{aligned} \tag{18}$$

where $\nabla V_i = \partial V_i / \partial \delta_i \in \mathbb{R}^n$ and $V_i(0) = 0$.

Then one can get the Hamiltonian function as follows

$$\begin{aligned}H_i(\delta_i, \nabla V_i, u_i, u_{\mathcal{N}_i}) = \nabla V_i^T(\sum_{j\in\mathcal{N}_i} e_{ij}(A_i x_i - A_j x_j) + e_{i0}(A_i x_i - A_0 x_0) \\ + (d_i + e_{i0})B_i u_i - \sum_{j\in\mathcal{N}_i} e_{ij} B_j u_j) + \frac{1}{2}\delta_i^T Q_{ii}\delta_i + \frac{1}{2}U(u_i) + \frac{1}{2}\sum_{j\in\mathcal{N}_i} U(u_j).\end{aligned} \tag{19}$$

According to the first-order stationary condition in the Hamiltonians, the optimal control policy for each agent can be obtained as

$$\frac{\partial H_i}{\partial u_i} = 0 \rightarrow u_i^* = -(d_i + e_{i0})\lambda R_{ii}^{-T} B_i^T \nabla V_i^*. \tag{20}$$

System identifier using neural networks: Since the system matrices A_i and input matrices B_i are assumed to be unknown, the unknown dynamics of each agent are modeled by using NNs. Then the experience replay technique is used to formulate the identifier weights adaptive update law.

The NN-based identifiers are designed to approximate system dynamic, which is given as follows

$$A_i x_i = A_i^* x_i + \varepsilon_{A_i}, B_i u_i = B_i^* u_i + \varepsilon_{B_i}, i = 1, \ldots, N, \tag{21}$$

where $A_i^* \in ,R^{n\times n}$,$B_i^* \in \mathbb{R}^{n\times m}$ are unknown weights, $x_i \in \mathbb{R}^n$,$u_i \in \mathbb{R}^m$ are the basis functions, and ε_{A_i} and ε_{B_i} are the reconstruction errors.

Combining (21) and (11), the system can be reformulated as follows

$$\dot{x}_i = \vartheta_{A_iB_i}^* z_i(x_i, u_i) + \varepsilon_{A_iB_i}, i = 1, \ldots, N, \tag{22}$$

where $\vartheta_{A_iB_i}^* = [A_i^* \ B_i^*] \in \mathbb{R}^{n\times d}$, $z_i(x_i, u_i) = [x_i^T \ u_i^T]^T \in \mathbb{R}^d$ is the regressor vector. $\varepsilon_{A_iB_i} = \varepsilon_{A_i} + \varepsilon_{B_i}$ is the model approximation error.

Assumption 2. *On a given compact set $\Omega \subset \mathbb{R}^n$, the approximator reconstruction errors ε_{A_i} and ε_{B_i}, $i = 1, \ldots, N$ and their gradients are bounded, i.e., $\|\varepsilon_{A_i}\| \leq \bar{\varepsilon}_{A_i}$,$\|\varepsilon_{B_i}\| \leq \bar{\varepsilon}_{B_i}$, and the approximator basis functions and their gradients are bounded.*

Remark 1. *According to Assumption 2, the model approximation error $\varepsilon_{A_iB_i}$ is bounded, that is, $\|\varepsilon_{A_iB_i}\| \leq \bar{\varepsilon}_{A_iB_i} = \bar{\varepsilon}_{A_i} + \bar{\varepsilon}_{B_i}$.*

A filtered regressor is proposed for (22), which can be expressed as

$$x_i = \vartheta_{A_iB_i}^* h_i(x_i) + c l_i(x_i) + \varepsilon_{x_i}, \tag{23}$$

$$\begin{aligned} \dot{h}_i(x_i) &= -c h_i(x_i) + z(x_i, u_i), h_i(0) = 0, \\ \dot{l}_i(x_i) &= -C l_i(x_i) + x_i, l_i(0) = 0, \end{aligned} \tag{24}$$

where $C = cI_{n\times n}, c > 0$, $h_i(x_i) \in \mathbb{R}^d$ is a filtered regressor version of $z(x_i, u_i)$, $l_i(x_i) \in \mathbb{R}^n$ is a filtered regressor version of the state x_i. $\varepsilon_{x_i} = e^{-Ct}x_i(0) + \int_0^t e^{-C(t-\tau)}\varepsilon_{A_iB_i} d\tau$ is bounded, since $\varepsilon_{A_iB_i}$ is bounded. $x_i(0)$ is the initial state of (22).

To obtain the adaptive tuning law that does not affected by the system instability, both side of the filtered regressor (23) are divided by a normalizing signal $n_{s_i} = 1 + h_i^T h_i + l_i^T l_i$,

$$\bar{x}_i = \vartheta_{A_iB_i}^* \bar{h}_i(x_i) + c\bar{l}_i(x_i) + \bar{\varepsilon}_{x_i}, \tag{25}$$

where $\bar{x}_i = x_i/n_{x_i}$, $\bar{h}_i = h_i/n_{x_i}$, $\bar{l}_i = l_i/n_{x_i}$, $\bar{\varepsilon}_{x_i} = \varepsilon_{x_i}/n_{x_i}$. Obviously, $\bar{\varepsilon}_{x_i}$ is bounded.

Based on (21), (23) and (25), the form of the identifier weights estimator of agent i can be expressed as

$$\hat{\bar{x}}_i = \hat{\vartheta}_{A_iB_i} \bar{h}_i(x_i) + c\bar{l}_i(x_i), i = 1, \ldots, N \tag{26}$$

where $\hat{\vartheta}_{A_iB_i} = [\hat{A}_i \ \hat{B}_i]$ is the estimated value of the identifier weights matrix $\vartheta_{A_iB_i}^*$.

Thus, the state estimation error $e_i \in \mathbb{R}^n, i = 1, \ldots, N$ can be defined as

$$e_i(t) = \hat{\bar{x}}_i - \bar{x}_i = \tilde{\vartheta}_{A_iB_i}(t)\bar{h}_i(x_i) - \bar{\varepsilon}_{x_i}, \tag{27}$$

where $\tilde{\vartheta}_{A_iB_i}(t) = \hat{\vartheta}_{A_iB_i}(t) - \vartheta^*_{A_iB_i}(t), i = 1, \ldots, N$ is the parameter estimation error of agent i at time t.

The experience replay technique is utilized to formulate the identifier weights adaptive tuning law in the following. The idea of this technique is to store or record linearly independent historical data along with current data, so as to improve data utilization.

Then, we set

$$Z_i = [\bar{h}_i(x_i(t_1)), \ldots, \bar{h}_i(x_i(t_{p_i}))] \tag{28}$$

to be the recorded historical data stack of each agent i at the past times $t_1, \ldots, t_{p_i}$.

Remark 2. *It is noted that the number of linearly independent elements in Z_i should be equal to the dimension of the $h_i(x_i)$ in (23), i.e., $rank(Z_i) = d$. This condition aims to satisfying the PE condition and can easily be checked online.*

Then, based on the experience replay technique, a weight tuning law is designed for the identifier of agent i as follows

$$\dot{\hat{\vartheta}}_{A_iB_i}(t) = -\Gamma_i e_i(t)\bar{h}_i^T(x_i(t)) - \Gamma_i \sum_{k=1}^{p_i} e_i(t_k)\bar{h}_i^T(x_i(t_k)), \tag{29}$$

where $\Gamma_i > 0, i = 1, \ldots, N$ is a positive definite learning rate matrix.

It is noted from Remark 2 that, with the aid of experience replay technique, the PE condition can be checked by monitoring the rank of the recorded historical data, but it usually consumes large computing resources, resulting in low learning efficiency. Therefore, how to design an identification method that can take into account the learning efficiency and relaxed the PE condition is an interesting and challenging research direction.

6. Conclusions and Future Work

In this paper, we have reviewed the development of ADP-based learning optimal control, several model identification techniques, and their applications to the learning and control of MASs. Based on these reviews, it is noted that the model identification-based ADP method has made significant progress in both theoretical research and practical applications. However, the model identification-based ADP methods still have many challenges in theory and algorithm design that have not yet been resolved. Through the above summary and analysis of the model identification-based ADP methods, some related issues for future research directions are outlined as follows:

- In fact, the model identification-based ADP method is mainly focused on the design of a single controller currently, but not so much on the design of multiple controllers. It will be a very beneficial work to use the model identification-based ADP method to realize the distributed coordinated control of MASs.
- Most of the existing model identification-based ADP methods need to satisfy the PE condition. However, PE conditions are difficult to verify in practical applications. How to design a novel identification-based ADP method such that the PE condition is easier to be checked and remain low pressure [82].
- For more complex MASs such as power grids and transportation, where their accurate models cannot be obtained, the model identification-based ADP method may be used to solve large-scale practical optimization problems, which have important practical applications.

Author Contributions: Conceptualization, R.L. and Z.P.; methodology, R.L., Z.P. and J.H.; software, R.L. and Z.P.; validation, R.L. and Z.P.; investigation, R.L. and Z.P.; writing—original draft, R.L.; writing—review and editing, R.L., Z.P. and J.H.; visualization, R.L. and Z.P.; supervision, J.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 62203089, Grant 61473061, and Grant 12271083, in part by the Project funded by China Postdoctoral Science Foundation under Grant 2021M700695, and in part by the Sichuan Science and Technology Program under Grant 2022NSFSC0890, Grant 2021YFG0184 and Grant 2018GZDZX0037.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hu, J.; Liu, Z.; Wang, J.; Wang, L.; Hu, X. Estimation, intervention and interaction of multi-agent systems. *Acta Autom. Sin.* **2013**, *39*, 1796–1804. [CrossRef]
2. Ji, Y.; Wang, G.; Li, Q.; Wang, C. Event-triggered optimal consensus of heterogeneous nonlinear multi-agent systems. *Mathematics* **2022**, *10*, 4622. [CrossRef]
3. Hu, J. Second-order event-triggered multi-agent consensus control. In Proceedings of the 31th Chinese Control Conference, Hefei, China, 25–27 July 2012; pp. 6339–6344.
4. Hu, J.; Feng, G. Quantized tracking control for a multi-agent system with high-order leader dynamics. *Asian J. Control* **2011**, *13*, 988–997. [CrossRef]
5. Wang, Q.; Hu, J.; Wu, Y.; Zhao, Y. Output synchronization of wide-area heterogeneous multi-agent systems over intermittent clustered networks. *Inf. Sci.* **2023**, *619*, 263–275. [CrossRef]
6. Chen, B.; Hu, J.; Zhao, Y.; Ghosh, B.K. Finite-time velocity-free rendezvous control of multiple AUV systems with intermittent communication. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 6618–6629. [CrossRef]
7. Peng, Y.; Zhao, Y.; Hu, J. On the role of community structure in evolution of opinion formation: A new bounded confidence opinion dynamics. *Inf. Sci.* **2023**, *621*, 672–690. [CrossRef]
8. Murray, J.J.; Cox, C.J.; Lendaris, G.G.; Saeks, R. Adaptive dynamic programming. *IEEE Trans. Syst. Man Cybern. Syst.* **2002**, *32*, 140–153. [CrossRef]
9. Wang, F.Y.; Zhang, H.; Liu, D. Adaptive dynamic programming: An introduction. *IEEE Comput. Intell. Mag.* **2009**, *4*, 39–47. [CrossRef]
10. Wu, Y.; Liang, Q.; Hu, J. Optimal output regulation for general linear systems via adaptive dynamic programming. *IEEEE Trans. Cybern.* **2022**, *52*, 11916–11926. [CrossRef] [PubMed]
11. Werbos, P. *Approximate Dynamic Programming for Realtime Control and Neural Modelling*; White, D.A., Sofge, D.A., Eds., Van Nostrand: New York, NY, USA, 1992.
12. Bertsekas, D.P. *Dynamic Programming and Optimal Control*; Athena Scientific: Belmont, MA, USA, 1995.
13. Prokhorov, D.V.; Wunsch, D.C. Adaptive critic designs. *IEEE Trans. Neural Netw.* **1997**, *8*, 997–1007. [CrossRef]
14. Bellman, R. Dynamic programming. *Science* **1966**, *153*, 34–37. [CrossRef]
15. Werbos, P. Advanced forecasting methods for global crisis warning and models of intelligence. *Gen. Syst. Yearb.* **1977**, *22*, 25–38.
16. Zhang, H.-G.; Zhang, X.; Luo, Y.-H.; Yang, J. An overview of research on adaptive dynamic programming. *Acta Autom. Sin.* **2013**, *39*, 303–311. [CrossRef]
17. Lewis, F.L.; Vrabie, D. Reinforcement learning and adaptive dynamic programming for feedback control. *IEEE Circuits Syst. Mag.* **2009**, *9*, 32–50. [CrossRef]
18. AbuKhalaf, M.; Lewis, F.L. Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network hjb approach. *Automatica* **2005**, *41*, 779–791. [CrossRef]
19. Vrabie, D.; Lewis, F.L. Neural network approach to continuoustime direct adaptive optimal control for partially unknown nonlinear systems. *Neural Netw.* **2009**, *22*, 237–246. [CrossRef]
20. Modares, H.; Lewis, F.L. Optimal tracking control of nonlinear partiallyunknown constrainedinput systems using integral reinforcement learning. *Automatica* **2014**, *50*, 1780–1792. [CrossRef]
21. Vamvoudakis, K.G.; Lewis, F.L. Online actor-critic algorithm to solve the continuoustime infinite horizon optimal control problem. *Automatica* **2010**, *46*, 878–888. [CrossRef]
22. Zhang, H.; Wei, Q.; Luo, Y. A novel infinitetime optimal tracking control scheme for a class of discretetime nonlinear systems via the greedy hdp iteration algorithm. *IEEE Trans. Syst. Man Cybern. Syst. Part B (Cybernetics)* **2008**, *38*, 937–942. [CrossRef]
23. AlTamimi, A.; Lewis, F.L.; AbuKhalaf, M. Discretetime nonlinear hjb solution using approximate dynamic programming: Convergence proof. *IEEE Trans. Syst. Man Cybern. Syst. Part B (Cybernetics)* **2008**, *38*, 943–949. [CrossRef]
24. Liu, D.; Wang, D.; Zhao, D.; Wei, Q.; Jin, N. Neuralnetworkbased optimal control for a class of unknown discretetime nonlinear systems using globalized dual heuristic programming. *IEEE Trans. Autom. Sci. Eng.* **2012**, *9*, 628–634. [CrossRef]
25. Liu, D.; Wei, Q. Policy iteration adaptive dynamic programming algorithm for discretetime nonlinear systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *25*, 621–634. [CrossRef] [PubMed]

26. Kiumarsi, B.; Vamvoudakis, K.G.; Modares, H.; Lewis, F.L. Optimal and autonomous control using reinforcement learning: A survey. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *29*, 2042–2062. [CrossRef] [PubMed]
27. Hou, Z.S.; Wang, Z. From modelbased control to datadriven control: Survey, classification and perspective. *Inf. Sci.* **2013**, *235*, 3–35. [CrossRef]
28. Peng, Z.; Luo, R.; Hu, J.; Shi, K.; Nguang, S.K.; Ghosh, B.K. Optimal tracking control of nonlinear multiagent systems using internal reinforce Q-learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 4043–4055. [CrossRef] [PubMed]
29. Peng, Z.; Zhao, Y.; Hu, J.; Ghosh, B.K. Data-driven optimal tracking control of discrete-time multi-agent systems with two-stage policy iteration algorithm. *Inf. Sci.* **2019**, *481*, 189–202. [CrossRef]
30. Peng, Z.; Zhao, Y.; Hu, J.; Luo, R.; Ghosh, B.K.; Nguang, S.K. Input-output data-based output antisynchronization control of multi-agent systems using reinforcement learning approach. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7359–7367. [CrossRef]
31. Modares, H.; Lewis, F.L.; Naghibi-Sistani, M.B. Adaptive optimal control of unknown constrained-input systems using policy iteration and neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 1513–1525. [CrossRef]
32. Ding, F.; Wang, F.F. Recursive least squares identification algorithms for linear-in-parameter systems with missing data. *Control Decis.* **2016**, *31*, 2261–2266.
33. Ding, F.; Wang, F.F.; Xu, L.; Wu, M.H. Decomposition based least squares iterative identification algorithm for multivariate pseudo-linear ARMA systems using the data filtering. *J. Franklin Inst.* **2017**, *354*, 1321–1339. [CrossRef]
34. Elisei-Iliescu, C.; Stanciu, C.; Paleologu, C.; Benesty, J.; Anghel, C.; Ciochina, S. Efficient recursive least-squares algorithms for the identification of bilinear forms. *Digit. Signal Process* **2018**, *83*, 280–296. [CrossRef]
35. Huang, W.; Ding, F.; Hayat, T.; Alsaedi, A. Coupled stochastic gradient identification algorithms for multivariate output-error systems using the auxiliary model. *Int. J. Control Autom.* **2017**, *15*, 1622–1631. [CrossRef]
36. Ding, F.; Xu, L.; Meng, D.; Jin, X.-B.; Alsaedi, A.; Hayat, T. Gradient estimation algorithms for the parameter identification of bilinear systems using the auxiliary model. *J. Comput. Appl. Math.* **2020**, *369*, 112575. [CrossRef]
37. Åström, K.J.; Wittenmark, B. *Adaptive Control*; Courier Corporation: Mineola, NY, USA, 2013.
38. Hu, J.; Hu, X. Optimal target trajectory estimation and filtering using networked sensors. In Proceedings of the 27th Chinese Control Conference, Kunming, China, 16–18 July 2008; pp. 540–545.
39. Lion, P.M. Rapid identification of linear and nonlinear systems. *AIAA J.* **1967**, *5*, 1835–1842. [CrossRef]
40. Kreisselmeier, G. Adaptive observers with exponential rate of convergence. *IEEE Trans. Autom. Control* **1977**, *22*, 2–8. [CrossRef]
41. Duarte, M.A.; Narendra, K.S. Combined direct and indirect approach to adaptive control. *IEEE Trans. Autom. Control* **1989**, *34*, 1071–1075. [CrossRef]
42. Slotine, J.E.; Li, W. Composite adaptive control of robot manipulators. *Automatica* **1989**, *25*, 509–519. [CrossRef]
43. Panteley, E.; Ortega, R.; Moya, P. Overcoming the detectability obstacle in certainty equivalence adaptive control. *Automatica* **2002**, *38*, 1125–1132. [CrossRef]
44. Lavretsky, E. Combined composite model reference adaptive control. *IEEE Trans. Autom. Control* **2009**, *54*, 2692–2697. [CrossRef]
45. Chowdhary, G.; Yucelen, T.; Muhlegg, M.; Johnson, E. Concurrent learning adaptive control of linear systems with exponentially convergent bounds. *Int. J. Adapt. Control Signal Process* **2013**, *27*, 280–301. [CrossRef]
46. Cho, N.; Shin, H.; Kim, Y.; Tsourdos, A. Composite MRAC with parameter convergence under finite excitation. *IEEE Trans. Autom. Control* **2018**, *63*, 811–818. [CrossRef]
47. Roy, S.; Bhasin, S.; Kar, I. A UGES switched MRAC architecture using initial excitation. In Proceedings of the 2017 20th IFAC World Congress, Toulouse, France, 9–14 July 2017; pp. 7044–7051.
48. Krause, J.; Khargonekar, P. Parameter information content of measurable signals in direct adaptive control. *IEEE Trans. Autom. Control* **1987**, *32*, 802–810. [CrossRef]
49. Ortega, R. An on-line least-squares parameter estimator with finite convergence time. *IEEE Inst. Electr. Electron. Eng.* **1988**, *76*, 847–848. [CrossRef]
50. Roy, S.; Bhasin, S.; Kar, I. Combined MRAC for unknown MIMO LTI systems with parameter convergence. *IEEE Trans. Autom. Control* **2018**, *63*, 283–290. [CrossRef]
51. Adetola, V.; Guay, M. Finite-time parameter estimation in adaptive control of nonlinear systems. *IEEE Trans. Autom. Control* **2008**, *53*, 807–811. [CrossRef]
52. Aranovskiy, S.; Bobtsov, A.; Ortega, R.; Pyrkin, A. Performance enhancement of parameter estimator via dynamic regressor extension and mixing. *IEEE Trans. Autom. Control* **2017**, *62*, 3546–3550. [CrossRef]
53. Panuska, V.; Rogers, A.E.; Steiglitz, K. On the maximum likelihood estimation of rational pulse transfer-function parameters. *IEEE Trans. Autom. Control* **1968**, *13*, 304–305. [CrossRef]
54. Dempster, A.P.; Laird, N.M.; Rubin, D.B. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc. Series B Stat. Methodol.* **1977**, *39*, 1–22.
55. Sammaknejad, N.; Zhao, Y.; Huang, B. A review of the expectation maximization algorithm in data-driven process identification. *J. Process Control* **2019**, *73*, 123–136. [CrossRef]
56. Yang, X.; Liu, X.; Han, B. LPV model identification with an unknown scheduling variable in the presence of missing observations—A robust global approach. *IET Control Theory Appl.* **2018**, *12*, 1465–1473. [CrossRef]
57. Wang, D.; Zhang, S.; Gan, M.; Qiu J. A novel EM identification method for Hammerstein systems with missing output data. *Trans. Ind. Inform.* **2019**, *16*, 2500–2508. [CrossRef]

58. Coban, R. A context layered locally recurrent neural network for dynamic system identification. *Eng. Appl. Artif. Intell.* **2013**, *26*, 241–250. [CrossRef]
59. Nguyen, S.N.; Ho-Huu, V.A.; Ho, P.H. A neural differential evolution identification approach to nonlinear systems and modelling of shape memory alloy actuator. *Asian J. Control* **2018**, *20*, 57–70. [CrossRef]
60. Aguilar, C.J.Z.; Gómez-Aguilar, J.F.; Alvarado-Martínez, V.M.; Romero-Ugalde, H.M. Fractional order neural networks for system identification. *Chaos Solitons Fractals* **2020**, *130*, 109444. [CrossRef]
61. Li, H.; Zhang, L. A bilevel learning model and algorithm for self-organizing feed-forward neural networks for pattern classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *32*, 4901–4915. [CrossRef]
62. Singh, U.P.; Jain, S.; Tiwari, A.; Singh, R.K. Gradient evolution-based counter propagation network for approximation of noncanonical system. *Soft Comput.* **2019**, *23*, 4955–4967. [CrossRef]
63. Qiao, J.F.; Han, H.G. Identification and modeling of nonlinear dynamical systems using a novel self-organizing RBF-based approach. *Automatica* **2012**, *48*, 1729–1734. [CrossRef]
64. Slimani, A.; Errachdi, A.; Benrejeb, M. Genetic algorithm for RBF multi-model optimization for nonlinear system identification. In Proceedings of the IEEE International Conference on Control, Automation and Diagnosis, Grenoble, France, 2–4 July 2019; pp. 2–4.
65. Errachdi, A.; Benrejeb, M. Online identification using radial basis function neural network coupled with KPCA. *Int. J. Gen. Syst.* **2017**, *46*, 52–65. [CrossRef]
66. Han, H.G.; Lu, W.; Hou, Y.; Qiao, J.-F. An adaptive-PSO-based self-organizing RBF neural network. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *29*, 104–117. [CrossRef] [PubMed]
67. Qiao, J.; Li, F.; Yang, C.; Li, W.; Gu, K. A self-organizing RBF neural network based on distance concentration immune algorithm. *IEEE/CAA J. Autom. Sin.* **2019**, *7*, 276–291. [CrossRef]
68. Bhasina, S.; Kamalapurkar, R.; Johnson, M.; Vamvoudakis, K.G.; Lewis, F.L.; Dixon, W.E. A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems. *Automatica* **2013**, *49*, 82–92. [CrossRef]
69. Modares, H.; Lewis, F.L.; Naghibi-Sistani, M.-B. Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems. *Automatica* **2014**, *50*, 193–202. [CrossRef]
70. Modares, H.; Lewis, F.L.; Jiang, Z.P. H_∞ Tracking control of completely unknown continuous-time systems via off-policy reinforcement learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *26*, 2550–2562. [CrossRef] [PubMed]
71. Zhao, D.; Zhang, Q.; Wang, D.; Zhu, W. Experience replay for optimal control of nonzero-sum game systems with unknown dynamics. *IEEE Trans. Cybern.* **2015**, *46*, 854–865. [CrossRef] [PubMed]
72. Yang, X.; He, H. Adaptive critic designs for event-triggered robust control of nonlinear systems with unknown dynamics. *IEEE Trans. Cybern.* **2018**, *49*, 2255–2267. [CrossRef]
73. Mu, C.; Zhang, Y.; Sun, C. Data-Based feedback relearning control for uncertain nonlinear systems with actuator faults. *IEEE Trans. Cybern.* **2022**, 1–14. [CrossRef]
74. Lv, Y.; Na, J.; Yang, Q.; Wu, X.; Guo, Y. Online adaptive optimal control for continuous-time nonlinear systems with completely unknown dynamics. *Int. J. Control Autom.* **2016**, *89*, 99–112. [CrossRef]
75. Lv, Y.; Na, J.; Ren, X. Online H_∞ control for completely unknown nonlinear systems via an identifier–critic-based ADP structure. *Int. J. Control Autom.* **2019**, *92*, 100–111. [CrossRef]
76. Lv, Y.; Ren, X.; Na, J. Online Nash-optimization tracking control of multi-motor driven load system with simplified RL scheme. *ISA Trans.* **2020**, *98*, 251–262. [CrossRef]
77. Na, J.; Lv, Y.; Zhang, K.; Zhao, J. Adaptive identifier-critic-based optimal tracking control for nonlinear systems with experimental validation. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 459–472. [CrossRef]
78. Tatari, F.; Naghibi-Sistani, M.B.; Vamvoudakis, K.G. Distributed optimal synchronization control of linear networked systems under unknown dynamics. In Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017; pp. 668–673.
79. Tatari, F.; Vamvoudakis, K.G.; Mazouchi, M. Optimal distributed learning for disturbance rejection in networked non-linear games under unknown dynamics. *IET Control. Theory Appl.* **2018**, *13*, 2838–2848. [CrossRef]
80. Shi, J.; Yue, D.; Xie, X. Optimal leader-follower consensus for constrained-input multiagent systems with completely unknown dynamics. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 1182–1191. [CrossRef]
81. Tan, W.; Peng, Z.; Ji, H.; Luo, R.; Kuang, Y.; Hu, J. Event-triggered model-free optimal consensus for unknown multi-agent systems with input constraints. In Proceedings of the 2022 Chinese Control Conference (CCC), Hefei, China, 25–27 July 2022; pp. 4729–4734.
82. Luo, R.; Peng, Z.; Hu, J.; Ghosh, B.K. Adaptive optimal control of completely unknown systems with relaxed PE conditions. In Proceedings of the IEEE 11th Data Driven Control and Learning Systems Conference, Chengdu, China, 3–5 August 2022; pp. 836–841.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.