Intelligent adaptive optimal control using incremental model-based global dual heuristic programming subject to partial observability

Sun, Bo; van Kampen, Erik Jan

DOI
10.1016/j.asoc.2021.107153

Publication date
2021

Document Version
Final published version

Published in
Applied Soft Computing

Citation (APA)
Sun, B., & van Kampen, E. J. (2021). Intelligent adaptive optimal control using incremental model-based global dual heuristic programming subject to partial observability. *Applied Soft Computing*, *103*, Article 107153. https://doi.org/10.1016/j.asoc.2021.107153

Applied Soft Computing 103 (2021) 107153

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Intelligent adaptive optimal control using incremental model-based global dual heuristic programming subject to partial observability☆

Bo Sun *, Erik-Jan van Kampen

Department of Control and Operations, Delft University of Technology, Delft, 2629HS, The Netherlands

ARTICLE INFO

Article history:
Received 12 July 2020
Received in revised form 15 January 2021
Accepted 24 January 2021
Available online 1 February 2021

Keywords:
Partial observability
Intelligent control
Global dual heuristic programming
Artificial neural network
Adaptive optimal control

ABSTRACT

The scarcity of information regarding dynamics and full-state feedback increases the demand for a model-free control technique that can cope with partial observability. To deal with the absence of prior knowledge of system dynamics and perfect measurements, this paper develops a novel intelligent control scheme by combining global dual heuristic programming with an incremental model-based identifier. An augmented system consisting of the unknown nonlinear plant and unknown varying references is identified online using a locally linear regression technique. The actor–critic is implemented using artificial neural networks, and the actuator saturation constraint is addressed by exploiting a symmetrical sigmoid activation function in the output layer of the actor network. Numerical experiments are conducted by applying the proposed method to online adaptive optimal control tasks of an aerospace system. The results reveal that the developed method can deal with partial observability with performance comparable to the full-state feedback control, while outperforming the global model-based method in stability and adaptability.

1. Introduction

Conventional controller design for aerospace systems is commonly based on the known linearized models at different equilibrium or trimmed conditions and on PID controllers with human-scheduled gains to cover the complete operating envelope [1]. However, more complex demands such as optimization and adaptation have emerged recently, which cannot be tackled within this traditional scheme. The demand for optimization involves optimal control, in which the dynamic programming (DP) principle plays a fundamental role [2], and the Hamilton–Jacobi–Bellman (HJB) equation is often involved [3]. However, as a partial differential equation, the HJB equation is arduous to be solved analytically due to its nonlinear nature. Besides, DP-based approaches are by nature offline planning approaches in a backwards-in-time way and generally require the full knowledge of the system dynamics [4]. However, for complex systems, sometimes not only the internal dynamics, but also the information to infer its internal states can be unaccessible, i.e., full-state feedback (FSF) is no longer available [5–7]. These factors prevent traditional optimal control methods from further applications. On the other hand, adaptive control is another focal point of aerospace systems control [1,8], which is generally considered a separate paradigm from optimal control [9]. Adaptive control concentrates on how the controller can adapt to uncertain system dynamics, and changing environments and tasks, and does not feature optimality as its paramount target. Both optimal control and adaptive control can be significant for aerospace systems. Therefore, the purpose of this paper is to develop an adaptive optimal control approach so as to improve the optimal tracking performance without known system dynamics and perfect measurements.

Reinforcement learning (RL) is a class of bio-inspired artificial intelligence techniques, by which the agent improves its policy to maximize the received reward (or minimize the penalty) during interaction with the environment [10]. From a theoretical point of view, RL is closely linked with adaptive optimal control methods [11,12]. A fruitful cross fertilization of RL and control theory produces adaptive/approximate dynamic programming (ADP), whose essential goal lies in approximating the solutions of DP [13,14]. With two essential ingredients of temporal difference (TD) error and value function approximation (VFA) [12], ADP is a class of effective approaches to deal with adaptive optimal control problems. ADP divides the learning process into two parts, namely policy evaluation and policy improvement, which, in comparison to conventional DP, enables the controller to be computed forward in time and makes online computation feasible. Linear ADP (LADP) is a widely used technique to deal with linear optimal control problems with a quadratic performance index function [5,6,15,16], and an explicit

☆ The first author's Ph.D. is financially supported by China Scholarship Council, project reference number 201806290007.

* Corresponding author.
E-mail addresses: b.sun-1@tudelft.nl (B. Sun), E.vanKampen@tudelft.nl (E. van Kampen).

https://doi.org/10.1016/j.asoc.2021.107153

solution can be constructed [17]. Nevertheless, relying on the assumption that the dynamic system is linear time-invariant (LTI), LADP is not suitable for dealing with nonlinear or time-varying systems [6]. What is more, LADP is constricted to only employ a linear quadratic form cost, i.e., $\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}$, where $\mathbf{Q}$ is a positive semi-definite weight matrix and $\mathbf{R}$ is a positive definite weight matrix. This prevents LADP from further applications with different demands, such as addressing input saturation constraints [18–22], or releasing the input constraints [7,23–25], i.e., $\mathbf{R}$ is positive semi-definite.

As an expansion of LADP, adaptive critic designs (ACDs) break the linear quadratic constraints that exists in LADP, and have demonstrated impressive success in adaptive optimal control problems [26,27]. ACDs normally exploit artificial neural networks (ANNs) to approximate evaluation (critic) and improvement (actor) of the control policy, and consequently they can be applied to nonlinear system control problems with complicated rewards. Based on the information utilized by the critic network, ACDs are generally categorized as heuristic dynamic programming (HDP), dual heuristic programming (DHP), and global dual heuristic programming (GDHP) [28]. Among them, GDHP combines the information used by HDP and DHP and thus takes advantage of the two methods [24,25,29–31]. There are several structures of the critic network for GDHP [28] and the most widely used structure is the straightforward form that approximates the performance index function and its derivatives simultaneously [24,30,31]. However, this structure can introduce undesired inconsistent errors, so this paper employs explicit analytical calculations derived in [25] to eliminate these inconsistent errors.

Directly learning from unknown real systems usually requires a lot of trials or episodes [27] and may even cause disasters such as misconvergence or even divergence in some extreme cases [32]. Therefore, for complex and delicate systems, such as aerospace systems, information about state transition is required. For instance, for a time-invariant affine system, given the explicit information of control effectiveness, a convergent control policy can be generated based some assumptions [21,33]. However, sometimes system dynamics are completely unknown. Consequently, an extra structure, such as an ANN, is introduced to approximate the system model in some literature [29,31,34–36]. Because training ANNs requires some efforts before the parameters converge, the model network is trained offline and kept unchanged for online application in [29,31,35], which lacks capability to adapt if the system is changed, whereas in [34,36], the information of partial system dynamics is still required for online identification.

To tackle the limits of learning global system models and to achieve online fast adaptation, an incremental model is introduced in this paper. According to a local linear regression (LLR) technique [37], the incremental model only approximates the local dynamics of the original nonlinear system instead of the global model, on the assumption of sufficiently high sampling frequency [25]. The incremental technique has been successfully combined with various classic control methods to obtain adaptive nonlinear control approaches, such as incremental nonlinear dynamic inversion (INDI) [38] and incremental sliding mode control (ISMC) [39,40]. These approaches have shown success in reduction of the model dependency and fault tolerant, but have still not addressed the optimality. On the other hand, the synthesis of incremental techniques and ACDs leads to the incremental model-based adaptive critic designs (IACDs) [23–25]. These approaches have been applied to several flight control problems and performed well to generate adaptive optimal controllers with FSF. Nevertheless, real applications are often more complex and FSF can be unrealistic, which results in a partial observability (PO) problem. According to [6,41], the methods coping with deterministic systems and measurements are often regarded as output feedback methods [5,7,9,16,36,42,43], whereas if stochastic time-varying dynamics are involved, the control problems are linked with partially observable Markov decision processes (POMDPs) [6,41,44,45]. PO often occurs in aerospace systems, whose internal dynamics can be difficult to obtain and may be time-varying or stochastic, such as liquid sloshing in spacecraft with fuel tanks, infrared camera tracking with unpredictable target maneuvering, and unforeseen damages to aircraft structures changing system dynamics suddenly [6,41]. In [6], the incremental model is for the first time applied to a flight control problem with only tracking errors directly measurable by improving LADP, and extends the approach by combining the HDP approach in [41]. However, HDP has shown inferiority in convergence speed and control precision compared to DHP and GDHP [23,25]. In addition, the convergence of the identification technique is not analyzed in all existing literature adopting the incremental model.

The main contribution of this paper is dealing with the PO condition in adaptive optimal tracking control of unknown nonlinear systems by introducing an augmented incremental model into the GDHP algorithm, such that an incremental model-based GDHP (IGDHP) approach is developed. The principal advantage of the IGDHP approach lies in that the incremental model accelerates the online policy learning without knowing global system dynamics or offline training a model network, which allows for quick adaptation to system changes. Although some previous works are based on the incremental model [5–7,23–25,29,41], this paper discusses the convergence of the identification technique and achieves the highest 100% success ratio for the first time. The output layer of the actor network exploits a symmetrical sigmoid activation function, to satisfy the demands for tackling input saturation constraints, by multiplying an additive determined weight vector. Different from [25], this paper focuses on the PO situation, and improves the previous IGDHP approach by an augmented incremental model so as to deal with the unavailability of the information referring to inner system states and the unknown time-varying reference. The present research aims at bridging the gap between the discussed algorithms and real world systems, by taking more realistic application scenarios into consideration for verification, including sensor noises, fault-tolerant tasks, parameter variations, load disturbances, and combination with other controllers in higher level control.

The remainder of this paper is structured as follows. Section 2 presents the basic formulation of the continuous optimal tracking control problem subject to input constraints. In Section 3, the incremental technique is introduced for online identification in both FSF and PO conditions. Section 4 presents the IGDHP algorithm with explicit analytical calculations and addresses the input constraints via the actor network. Then Section 5 verifies the developed IGDHP method by applying it to various control problems of an aerospace system. Finally, the conclusion and future research are presented in Section 6.

2. Problem statement

Consider a nonlinear continuous system described by:

$$\dot{\mathbf{x}} = f[\mathbf{x}(t), \mathbf{u}(t)], \tag{1}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, and $\mathbf{u}(t) \in \mathbb{R}^m$ are the state vector and control vector, respectively, and $f[\mathbf{x}(t), \mathbf{u}(t)] \in \mathbb{R}^n$ provides the physical evaluation of the state vector over time. Assume that f is Lipschitz continuous on a set $\Omega_s \subset \mathbb{R}^n$ and that the system (1) is controllable on Ω_s.

The output of the nonlinear system is represented as:

$$\mathbf{y}(t) = h[\mathbf{x}(t)], \tag{2}$$

where $\mathbf{y}(t) \in \mathbb{R}^p$, and $h[\mathbf{x}(t)] \in \mathbb{R}^p$ is the Lipschitz continuous output function. The system is also assumed to be observable.

The problem investigated in this study is in the framework of optimal tracking control problem, so the objective of the controller is to minimize the tracking error between system output $\mathbf{y}(t)$ and reference trajectory $\mathbf{y}^{\text{ref}}(t)$, which is defined as:

$$\mathbf{e}(t) = \mathbf{y}(t) - \mathbf{y}^{\text{ref}}(t), \tag{3}$$

where $\mathbf{e}(t) \in \mathbb{R}^p$ and $\mathbf{y}^{\text{ref}}(t) \in \mathbb{R}^p$.

In the ADP scheme, the performance index function, also called cost-to-go, of optimal tracking control problem is usually presented as the cumulative sum of future costs from any initial time t:

$$J(\mathbf{e}(t), \bar{\mathbf{u}}(t:\infty)) = \int_t^\infty \gamma^{\tau-t} r(\mathbf{e}(\tau), \mathbf{u}(\tau)) \mathrm{d}\tau, \tag{4}$$

where $\bar{\mathbf{u}}(t:\infty) = \{\mathbf{u}(\tau) : t \le \tau < \infty\}$ denotes the system control produced by control law $\mu(\mathbf{e}(\tau)) \in \mathbb{R}^m$ from time instant t to ∞, $r(\mathbf{e}(t), \mathbf{u}(t)) \in \mathbb{R}$ denotes the cost at the time instant t, and $\gamma \in (0, 1]$ is the discount factor that indicates the extent to which the short-term cost or long-term cost is concerned. For simplicity, $J(\mathbf{e}(t), \bar{\mathbf{u}}(t:\infty))$ is denoted by $J(t)$ and $r(\mathbf{e}(t), \mathbf{u}(t))$ is denoted by $r(t)$ in the following part.

Input constraints are taken into account in this paper, which cannot be tackled merely by the linear quadratic cost. A non-quadratic functional is employed in [18,20,22] for regulation optimal control problems with input constraints. Nevertheless, this non-quadratic functional can relatively improve the complexity of the GDHP technique, in that the backpropagation processes need to compute partial derivatives. Moreover, in the existing standard solution to the optimal tracking control problems, a transformation is conducted with the aid of a desired control input $\mathbf{u}_\mathrm{d}(t)$ to build a regulation optimal control formation concerning the tracking error $\mathbf{e}(t)$ and the feedback input $\mathbf{u}_e(t) = \mathbf{u}(t) - \mathbf{u}_\mathrm{d}(t)$. However, as claimed in [19], it is impossible to encode the input constraints into this new control problem simply by a non-quadratic functional, since only feedback part of the control input $\mathbf{u}_e(t)$ can be directly obtained by minimizing the performance function. Therefore, a saturation function is directly imposed upon the control commands to satisfy the input constraints, which will be addressed by modifying the structure of the actor network in Section 4. In this way, the tracking problem is transformed into a regular optimal control problem subject to input constraints.

Based on TD technique [12], the cost-to-go can also be represented as:

$$J(t) = \int_t^{t+T} r(\tau)\mathrm{d}\tau + \gamma J(t+T), \tag{5}$$

where $T > 0$ is a time horizon. According to Bellman's optimality principle, the optimal cost-to-go is given as:

$$J^*(t) = \min_{\bar{\mathbf{u}}(t:t+T)} \{\int_t^{t+T} r(\tau)\mathrm{d}\tau + \gamma J^*(t+T)\}, \tag{6}$$

where $\bullet^*$ stands for the optimal value of $\bullet$. Therefore, the optimal control law can be expressed as:

$$\mu^*(\mathbf{e}(t)) = \arg\min_{\bar{\mathbf{u}}(t:t+T)} J^*(t) = \arg\min_{\bar{\mathbf{u}}(t:t+T)} \{\int_t^{t+T} r(\tau)\mathrm{d}\tau + \gamma J^*(t+T)\}. \tag{7}$$

For nonlinear systems, the solution of Eq. (6) is usually intractable to be obtained analytically. Therefore, an IGDHP algorithm is introduced to iteratively solve this optimal control problem.

3. Incremental model implementation

The IGDHP algorithm requires the information of the cost at next time instant, so the predictability of the system states is significant. This paper considers a PO situation, where although the system is observable, the only measurement is the tracking error and even the reference can be unknown and changing. This scenario can happen in real applications in the aerospace systems control problems. For instance, the docking sensors for automated transfer vehicle and International Space Station are infrared cameras, which only measure the relative distance and angles between them as the navigation information [6]. Therefore, it is desired to build a new module to provide a mapping from the system input to the observation, which will be dealt with using the incremental model in this section.

3.1. Incremental model with FSF

The derivation of the incremental model starts from the FSF condition while for all incremental techniques, the following assumption is a prerequisite:

Assumption 1. The sampling frequency is sufficiently high, i.e., the sampling time Δt is sufficiently small, and the system dynamics are relatively slow time-varying.

Remark 1. There are two important parts referring to Assumption 1. Firstly, a discrete incremental model can be introduced to represent a continuous nonlinear plant and retain high enough precision. Secondly, the discrete model does not change the properties of the original system, including controllability and observability.

It is assumed that the system is first-order continuous with respect to time at around time instant $t - \Delta t$ (denoted by t_0). Then, taking the first order Taylor series expansion and omitting higher-order terms, the system dynamics of Eq. (1) at around time instant t_0 can approximately be linearized as follows:

$$\begin{aligned}\dot{\mathbf{x}}(t) = \dot{\mathbf{x}}(t_0) &+ \mathbf{F}[\mathbf{x}(t_0), \mathbf{u}(t_0)][\mathbf{x}(t) - \mathbf{x}(t_0)] \\ &+ \mathbf{G}[\mathbf{x}(t_0), \mathbf{u}(t_0)][\mathbf{u}(t) - \mathbf{u}(t_0)] + O((\Delta\mathbf{x}(t))^2, (\Delta\mathbf{u}(t))^2),\end{aligned} \tag{8}$$

where $\mathbf{F}[\mathbf{x}(t_0), \mathbf{u}(t_0)] = \frac{\partial f^\mathsf{T}[\mathbf{x}(t), \mathbf{u}(t)]}{\partial \mathbf{x}(t)}|_{\mathbf{x}(t_0), \mathbf{u}(t_0)} \in \mathbb{R}^{n\times n}$ denotes the system transition matrix and $\mathbf{G}[\mathbf{x}(t_0), \mathbf{u}(t_0)] = \frac{\partial f^\mathsf{T}[\mathbf{x}(t), \mathbf{u}(t)]}{\partial \mathbf{u}(t)}|_{\mathbf{x}(t_0), \mathbf{u}(t_0)} \in \mathbb{R}^{n\times m}$ denotes the control effectiveness matrix. $\mathbf{F}[\mathbf{x}(t_0), \mathbf{u}(t_0)]$ and $\mathbf{G}[\mathbf{x}(t_0), \mathbf{u}(t_0)]$ are bounded due to the Lipschitz continuity of f in Eq. (1).

For ADP-based methods, the control policy cannot be determined in advance, and therefore the higher-order term can behave as a perturbation term to affect the closed-loop performance. Nevertheless, as claimed in [39,46], the higher-order term, $O((\Delta\mathbf{x}(t))^2, (\Delta\mathbf{u}(t))^2)$ satisfies:

$$\lim_{\Delta t \to 0} \|O((\Delta\mathbf{x}(t))^2, (\Delta\mathbf{u}(t))^2)\|_2 = 0, \ \ \forall \mathbf{x} \in \mathbb{R}^n, \forall \mathbf{u} \in \mathbb{R}^m, \tag{9}$$

which means the norm value of the higher-order term is negligible given sufficiently high sampling frequency. Eq. (9) also indicates that $\forall \bar{O} > 0, \exists \overline{\Delta t} > 0$, satisfies that for all $0 < \Delta t \le \overline{\Delta t}$, $\forall \mathbf{x} \in \mathbb{R}^n$, $\forall \mathbf{u} \in \mathbb{R}^m$, $\forall t \ge t_0$, $\|O((\Delta\mathbf{x}(t))^2, (\Delta\mathbf{u}(t))^2)\|_2 \le \bar{O}$, i.e., there exists a Δt that guarantees the boundedness of the higher-order term and the bound can be further diminished with the increase of the sampling frequency. Besides, the LLR technique is adopted and the linearization errors will not accumulate but only affect the local system identification. Furthermore, the real-world experiments, including the ground robot [38] and aerospace systems [40,47], have been successfully carried out based on this linearization process. Therefore, in the following part, the higher-order term is omitted for the convenience of controller design.

Assuming the states and state derivatives of the system are measurable, i.e., $\Delta\dot{\mathbf{x}}(t)$, $\Delta\mathbf{x}(t)$, $\Delta\mathbf{u}(t)$ are measurable, an incremental model can be utilized to describe the system (8):

$$\Delta\dot{\mathbf{x}}(t) \approx \mathbf{F}[\mathbf{x}(t_0), \mathbf{u}(t_0)]\Delta\mathbf{x}(t) + \mathbf{G}[\mathbf{x}(t_0), \mathbf{u}(t_0)]\Delta\mathbf{u}(t). \tag{10}$$

Despite the fact that physical systems are usually continuous, modern processors work in a discrete way, leading to discrete measurements and computations [24,25]. Consequently, given a sufficiently small sampling time Δt, based on Assumption 1, the plant model (10) can be represented approximately in a discrete form:

$$\frac{\mathbf{x}_{t+1} - \mathbf{x}_t}{\Delta t} - \frac{\mathbf{x}_t - \mathbf{x}_{t-1}}{\Delta t} \approx \mathbf{F}_{t-1}(\mathbf{x}_t - \mathbf{x}_{t-1}) + \mathbf{G}_{t-1}(\mathbf{u}_t - \mathbf{u}_{t-1}), \tag{11}$$

in which the subscript t stands for the current sampling time instant, $\mathbf{F}_{t-1} = \frac{\partial f^{\mathsf{T}}(\mathbf{x},\mathbf{u})}{\partial \mathbf{x}}|_{\mathbf{x}_{t-1},\mathbf{u}_{t-1}} \in \mathbb{R}^{n\times n}$ and $\mathbf{G}_{t-1} = \frac{\partial f^{\mathsf{T}}(\mathbf{x},\mathbf{u})}{\partial \mathbf{u}}|_{\mathbf{x}_{t-1},\mathbf{u}_{t-1}} \in \mathbb{R}^{n\times m}$ denote the system transition matrix and the input distribution matrix at time instant $t-1$ for the discretized systems, respectively. From Eq. (11), the following incremental form of the new discrete system can be obtained:

$$\Delta\mathbf{x}_{t+1} \approx (\mathbf{I}_n + \mathbf{F}_{t-1}\Delta t)\Delta\mathbf{x}_t + \mathbf{G}_{t-1}\Delta t\Delta\mathbf{u}_t, \tag{12}$$

where $\mathbf{I}_n$ denotes an identity matrix and subscript n shows its dimension.

In the FSF situation, matrices $\mathbf{F}_{t-1}$ and $\mathbf{G}_{t-1}$ can be identified online with a recursive least square (RLS) algorithm [25] and each update only requires the latest data.

3.2. Augmented incremental model

This subsection will focus on the construction of the locally incremental model using tracking error and input measurements based on the augmented state.

Considering Eq. (2), the output of the system around time instant t_0 can be linearized with Taylor expansion:

$$\mathbf{y}(t) \approx \mathbf{y}(t_0) + \mathbf{H}[\mathbf{x}(t_0)][\mathbf{x}(t) - \mathbf{x}(t_0)], \tag{13}$$

where $\mathbf{H}[\mathbf{x}(t_0)] = \frac{\partial h^{\mathsf{T}}[\mathbf{x}(t)]}{\partial \mathbf{x}(t)}|_{\mathbf{x}(t_0)} \in \mathbb{R}^{p\times n}$ denotes the observation matrix. Given a sampling time Δt, the incremental dynamics of the system output can be written as:

$$\Delta\mathbf{y}_{t+1} \approx \mathbf{H}_t\Delta\mathbf{x}_{t+1}, \tag{14}$$

in which $\mathbf{H}_t = \frac{\partial h^{\mathsf{T}}(\mathbf{x})}{\partial \mathbf{x}}|_{\mathbf{x}_t} \in \mathbb{R}^{p\times n}$ denotes the discrete observation matrix. It has been examined that, if a nonlinear system is completely observable with its output, then the system can still be regarded as deterministic [5,6], suggesting that the unmeasurable internal states can be reconstructed with the adequate observations to provide transition information [7].

Lemma 1. *Given the measured input/output data over a long-enough time horizon, $[t-N+1, t]$, $N \geq n/p$, the output increment $\Delta\mathbf{y}_{t+1}$ can uniquely be determined as follows:*

$$\Delta\mathbf{y}_{t+1} \approx \underline{\mathbf{F}}_t\overline{\Delta\mathbf{y}}_{t,N} + \underline{\mathbf{G}}_t\overline{\Delta\mathbf{u}}_{t,N}, \tag{15}$$

where $\underline{\mathbf{F}}_t \in \mathbb{R}^{p\times Np}$ denotes the extended discrete system transition matrix, $\underline{\mathbf{G}}_t \in \mathbb{R}^{p\times Nm}$ denotes the extended discrete input distribution matrix, and $\overline{\Delta\mathbf{u}}_{t,N} = [\Delta\mathbf{u}_t^{\mathsf{T}}, \Delta\mathbf{u}_{t-1}^{\mathsf{T}}, \ldots, \Delta\mathbf{u}_{t-N+1}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{R}^{Nm}$ and $\overline{\Delta\mathbf{y}}_{t,N} = [\Delta\mathbf{y}_t^{\mathsf{T}}, \Delta\mathbf{y}_{t-1}^{\mathsf{T}}, \ldots, \Delta\mathbf{y}_{t-N+1}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{R}^{Np}$ are the measured input/output data of N previous steps, respectively.

Proof. Based on Assumption 1, the new discrete system described by Eqs. (12) and (14) is observable. Then the detailed proof can be found in [5,15] and is omitted here.

If the reference signal is slow-varying in comparison to the system dynamics, then in the time horizon $[t-N+1, t]$, the increment of the reference signal can be ignored. Accordingly, considering Eqs. (3) and (15), the output tracking error at the next time instant can be written as:

$$\begin{aligned}\mathbf{e}_{t+1} &= \mathbf{y}_{t+1} - \mathbf{y}_{t+1}^{\text{ref}} \\ &\approx \mathbf{y}_t + \underline{\mathbf{F}}_t\overline{\Delta\mathbf{y}}_{t,N} + \underline{\mathbf{G}}_t\overline{\Delta\mathbf{u}}_{t,N} - (\mathbf{y}_t^{\text{ref}} + \Delta\mathbf{y}_{t+1}^{\text{ref}}) \\ &\approx \mathbf{e}_t + \underline{\mathbf{F}}_t\overline{\Delta\mathbf{y}}_{t,N} + \underline{\mathbf{G}}_t\overline{\Delta\mathbf{u}}_{t,N} \\ &\approx \mathbf{e}_t + \underline{\mathbf{F}}_t\overline{\Delta\mathbf{e}}_{t,N} + \underline{\mathbf{G}}_t\overline{\Delta\mathbf{u}}_{t,N},\end{aligned} \tag{16}$$

where $\mathbf{e}_{t+1} \in \mathbb{R}^p$ and $\mathbf{y}_{t+1}^{\text{ref}} \in \mathbb{R}^p$. However, it is impossible to directly identify Matrices $\underline{\mathbf{F}}_t$ and $\underline{\mathbf{G}}_t$ since the reference is unknown, and put another way, the system output cannot be measured separately. Furthermore, the last approximation in Eq. (16) relies on the assumption that the reference remains constant within the time horizon $[t-N+1, t]$, which can be invalid in numerous scenarios.

Consequently, a more general situation corresponding to POMDP is taken into account, and the following assumption is given:

Assumption 2. The bandwidth of the reference signal is comparable with that of the system dynamics, and the dynamics of the reference signal can be represented as:

$$\dot{\mathbf{y}}^{\text{ref}} = f^{\text{ref}}(\mathbf{y}^{\text{ref}}(t), \mathbf{y}(t)), \tag{17}$$

where f^{ref} is Lipschitz continuous on a set $\Omega_{\text{r}} \subset R^p$, and differentiable almost everywhere except for finite isolated points.

The reference signal is often independent of the system output, while in some other cases the reference can partially be determined by the system output, such as moving targets equipped with anti-tracking systems [6]. Eq. (17) provides a general reference description that can also be expressed by the time-based function, as long as the reference signal is continuous and piecewise differentiable. Similar to Eq. (12), the reference signal can be represented as a discrete incremental form by Taylor expansion and discretization:

$$\Delta\mathbf{y}_{t+1}^{\text{ref}} \approx (\mathbf{I}_p + \mathbf{F}_{t-1}^{\text{ref}}\Delta t)\Delta\mathbf{y}_t^{\text{ref}} + \mathbf{G}_{t-1}^{\text{ref}}\Delta t\Delta\mathbf{y}_t, \tag{18}$$

where $\mathbf{F}_{t-1}^{\text{ref}} = \frac{\partial f^{\text{ref}\ \mathsf{T}}(\mathbf{y}^{\text{ref}},\mathbf{y})}{\mathbf{y}^{\text{ref}}}|_{\mathbf{y}_t^{\text{ref}}} \in \mathbb{R}^{p\times p}$, and $\mathbf{G}_{t-1}^{\text{ref}} = \frac{\partial f^{\text{ref}\ \mathsf{T}}(\mathbf{y}^{\text{ref}},\mathbf{y})}{\mathbf{y}}|_{\mathbf{y}_t} \in \mathbb{R}^{p\times p}$. $\mathbf{F}_{t-1}^{\text{ref}}$ and $\mathbf{G}_{t-1}^{\text{ref}}$ can be time-varying and since f^{ref} is Lipschitz continuous, $\mathbf{F}_{t-1}^{\text{ref}}$ and $\mathbf{G}_{t-1}^{\text{ref}}$ are bounded. If the reference is independent of the controlled system, the matrix $\mathbf{G}_{t-1}^{\text{ref}}$ is a zero matrix.

Accordingly, an augmented system that consists of the system state and reference dynamics can be constructed by combining system representation Eqs. (12) and (14) and reference representation Eq. (18) [16]. Define $\mathbf{z}_t = [\mathbf{x}_t^{\mathsf{T}}, \mathbf{y}_t^{\text{ref}\ \mathsf{T}}]^{\mathsf{T}}$ and $\Delta\mathbf{z}_t = [\Delta\mathbf{x}_t^{\mathsf{T}}, \Delta\mathbf{y}_t^{\text{ref}\ \mathsf{T}}]^{\mathsf{T}}$, and then the following augmented system can be obtained:

$$\Delta\mathbf{z}_{t+1} \approx \mathcal{F}_{t-1}\Delta\mathbf{z}_t + \mathcal{G}_{t-1}\Delta\mathbf{u}_t, \tag{19}$$

and

$$\Delta\mathbf{e}_{t+1} \approx \mathcal{H}_t\Delta\mathbf{z}_{t+1}, \tag{20}$$

where $\mathcal{F}_{t-1} = \begin{bmatrix} \mathbf{I}_n + \mathbf{F}_{t-1}\Delta t & 0 \\ \mathbf{G}_{t-1}^{\text{ref}}\mathbf{H}_{t-1}\Delta t & \mathbf{I}_p + \mathbf{F}_{t-1}^{\text{ref}}\Delta t \end{bmatrix} \in \mathbb{R}^{(n+p)\times(n+p)}$, $\mathcal{G}_{t-1} = [\mathbf{G}_{t-1}^{\mathsf{T}}\Delta t, 0]^{\mathsf{T}} \in \mathbb{R}^{(n+p)\times m}$, and $\mathcal{H}_t = [\mathbf{H}_t, -\mathbf{I}_p] \in \mathbb{R}^{p\times(n+p)}$.

Hence, given the current time instant t, the increment of system state and output reference can uniquely be represented by the historical data as an augmented state equation:

$$\Delta\mathbf{z}_{t+1} \approx \tilde{\mathcal{F}}_{t-1,t-M}\Delta\mathbf{z}_{t-M+1} + \mathcal{U}_M\overline{\Delta\mathbf{u}}_{t,M}, \tag{21}$$

where $\tilde{\mathcal{F}}_{t-a,t-b} = \prod_{i=t-a}^{t-b} \mathcal{F}_i$, and $\mathcal{U}_M = [\mathcal{G}_{t-1}, \mathcal{F}_{t-1}\mathcal{G}_{t-2}, \ldots, \tilde{\mathcal{F}}_{t-1,t-M+1}\mathcal{G}_{t-M}] \in \mathbb{R}^{(n+p)\times mM}$. Similarly, the tracking error can be represented by previous data:

$$\overline{\Delta \mathbf{e}}_{t,M} \approx \bar{\mathcal{V}}_M \Delta \mathbf{z}_{t-M+1} + \bar{\mathcal{T}}_M \overline{\Delta \mathbf{u}}_{t,M}, \tag{22}$$

where $\overline{\Delta \mathbf{e}}_{t,M} = [\Delta \mathbf{e}_t^{\mathsf{T}}, \Delta \mathbf{e}_{t-1}^{\mathsf{T}}, \ldots, \Delta \mathbf{e}_{t-M+1}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{R}^{pM}$, $\bar{\mathcal{V}}_M = [(\mathcal{H}_{t-1}\tilde{\mathcal{F}}_{t-2,t-M})^{\mathsf{T}}, (\mathcal{H}_{t-2}\tilde{\mathcal{F}}_{t-3,t-M})^{\mathsf{T}}, \ldots, \mathcal{H}_{t-M}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{R}^{pM\times(n+p)}$ and

$$\bar{\mathcal{T}}_M = \begin{bmatrix} 0 & \mathcal{H}_{t-1}\mathcal{G}_{t-2} & \mathcal{H}_{t-1}\mathcal{F}_{t-2}\mathcal{G}_{t-3} & \cdots & \mathcal{H}_{t-1}\tilde{\mathcal{F}}_{t-2,t-M+1}\cdot\mathcal{G}_{t-M} \\ 0 & 0 & \mathcal{H}_{t-2}\mathcal{G}_{t-3} & \cdots & \mathcal{H}_{t-2}\tilde{\mathcal{F}}_{t-3,t-M+1}\cdot\mathcal{G}_{t-M} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \mathcal{H}_{t-M+1}\cdot\mathcal{G}_{t-M} \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{pM\times mM}.$$

Vectors $\overline{\Delta \mathbf{e}}_{t,M}$ and $\overline{\Delta \mathbf{u}}_{t,M}$ are the increments of observation and input sequences over the time interval $[t-M+1, t]$, which represent the available measured data. Then the following lemma is given:

Lemma 2. *Let the augmented system described by Eqs. (19) and (20) be observable. Then the increment of the system state and output reference are determined uniquely in terms of the previous data sequences over a sufficiently long time horizon* $[t-M+1, t]$, $M \geq (n+p)/p$.

Proof. Since the augmented system is observable, there exists a K, the observability index, such that $rank(\bar{\mathcal{V}}_M) < n+p$ for $M < K$, and that $rank(\bar{\mathcal{V}}_M) = n+p$ for $M \geq K$. Note that $K \geq (n+p)/p$. Therefore, let $M \geq K$, and there exists a matrix $\bar{\mathbf{M}} \in \mathbb{R}^{(n+p)\times pM}$ such that:

$$\tilde{\mathcal{F}}_{t-1,t-M} = \bar{M}\bar{\mathcal{V}}_M. \tag{23}$$

Since $\bar{\mathcal{V}}_M$ has a full column rank, and its left inverse $\bar{\mathcal{V}}_M^{\text{left}}$ is given by:

$$\bar{\mathcal{V}}_M^{\text{left}} = \left(\bar{\mathcal{V}}_M^{\mathsf{T}}\bar{\mathcal{V}}_M\right)^{-1}\bar{\mathcal{V}}_M^{\mathsf{T}}, \tag{24}$$

then

$$\bar{\mathbf{M}} = \tilde{\mathcal{F}}_{t-1,t-M}\bar{\mathcal{V}}_M^{\text{left}} + \mathbf{Z}(\mathbf{I}_{n+p} - \bar{\mathcal{V}}_M\bar{\mathcal{V}}_M^{\text{left}}) \equiv \bar{\mathbf{M}}_0 + \bar{\mathbf{M}}_1 \tag{25}$$

holds for any matrix $\mathbf{Z}$, with $\bar{\mathbf{M}}_0$ denoting the minimum norm operator and $P(\mathbb{R}^{\perp}(\bar{\mathcal{V}}_M)) = \mathbf{I}_{n+p} - \bar{\mathcal{V}}_M\bar{\mathcal{V}}_M^{\text{left}}$ being the projection onto a range perpendicular to $\bar{\mathcal{V}}_M$ [15].

Note that $\tilde{\mathcal{F}}_{t-1,t-M}\Delta \mathbf{z}_{t-M+1} = \bar{\mathbf{M}}\bar{\mathcal{V}}_M\Delta \mathbf{z}_{t-M+1}$ so that, according to Eq. (22),

$$\tilde{\mathcal{F}}_{t-1,t-M}\Delta \mathbf{z}_{t-M+1} = \bar{\mathbf{M}}\bar{\mathcal{V}}_M\Delta \mathbf{z}_{t-M+1} \approx \bar{\mathbf{M}}\overline{\Delta \mathbf{e}}_{t,M} - \bar{\mathbf{M}}\bar{\mathcal{T}}_M\overline{\Delta \mathbf{u}}_{t,M}, \tag{26}$$

$$(\bar{\mathbf{M}}_0 + \bar{\mathbf{M}}_1)\bar{\mathcal{V}}_M\Delta \mathbf{z}_{t-M+1} \approx (\bar{\mathbf{M}}_0 + \bar{\mathbf{M}}_1)\overline{\Delta \mathbf{e}}_{t,M} - (\bar{\mathbf{M}}_0 + \bar{\mathbf{M}}_1)\bar{\mathcal{T}}_M\overline{\Delta \mathbf{u}}_{t,M}. \tag{27}$$

Note, however, that $\bar{\mathbf{M}}_1\bar{\mathcal{V}}_M = 0$ so that $\bar{\mathbf{M}}\bar{\mathcal{V}}_M\Delta \mathbf{z}_{t-M+1} = \bar{\mathbf{M}}_0\bar{\mathcal{V}}_M\Delta \mathbf{z}_{t-M+1}$, and apply $\bar{\mathbf{M}}_1$ to Eq. (22), then

$$\bar{\mathbf{M}}_1\overline{\Delta \mathbf{e}}_{t,M} - \bar{\mathbf{M}}_1\bar{\mathcal{T}}_M\overline{\Delta \mathbf{u}}_{t,M} \approx 0. \tag{28}$$

Therefore,

$$\tilde{\mathcal{F}}_{t-1,t-M}\Delta \mathbf{z}_{t-M+1} = \bar{\mathbf{M}}_0\bar{\mathcal{V}}_M\Delta \mathbf{z}_{t-M+1} \approx \bar{\mathbf{M}}_0\overline{\Delta \mathbf{e}}_{t,M} - \bar{\mathbf{M}}_0\bar{\mathcal{T}}_M\overline{\Delta \mathbf{u}}_{t,M} \tag{29}$$

independently of $\bar{\mathbf{M}}_1$. Then from Eq. (21), it can be obtained that:

$$\Delta \mathbf{z}_{t+1} \approx \bar{\mathbf{M}}_0\overline{\Delta \mathbf{e}}_{t,M} + (\mathcal{U}_M - \bar{\mathbf{M}}_0\bar{\mathcal{T}}_M)\overline{\Delta \mathbf{u}}_{t,M}. \tag{30}$$

This result expresses the increment of the system state and reference $\Delta \mathbf{z}_{t+1}$ in terms of the inputs and observations from time instant $t-N+1$ to time instant t, which ends the proof.

Lemma 2 provides a deterministic relationship between the historical data and future states. To build a direct mapping from the historical observations and inputs to the future observations regardless of the inner states, the following theorem is presented based on Lemma 2:

Theorem 1. *Let the augmented system described by Eqs. (19) and (20) be observable. The tracking error increment* $\Delta \mathbf{e}_{t+1}$ *can be determined uniquely from the observations and control inputs over a sufficiently long time horizon,* $[t-M+1, t]$, $M \geq (n+p)/p$:

$$\Delta \mathbf{e}_{t+1} \approx \underline{\mathcal{F}}_t\overline{\Delta \mathbf{e}}_{t,M} + \underline{\mathcal{G}}_t\overline{\Delta \mathbf{u}}_{t,M}, \tag{31}$$

where $\underline{\mathcal{F}}_t = \mathcal{H}_t\tilde{\mathcal{F}}_{t-1,t-M}\bar{\mathcal{V}}_M^{left} \in \mathbb{R}^{p\times Mp}$ *is the augmented transition matrix, and* $\underline{\mathcal{G}}_t = (\mathcal{H}_t\mathcal{U}_M - \mathcal{H}_t\tilde{\mathcal{F}}_{t-1,t-M}\bar{\mathcal{V}}_M^{left}\bar{\mathcal{T}}_M) \in \mathbb{R}^{p\times Mm}$ *is the augmented input distribution matrix.*

Proof. Substitute Eq. (30) into Eq. (20) and the dynamics of the measurement can directly be obtained:

$$\Delta \mathbf{e}_{t+1} \approx \mathcal{H}_t\tilde{\mathcal{F}}_{t-1,t-M}\bar{\mathcal{V}}_M^{\text{left}}\overline{\Delta \mathbf{e}}_{t,M} + (\mathcal{H}_t\mathcal{U}_M - \mathcal{H}_t\tilde{\mathcal{F}}_{t-1,t-M}\bar{\mathcal{V}}_M^{\text{left}}\bar{\mathcal{T}}_M)\overline{\Delta \mathbf{u}}_{t,M}. \tag{32}$$

This completes the proof.

Theorem 1 has a similar form to Lemma 1 but includes the reference signal in its representation, which enables the incremental model predict tracking error without knowing the reference function. Matrices $\underline{\mathcal{F}}_t$ and $\underline{\mathcal{G}}_t$ in Eq. (31) can be identified using the RLS algorithm and then the one-step prediction of the tracking error can be made as:

$$\hat{\mathbf{e}}_{t+1} = \mathbf{e}_t + \hat{\underline{\mathcal{F}}}_{11,t}\Delta \mathbf{e}_t + \hat{\underline{\mathcal{F}}}_{12,t}\overline{\Delta \mathbf{e}}_{t-1,M-1} + \hat{\underline{\mathcal{G}}}_{11,t}\Delta \mathbf{u}_t + \hat{\underline{\mathcal{G}}}_{12,t}\overline{\Delta \mathbf{u}}_{t-1,M-1}, \tag{33}$$

where $\hat{\cdot}$ stands for the estimated or approximated value, $\hat{\underline{\mathcal{F}}}_{11,t} \in \mathbb{R}^{p\times p}$ and $\hat{\underline{\mathcal{F}}}_{12,t} \in \mathbb{R}^{p\times(M-1)p}$ are partitioned matrices from $\hat{\underline{\mathcal{F}}}_t$, and $\hat{\underline{\mathcal{G}}}_{11,t} \in \mathbb{R}^{p\times m}$ and $\hat{\underline{\mathcal{G}}}_{12,t} \in \mathbb{R}^{p\times(M-1)m}$ are partitioned matrices from $\hat{\underline{\mathcal{G}}}_t$.

In this way, the original continuous non-affine system is transformed approximately into a new discrete affine system, based on which, the IGHDP algorithm can design the control increment $\Delta \mathbf{u}_t$.

3.3. Online identification with RLS algorithm

A RLS algorithm is applied to the pending matrices $\underline{\mathcal{F}}_t$ and $\underline{\mathcal{G}}_t$ online [6,25]. For convenience, Eq. (31) is represented a row-by-row form as follows:

$$\Delta \mathbf{e}_{t+1}^{\mathsf{T}} \approx \begin{bmatrix} \overline{\Delta \mathbf{e}}_{t,M}^{\mathsf{T}} & \overline{\Delta \mathbf{u}}_{t,M}^{\mathsf{T}} \end{bmatrix} \cdot \begin{bmatrix} \underline{\mathcal{F}}_t^{\mathsf{T}} \\ \underline{\mathcal{G}}_t^{\mathsf{T}} \end{bmatrix}. \tag{34}$$

Define $\bar{\mathbf{x}}_t = [\overline{\Delta \mathbf{e}}_{t,M}^{\mathsf{T}}, \overline{\Delta \mathbf{u}}_{t,M}^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{R}^{M(p+m)\times 1}$ as the input information of the augmented incremental model identification, and $\underline{\Theta}_t = [\underline{\mathcal{F}}_t, \underline{\mathcal{G}}_t]^{\mathsf{T}} \in \mathbb{R}^{M(p+m)\times p}$ as the pending augmented matrix to be determined using the RLS algorithm.

A sliding window technique is employed to store sufficient historical data for online identification [7,37]. Considering the demands for fast computation, identification and adaptation, the width of data window should be as small as possible with guaranteed accuracy. Consequently, according to Lemma 2 and Theorem 1, let $M = (n+p)/p$.

The main procedure of the RLS algorithm is presented as follows [24,48]:

$$\Delta \hat{\mathbf{e}}_{t+1}^{\mathsf{T}} = \bar{\mathbf{x}}_t^{\mathsf{T}}\hat{\underline{\Theta}}_{t-1}, \tag{35}$$

$$\epsilon_t = \Delta \mathbf{e}_{t+1}^{\mathsf{T}} - \Delta \hat{\mathbf{e}}_{t+1}^{\mathsf{T}}, \tag{36}$$

$$\hat{\underline{\Theta}}_t = \hat{\underline{\Theta}}_{t-1} + \frac{\underline{\mathrm{Cov}}_{t-1}\bar{\mathbf{x}}_t}{\gamma_{\mathrm{RLS}} + \bar{\mathbf{x}}_t^{\mathrm{T}}\underline{\mathrm{Cov}}_{t-1}\bar{\mathbf{x}}_t}\epsilon_t, \tag{37}$$

$$\underline{\mathrm{Cov}}_t = \frac{1}{\gamma_{\mathrm{RLS}}}\left(\underline{\mathrm{Cov}}_{t-1} - \frac{\underline{\mathrm{Cov}}_{t-1}\bar{\mathbf{x}}_t\bar{\mathbf{x}}_t^{\mathrm{T}}\underline{\mathrm{Cov}}_{t-1}}{\gamma_{\mathrm{RLS}} + \bar{\mathbf{x}}_t^{\mathrm{T}}\underline{\mathrm{Cov}}_{t-1}\bar{\mathbf{x}}_t}\right), \tag{38}$$

where $\epsilon_t \in \mathbb{R}^p$ denotes the prediction error, $\underline{\mathrm{Cov}}_t \in \mathbb{R}^{(p+m)M\times(p+m)M}$ stands for the estimation covariance matrix, which is symmetric and positive definite, and $\gamma_{\mathrm{RLS}} \in (0, 1]$ is the forgetting factor in the RLS algorithm. It is noted that $\Delta\hat{\mathbf{e}}_{t+1}$ is approximated by $\hat{\underline{\Theta}}_{t-1}$ during the implementation because $\hat{\underline{\Theta}}_t$ is obtained by the RLS algorithm after $\Delta\mathbf{e}_{t+1}$ is observed.

Assumption 1 implies that in a certain horizon $\mathscr{A} = [1, t]$, $M \le t \le P$, $M \ll P < \infty$, the slowly varying augmented system dynamics can be regarded as a linear plant with constant pending parameters. Hence, based on the following assumption [48], the locally approximate convergence of the RLS algorithm is analyzed.

Assumption 3. For the locally linear system (33), in the local domain $\mathscr{A}$, the observed vectors $\bar{\mathbf{x}}_M, \ldots, \bar{\mathbf{x}}_t$ constitute the samples of an ergodic process, such that the time averages can be utilized. The unmodeled dynamics noises within one sliding window are formulated as a zero-mean white noise vector as:

$$\Delta\mathbf{e}_{t+1}^{\mathrm{T}} = \bar{\mathbf{x}}_t^{\mathrm{T}}\underline{\Theta} + \mathbf{e}_{o,t}, \tag{39}$$

where $\mathbf{e}_{o,t}$ is the equivalent plant noise independent of the samples $\bar{\mathbf{x}}_t$.

Theorem 2. *If Assumptions 1–3 hold, and the RLS algorithm is implemented using Eqs. (35)–(38), the approximate augmented matrix $\hat{\underline{\Theta}}_t$ has the trend of converging to the locally optimal matrix $\underline{\Theta}$.*

Proof. Because the optimal augmented matrix $\underline{\Theta}$ is valid over $\mathscr{A}$, the previous observations can uniformly be written as:

$$\Delta\mathbf{E}_{t+1}^{\mathrm{T}} = \bar{\mathbf{X}}_t^{\mathrm{T}}\underline{\Theta} + \mathbf{E}_{o,t}, \tag{40}$$

where $\Delta\mathbf{E}_{t+1} = [\Delta\mathbf{e}_{M+1}, \ldots, \Delta\mathbf{e}_{t+1}]$, $\bar{\mathbf{X}}_t = [\bar{\mathbf{x}}_M, \ldots, \bar{\mathbf{x}}_t]$, and $\mathbf{E}_{o,t} = [\mathbf{e}_{o,M}, \ldots, \mathbf{e}_{o,t}]$. The PE condition is indispensable for convergence analysis, which guarantees $\bar{\mathbf{X}}_t\bar{\mathbf{X}}_t^{\mathrm{T}}$ is positive definite. According to [48], it can be obtained that $\underline{\mathrm{Cov}}_t^{-1} = \bar{\mathbf{X}}_t\Gamma_t\bar{\mathbf{X}}_t^{\mathrm{T}}$, where $\Gamma_t = \mathrm{diag}([\gamma_{\mathrm{RLS}}^{t-M}, \gamma_{\mathrm{RLS}}^{t-M-1}, \ldots, 1])$, and $\mathrm{diag}(\bullet)$ reshapes the vector to a diagonal matrix. Therefore, the approximate augmented matrix $\hat{\underline{\Theta}}_t$ can be represented as:

$$\hat{\underline{\Theta}}_t = \underline{\Theta} + \tilde{\underline{\Theta}}_t = \underline{\Theta} + \underline{\mathrm{Cov}}_t\bar{\mathbf{X}}_t\Gamma_t\mathbf{E}_{o,t}, \tag{41}$$

where $\tilde{\underline{\Theta}}_t$ is the approximate error vector.

Define the approximate error correlation matrix as:

$$\hat{L}_t = E(\tilde{\underline{\Theta}}_t\tilde{\underline{\Theta}}_t^{\mathrm{T}}), \tag{42}$$

where $E(\bullet)$ is the expectation operation. Substituting Eq. (41) into Eq. (42), and noticing both $\underline{\mathrm{Cov}}_t$ and Γ_t are symmetrical matrices, we can obtain that:

$$\hat{L}_t = E(\underline{\mathrm{Cov}}_t\bar{\mathbf{X}}_t\Gamma_t\mathbf{E}_{o,t}\mathbf{E}_{o,t}^{\mathrm{T}}\Gamma_t\bar{\mathbf{X}}_t^{\mathrm{T}}\underline{\mathrm{Cov}}_t). \tag{43}$$

Recalling the independence of $\mathbf{e}_{o,t}$ and $\bar{\mathbf{x}}_t$, and the weight noise property of $\mathbf{e}_{o,t}$ yields:

$$\hat{L}_t = E(\underline{\mathrm{Cov}}_t\bar{\mathbf{X}}_t\Gamma_t E(\mathbf{E}_{o,t}\mathbf{E}_{o,t}^{\mathrm{T}})\Gamma_t\bar{\mathbf{X}}_t^{\mathrm{T}}\underline{\mathrm{Cov}}_t) = \sigma_o^2 E(\underline{\mathrm{Cov}}_t\underline{\mathrm{Cov}}_{2,t}^{-1}\underline{\mathrm{Cov}}_t), \tag{44}$$

where σ_o^2 is the variance of $\mathbf{e}_{o,t}$, and $\underline{\mathrm{Cov}}_{2,t}^{-1} = \bar{\mathbf{X}}_t\Gamma_t^2\bar{\mathbf{X}}_t^{\mathrm{T}}$.

Rigorous evaluation of Eq. (44) is intractable. Hence, Assumption 3 is utilized to facilitate an approximate evaluation of $\hat{L}_t$ [48].

It can be found that $\underline{\mathrm{Cov}}_t^{-1}$ is a weighted sum of the outer products $\bar{\mathbf{x}}_t\bar{\mathbf{x}}_t^{\mathrm{T}}, \cdots, \bar{\mathbf{x}}_M\bar{\mathbf{x}}_M^{\mathrm{T}}$. Therefore, based on Assumption 3, the following approximation holds:

$$\underline{\mathrm{Cov}}_t^{-1} \approx \frac{1-\gamma_{\mathrm{RLS}}^{t-M+1}}{1-\gamma_{\mathrm{RLS}}}\mathbf{E}_o, \tag{45}$$

where $\mathbf{E}_o = E(\bar{\mathbf{x}}_t\bar{\mathbf{x}}_t^{\mathrm{T}})$ is the correlation matrix of observations. If the PE condition is satisfied, $\bar{\mathbf{x}}_t\bar{\mathbf{x}}_t^{\mathrm{T}}$ is positive definite and $\mathbf{E}_o^{-1}$ can be expected.

Substituting Eq. (45) into Eq. (44) yields:

$$\begin{aligned}\hat{L}_t &= \sigma_o^2\left(\frac{1-\gamma_{\mathrm{RLS}}}{1-\gamma_{\mathrm{RLS}}^{t-M+1}}\right)^2\cdot\frac{1-\gamma_{\mathrm{RLS}}^{2(t-M+1)}}{1-\gamma_{\mathrm{RLS}}^2}\\ \mathbf{E}_o^{-1} &= \sigma_o^2\frac{1-\gamma_{\mathrm{RLS}}}{1+\gamma_{\mathrm{RLS}}}\cdot\frac{1+\gamma_{\mathrm{RLS}}^{t-M+1}}{1-\gamma_{\mathrm{RLS}}^{t-M+1}}\mathbf{E}_o^{-1}.\end{aligned} \tag{46}$$

In the steady state, i.e., $t \to P \to \infty$, the following equation holds:

$$\hat{L}_P = \sigma_o^2\frac{1-\gamma_{\mathrm{RLS}}}{1+\gamma_{\mathrm{RLS}}}\mathbf{E}_o^{-1}. \tag{47}$$

It can be found that if γ_{RLS} is very close to 1, then $\hat{L}_P \to 0$, which means the approximate augmented matrix converges to the optimal matrix. This completes the proof.

4. The IGDHP algorithm

Since the incremental model discretely identifies the system dynamics, it is also necessary to design the controller in a discrete manner. It can be found that the optimal cost-to-go (6) and optimal control law (7), which are presented in the continuous domain, have similar forms of discrete representation [12]. Letting the time horizon in Eqs. (6) and (7) be equivalent to the sampling time, i.e., $T = \Delta t$, we can discretize Eqs. (6) and (7) as:

$$J_t^* = \min_{\mathbf{u}_t}\{r_t + \gamma J_{t+1}^*\}, \tag{48}$$

and

$$\mu^*(\mathbf{e}(t)) = \arg\min_{\mathbf{u}_t} J_t^* = \arg\min_{\mathbf{u}_t}\{r_t + \gamma J_{t+1}^*\}, \tag{49}$$

where r_t is the one-step cost function of the discrete-time design and can still be formulated as a linear quadratic form:

$$r_t = \mathbf{e}(t)^{\mathrm{T}}\mathbf{Q}\mathbf{e}(t) + \mathbf{u}(t)^{\mathrm{T}}\mathbf{R}\mathbf{u}(t), \tag{50}$$

where both $\mathbf{Q} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{m\times m}$ are positive semi-definite. The weight matrix $\mathbf{R}$ is used to control the energy cost and note that it does not have to be positive definite. With the incremental model technique, the IGDHP algorithm can iteratively solve this discrete-time optimal control problem with an actor–critic scheme. Based on current information, the actor network generates control inputs for both real system and plant model. The incremental model predicts the tracking errors at the next time instant, which are utilized by the critic network to approximate cost-to-go, whose derivatives are computed analytically. The structure of the IGDHP algorithm is illustrated in Fig. 1.

For simplicity, the introduced ANNs in both the critic and actor networks are fully connected and feedforward, and consist of only three layers of nodes: an input layer, a hidden layer and an output layer. The activation function employed in the input layer is a unit-proportion linear function and in the hidden layer is a symmetrical sigmoid function, which is denoted by σ. In the following detailed implementations, the variables or pathways corresponding to the critic and actor networks and the incremental model are denoted by the subscripts c, a, and m, respectively.

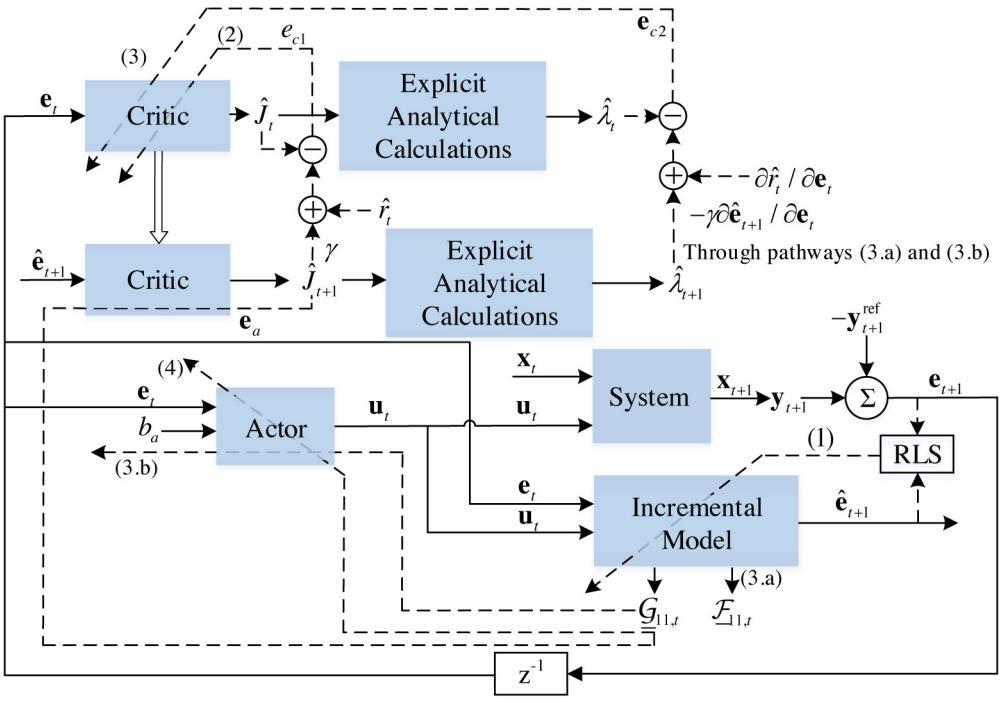

Fig. 1. The architecture of the IGDHP algorithm with PO taken into consideration, where solid lines represent the feedforward flow of signals, dashed lines are backpropagation pathways, and the thick arrow represents the weight transmission.

4.1. The critic network

Although the one-step cost function consists of $\mathbf{e}_t$ and $\mathbf{u}_t$, only $\mathbf{e}_t$ is set to be the input of the critic network. The main reason is that introducing $\mathbf{u}_t$ as an input will significantly improve the complexity of the backpropagation. Besides, $\mathbf{u}_t$ is also derived from $\mathbf{e}_t$ by the actor network, and can still affect the approximation in an indirect way through the cost function. There is no bias term in the critic input, because the critic output is desired to be 0 when $\mathbf{e}_t$ is a zero vector in this case.

In the structure with explicit analytical calculations, the critic network only directly outputs the approximated cost-to-go $\hat{J}_t$, and its output layer utilizes a unit-proportion linear activation function:

$$\hat{J}_t = \mathbf{w}_{c2,t}^{\mathrm{T}}\sigma(\mathbf{w}_{c1,t}^{\mathrm{T}}\mathbf{e}_t), \tag{51}$$

where $\mathbf{w}_{c1,t}$ and $\mathbf{w}_{c2,t}$ denote the critic weights between input layer and hidden layer, and the weights between hidden layer and output layer, respectively, and $\sigma(\cdot)$ denotes the activation function of the hidden layer. Then explicit analytical calculations [25] are carried out to compute $\hat{\lambda}_t$ directly using $\hat{J}_t$:

$$\hat{\lambda}_t = \frac{\partial \hat{J}_t}{\partial \mathbf{e}_t} = \mathbf{w}_{c1,t}(\mathbf{w}_{c2,t} \odot \sigma'(\mathbf{w}_{c1,t}^{\mathrm{T}}\mathbf{e}_t)), \tag{52}$$

where $\odot$ is the Hadamard product, and $\sigma'(\cdot)$ denotes the first order derivative of function $\sigma(\cdot)$.

A TD method is introduced to iteratively update the critic network [10], whose principle is trying to minimize the TD error, the error between the current and successive estimates of the state value. Similar to the transformation of the optimal cost-to-go, Eq. (5) can also be transformed into a discrete form:

$$J_t = r_t + \gamma J_{t+1}, \tag{53}$$

Therefore the TD errors produced by the critic network are given as follows:

$$e_{c1,t} = \hat{J}_t - r_t - \gamma \hat{J}_{t+1}, \tag{54}$$

and

$$\mathrm{e}_{c2,t} = \frac{\partial[\hat{J}_t - r_t - \gamma \hat{J}_{t+1}]}{\partial \mathbf{e}_t} = \hat{\lambda}_t - \frac{\partial r_t}{\partial \mathbf{e}_t} - \gamma \frac{\partial \hat{\mathbf{e}}_{t+1}}{\partial \mathbf{e}_t}\hat{\lambda}_{t+1}, \tag{55}$$

where $e_{c1,t}$ denotes the TD error of the estimated cost-to-go $\hat{J}_t$ with current network weights, $\mathrm{e}_{c2,t}$ denotes the TD error of the computed derivatives $\hat{\lambda}_t$ with current network weights. It is noted that $\hat{J}_{t+1}$ and $\hat{\lambda}_{t+1}$ are predicted using the weights at time instant t.

The computation of items in Eq. (55) is worth analyzing. Considering Eq. (50), the second item on the right hand side of Eq. (55) can be computed by:

$$\frac{\partial r_t}{\partial \mathbf{e}_t} = (\mathbf{Q}^{\mathrm{T}} + \mathbf{Q})\mathbf{e}_t + \frac{\partial \mathbf{u}_t}{\partial \mathbf{e}_t} \cdot ((\mathbf{R}^{\mathrm{T}} + \mathbf{R})\mathbf{u}_t), \tag{56}$$

where $\partial \mathbf{u}_t / \partial \mathbf{e}_t$ goes through the actor network, which will be introduced in the next subsection. Then $\partial \hat{\mathbf{e}}_{t+1}/\partial \mathbf{e}_t$ requires to be handled with care since there exist two pathways for $\mathbf{e}_t$ to influence $\hat{\mathbf{e}}_{t+1}$. As shown in Fig. 1, one pathway goes through the incremental model directly (pathway 3.a), while another pathway firstly passes through the actor network and then goes through the incremental model (pathway 3.b):

$$\frac{\partial \hat{\mathbf{e}}_{t+1}}{\partial \mathbf{e}_t} = \underbrace{\frac{\partial \hat{\mathbf{e}}_{t+1}}{\partial \mathbf{e}_t}|_m}_{\text{pathway (3.}a\text{)}} + \underbrace{\frac{\partial \mathbf{u}_t}{\partial \mathbf{e}_t}|_a \cdot \frac{\partial \hat{\mathbf{e}}_{t+1}}{\partial \mathbf{u}_t}|_m}_{\text{pathway (3.}b\text{)}}, \tag{57}$$

Based on the Markov theory, the states at the time instant $t+1$ only have the relationship with the states at the time instant t. Therefore, according to Eq. (33), Eq. (57) can be simplified by the incremental model as:

$$\frac{\partial \hat{\mathbf{e}}_{t+1}}{\partial \mathbf{e}_t} = (\mathbf{I}_p + \hat{\underline{\mathcal{F}}}_{11,t}^{\mathrm{T}}) + \frac{\partial \mathbf{u}_t}{\partial \mathbf{e}_t} \cdot \hat{\underline{\mathcal{G}}}_{11,t}^{\mathrm{T}}, \tag{58}$$

Then, IGDHP combines both kinds of TD errors in an overall error function $E_{c,t}$:

$$E_{c,t} = \beta \frac{1}{2} e_{c1,t}^2 + (1-\beta)\frac{1}{2}\mathrm{e}_{c2,t}^{\mathrm{T}}\mathrm{e}_{c2,t}, \tag{59}$$

where β is a scalar within a range of $[0, 1]$. If $\beta = 1$, then it becomes pure IHDP. If $\beta = 0$, then the tuning of weights merely depends on the TD error of computed derivatives $\hat{\lambda}_t$, and consequently it is equivalent to IDHP.

Given a learning rate η_c, the critic weights $\mathbf{w}_{ci}$, where $i = 1, 2$, are updated with a gradient-descent algorithm to minimize the overall error $E_{c,t}$:

$$\mathbf{w}_{ci,t+1} = \mathbf{w}_{ci,t} - \eta_c \cdot \frac{\partial E_{c,t}}{\partial \mathbf{w}_{ci,t}}, \tag{60}$$

where

$$\begin{aligned}\frac{\partial E_{c,t}}{\partial \mathbf{w}_{ci,t}} &= \frac{\partial \hat{J}_t}{\partial \mathbf{w}_{ci,t}} \cdot \frac{\partial E_{c,t}}{\partial \hat{J}_t} + \frac{\partial \hat{\lambda}_t}{\partial \mathbf{w}_{ci,t}} \cdot \frac{\partial E_{c,t}}{\partial \hat{\lambda}_t} \\ &= \beta \frac{\partial \hat{J}_t}{\partial \mathbf{w}_{ci,t}} \cdot e_{c1,t} + (1-\beta) \frac{\partial \hat{\lambda}_t}{\partial \mathbf{w}_{ci,t}} \cdot \mathrm{e}_{c2,t},\end{aligned} \tag{61}$$

in which, $\partial \hat{\lambda}_t / \partial \mathbf{w}_{ci,t}$ represents the second-order mixed gradient of the estimated cost-to-go $\hat{J}_t$, and how to compute it is given without derivation [25] as follows:

If $i = 2$, then

$$\frac{\partial \hat{\lambda}_t}{\partial \mathbf{w}_{c2,t}} = \mathrm{diag}(\sigma'(\mathbf{w}_{c1,t}^{\mathrm{T}} \mathbf{e}_t))(\mathbf{I}_1 \otimes \mathbf{w}_{c1,t}^{\mathrm{T}}), \tag{62}$$

where $\otimes$ is the Kronecker product; and if $i = 1$, denote n_c as the number of neurons in the hidden layer, and then

$$\begin{aligned}\frac{\partial \hat{\lambda}_t}{\partial \mathbf{w}_{c1,t}} &= (\mathbf{w}_{c2,t} \odot \sigma'(\mathbf{w}_{c1,t}^{\mathrm{T}} \mathbf{e}_t)) \otimes \mathbf{I}_p - \mathbf{K}^{\mathrm{T}}(\mathbf{e}_t \otimes \mathbf{I}_{n_c}) \\ &\quad \times \mathrm{diag}(\sigma'(\mathbf{w}_{c1,t}^{\mathrm{T}} \mathbf{e}_t))\mathrm{diag}(\sigma(\mathbf{w}_{c1,t}^{\mathrm{T}} \mathbf{e}_t)) \\ &\quad \times \mathrm{diag}(\mathrm{vec}(\mathbf{w}_{c2,t}))(\mathbf{I}_1 \otimes \mathbf{w}_{c1,t}^{\mathrm{T}}),\end{aligned} \tag{63}$$

where $\mathbf{K}$ is a commutation matrix of $\mathbf{w}_{c1,t}$, and $\mathrm{vec}(\bullet)$ is a vector reshaping function. Note that the tensor operation can reduce the dimensionality of a matrix, and therefore dimensionality analysis is involved to reshape the results after computing $\partial \hat{\lambda}_t / \partial \mathbf{w}_{ci,t}$.

4.2. The actor network

Although in [35] a single critic network structure is utilized, since the IGDHP algorithm is actually implemented in a discrete way, an actor network is required for approximating the optimal control. Safety is vital for real physical systems and thus some restrictions are usually added to system control. In this paper, the output layer of the actor network employs a symmetrical sigmoid activation function, and is multiplied by an additive unchanged weight vector $\mathbf{u}_b = [u_{b1}, \ldots, u_{bm}]^{\mathrm{T}}$, where $u_{bi} \geq 0$, for $i = 1, \ldots, m$, so that the system control $\mathbf{u}_t = [u_{1,t}, \ldots, u_{m,t}]^{\mathrm{T}}$ outputted by the actor network is bounded by $\mathbf{u}_b$, i.e., $|u_{i,t}| < u_{bi}$ for $i = 1, \ldots, m$, as shown in Fig. 2. Consequently, the actor network is presented as:

$$\mathbf{u}_t = \mathbf{u}_b \odot \sigma(\mathbf{w}_{a2,t}^{\mathrm{T}} \sigma(\mathbf{w}_{a1,t}^{\mathrm{T}} [\mathbf{e}_t^{\mathrm{T}}, b_a]^{\mathrm{T}})), \tag{64}$$

where b_a is a constant bias term, which is introduced because the system control may not be a zero vector given zero tracking error, $\mathbf{w}_{a1,t}$ and $\mathbf{w}_{a2,t}$ are the weights of the actor network, and the way to define them is similar to that of $\mathbf{w}_{c1,t}$ and $\mathbf{w}_{c2,t}$.

The purpose of the actor network is to generate a near optimal control policy to minimize the future approximated cost-to-go $\hat{J}_{t+1}$:

$$\mathbf{u}_t^* = \arg\min_{\mathbf{u}_t} E_{a,t} = \arg\min_{\mathbf{u}_t} \frac{1}{2} \mathrm{e}_{a,t}^2, \tag{65}$$

where $E_{a,t}$ denotes the overall error and $\mathrm{e}_{a,t}$ stands for the error between the approximated future cost-to-go $\hat{J}_{t+1}$ and the target 0 cost-to-go, i.e., $\mathrm{e}_{a,t} = \hat{J}_{t+1}$.

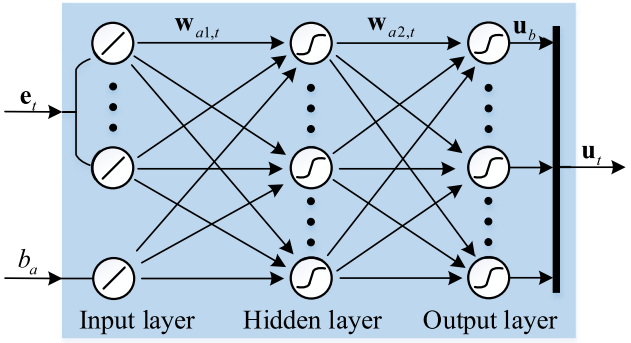

Fig. 2. The structure of the actor network, where the input layer employs a unit-proportion linear activation function while the hidden and output layer exploit a symmetrical sigmoid activation function.

The system control $\mathbf{u}_t$ is also an input of the incremental model, so even though $\mathbf{u}_t$ does not appear in the reward function, i.e., R is zero, it has an influence on the critic output at the next time instant. Therefore, a gradient-descent algorithm can be implemented to iteratively solve Eq. (65) by the 4th pathway starting from $\hat{J}_{t+1}$ through $\hat{\mathbf{e}}_{t+1}$ to $\mathbf{u}_t$. Different from the straightforward form, whose back-propagation pathways of the actor network can start from either $\hat{J}_{t+1}$ or $\hat{\lambda}_{t+1}$, there is only one back-propagation pathway for IGDHP with explicit analytical calculations to update the actor weights. Nevertheless, this explains exactly why the structure with explicit analytical calculations surpasses the straightforward structure, in that it releases the restriction of scalar β of being 0 or 1. Specifically, for the straightforward form, if $\beta = 0$, then the elements in $\mathbf{w}_{c2}$ linked to $\hat{J}_{t+1}$ will never be updated, and if the actor network is trained through the pathway leading from $\hat{J}_{t+1}$, the back-propagation cannot be carried out. Similarly, if the back-propagation channel of the actor network starts from $\hat{\lambda}_{t+1}$, then $\beta \neq 1$ is mandatory for the straightforward form. On the contrary, the structure with explicit analytical calculations has no such limitations on β, because even though $\beta = 0$, the critic network can still be trained.

As presented in Fig. 1, the actor weights are updated along the 4th back-propagation pathway with a learning rate η_a:

$$\mathbf{w}_{ai,t+1} = \mathbf{w}_{ai,t} - \eta_a \cdot \frac{\partial E_{a,t}}{\partial \mathbf{w}_{ai,t}}, \tag{66}$$

where $i = 1, 2$, and

$$\frac{\partial E_{a,t}}{\partial \mathbf{w}_{ai,t}} = \frac{\partial \mathbf{u}_t}{\partial \mathbf{w}_{ai,t}} \cdot \frac{\partial \hat{\mathbf{e}}_{t+1}}{\partial \mathbf{u}_t} \cdot \frac{\partial \hat{J}_{t+1}}{\partial \hat{\mathbf{e}}_{t+1}} \cdot \frac{\partial E_{a,t}}{\partial \hat{J}_{t+1}} = \frac{\partial \mathbf{u}_t}{\partial \mathbf{w}_{ai,t}} \cdot \hat{\mathcal{G}}_{11,t}^{\mathrm{T}} \cdot \hat{\lambda}_{t+1} \cdot \hat{J}_{t+1}. \tag{67}$$

So far the implementation of the proposed IGDHP with PO control scheme has been introduced. The procedure is briefly summarized in the following Algorithm 1, where line 6 is the dividing line between the current time instant and the next time step.

Remark 2. The convergence analysis of the online identification has been presented in Section 3.3, and the convergence analysis of the ADP scheme has been investigated in [29,30]. However, as stated in [25], it is currently unfeasible to theoretically prove the closed-loop stability of the IGDHP algorithm due to its completely online implementation. Accordingly, repeating numerical experiments are carried out in the next section for verification.

Algorithm 1: Design procedure of the IGDHP with PO control scheme.

Initialization: initialize system states, and parameters of the identifier, the critic network and the actor network;
while *terminal condition is not triggered* **do**
  compute the control input $\mathbf{u}_t$ using the actor network (Eq. (64)) with the current observation;
  predict the one-step observation $\hat{\mathbf{e}}_{t+1}$ using the identifier (Eq. (32)) with the stored data;
  evaluate the current control policy using the critic network (Eqs. (51) and (52));
  sample the real one-step observation $\mathbf{e}_{t+1}$ by applying $\mathbf{u}_t$ to the real plant;
  update the weights of the actor and critic networks via the backpropagation technique (Eqs. (60) and (66));
  if *sufficient samples are stored in the sliding window* **then**
    update the identifier using the RLS algorithm (Eqs. (36)–(38));
  else
    continue;
  end
  update the stored data in the sliding window;
end

5. Numerical experiments

This section assesses the developed IGDHP algorithm on a practical aerospace application. Firstly, traditional GDHP with PO (GDHP-PO), IGDHP with FSF (IGDHP-FSF) and IGDHP with PO (IGDHP-PO) are compared by applying them on an attitude tracking control task. Then the IGDHP-PO algorithm is adopted for an altitude control problem in combination with a hierarchy technique and PID controller.

5.1. Aerospace system model

A nonlinear model of aircraft is set up utilizing the public data [49] and only the longitudinal dynamics are taken into consideration. All parameters without special explanation in this paper are determined by trimming the aerodynamic model in a steady wings-level flight condition at 15000 ft and 600 ft/s, which will be referred to as the benchmark condition.

In this longitudinal control problem, there are 6 states, namely altitude h, airspeed v, pitch angle θ, angle of attack (AOA) α, flight path angle γ_F and pitch rate q. Nevertheless, for a specific task, it is feasible to select only some main states to reduce computation. For instance, for the AOA tracking task, which is an attitude control problem, only pitch rate q, the basic state in the rate loop, is chosen as the additional feature state in [25], while other states are considered as parts of the black box to be identified.

For the longitudinal aerodynamic model, there are 3 control inputs, namely leading edge flap deflection δ_{lef}, engine thrust T and elevator deflection δ_e. The control surface of leading edge flap cannot be directly changed by the pilot [49], and therefore is regarded as inner unknown dynamics. Out of simplicity for implementation and convenience for analysis, a simple PID thrust control to maintain the airspeed is designed in a separate control loop [39], such that only one control input, elevator deflection δ_e, is taken into concern and a mapping between one control input and one final output can be constructed. Before the elevator deflection is practically adjusted, the system control $\mathbf{u}_t$ generated by the actor, or called elevator deflection command δ_e^c in this aerospace application, has to go through the actuator,

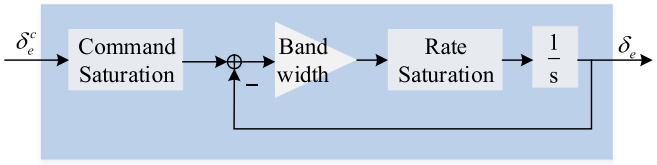

Fig. 3. The dynamics of actuator, a first-order system with rate and position limits [25].

Table 1
Magnitude of the noises having impacts on controller design.
Source: Adapted from [50].

	h [m]	θ, α, γ_F [deg]	q [deg/s]	δ_e [deg]
Mean	1	2.2×10^{-1}	1.7×10^{-3}	2.6×10^{-1}
Standard deviation	8×10^{-2}	1.8×10^{-3}	3×10^{-2}	4×10^{-2}

which is modeled as a first-order system with rate and position constraints, as illustrated in Fig. 3 [25]. The bandwidth of actuator is 20.2 rad/s, deflection cannot exceed the range of [−25 deg, 25 deg] and its changing rate is bounded in the range of [−90 deg/s, 90 deg/s] [39,49].

In real-word applications, noise is unavoidable, and unforeseen changes of system dynamics or even sudden failures might be encountered. Consequently, there is need to approximately model these uncertainties for verification of the developed method. Non-zero mean white measurement noise acting on the feedback signals and actuator is taken into account, and the magnitude of real-world phenomena utilized for simulation is given in Table 1 [50]. This noise can have impacts on the performance of the controller, but can also act as exploration noise to better satisfy the persistent excitation (PE) condition [25]. It is noted that only PO methods requires to consider measurement noise, while in the FSF condition, the acquired data is assumed noise-free.

Sudden structural damages may be encountered during flight, which require fault-tolerant control (FTC) [14]. Fault diagnosis is a significant part of FTC, but will not be discussed in this paper. Several kinds of sudden faults and their aftermath have been scrutinized in [39]. It has been investigated that moving mechanisms, such as actuators, are more likely to be affected by unexpected faults. There are four faults taken into consideration, namely reduction of bandwidth, strengthening of rate limitation, output deviation and reduction of control effectiveness, while the last situation can also be triggered by the structural damages changing aerodynamics [25]. As to the faults of static structure, damage of the horizontal stabilizer is selected to be investigated. This damage has significant impact on both static and dynamic stability for longitudinal dynamics, and mass loss is often encountered simultaneously, which will instantaneously change the center of gravity.

5.2. Simulation results

An AOA tracking problem is firstly taken into consideration. How to implement the GDHP method can be found in [25] and is omitted here. For better comparison, the settings in IGDHP-FSF, IGDHP-PO and GDHP-PO are as consistent as possible and the best possible parameters are ascertained by repeating the experiment. The actor, critic and model networks are all fully connected and consist of three layers of nodes. For balancing the accuracy with the computational efficiency, we experimentally set the number of hidden layer neurons in all three networks as 25 and a sigmoid function is adopted as the activation function. Both actor network and model network introduce a bias term

Table 2
Parameters about the descending learning rates.

	η_a	η_{c1}	η_{c2}	η_m
Initial value	2	1	0.2	1
Descending coefficient	0.999995	0.9995	0.995	0.999995
Lower bound	10^{-2}	10^{-4}	10^{-4}	10^{-2}

as an input, and $b_a = b_m = 0.01$. Large random initial weights can significantly affect learning performance, whereas small ones will decrease initial exploration efficiency. Therefore, as a trade off, All weights are randomly initialized within a small range of $[-0.1, 0.1]$, and bounded within the range of $[-20, 20]$. With the information of derivatives, the DHP technique generally surpasses HDP in tracking precision, convergence speed and success rate [6] and therefore β is chosen to be 0.1 to take advantage of the derivative information. We experimentally choose $\mathbf{Q} = 8$, $\mathbf{R} = 1$ and $\gamma = 0.99995$ to formulate the critic network. It should be noted that $\mathbf{R}$ can also be 0, but for the purpose of decreasing energy cost, it is set positive definite. As mentioned above, the system state only includes AOA and pitch angle, and therefore the minimum width of the sliding window is 3. To enhance the stability, the window width is set to 4 for IGDHP-PO and IGDHP-FSF, and 5 sets of data are utilized by GDHP-PO to ensure the same level of data use, in that both IGDHP-PO and IGDHP-FSF employ the increment information, which is acquired from 5 time instants. $\hat{\underline{\mathcal{F}}}_t$ is initialized to be composed by 4 identity matrices and $\hat{\underline{\mathcal{G}}}_t$ starts from a zero matrix. γ_{RLS} is chosen to be 0.99995. $\underline{\mathrm{Cov}}_t$ originally is a diagonal matrix with all main diagonal elements chosen to be 10^7 [48], since the trace of $\underline{\mathrm{Cov}}_t$ indicates the magnitude of the estimated errors, which can initially be infinite due to insufficient or incorrect knowledge of the system [51]. Besides, a descending method is applied to guarantee effective learning, which suggests that the learning rates are initially set to be large numbers and gradually decrease at each time step by being multiplied by a descending coefficient until bound is reached, and the corresponding parameters are given in Table 2. All experiments are carried out with a sampling frequency of 1kHz and computed using Euler's method. In order to achieve the PE condition, a probing signal is introduced at the initial exploration stage. How to design a good probing signal is still an open problem, but for this paper a 3211 disturbance signal that changes its sign over time as a particular proportion [5,25] is introduced to excite the system modes.

Firstly, the system is supposed to track a human-designed AOA reference signal online using IGDHP-FSF, IGDHP-PO and GDHP-PO, respectively. The AOA reference signal varies around the trimmed condition, namely 2.6638 deg. The comparison of the performance is given in Fig. 4, where the system is initialized to be at the benchmark condition. If successfully implemented, all three methods can complete the tracking task. Among them, IGDHP-FSF shows the best performance, with a control policy that converges fast. In the PO condition, both IGDHP-PO and GDHP-PO methods can track the reference after a small period of vibration and get similar performances. The settling time of IGDHP-PO is slightly smaller than that of GDHP-PO method but it can be ignored in the practical applications. Due to the influence of measurement noises, the tracking precision of these PO-based methods is lower than IGDHP-FSF. As presented in subfigure (b), the tracking errors (denoted by $\Delta\alpha$) of IGDHP-PO and GDHP-PO keep oscillating around the mean value of the AOA sensor noise, while the tracking error of IGDHP-FSF is approximately 0 at the stable stage.

Fig. 5 illustrates the elevator deflection command produced by the IGDHP-PO method starting from the benchmark condition. From the inset, it can be seen that a 3211 disturbance signal,

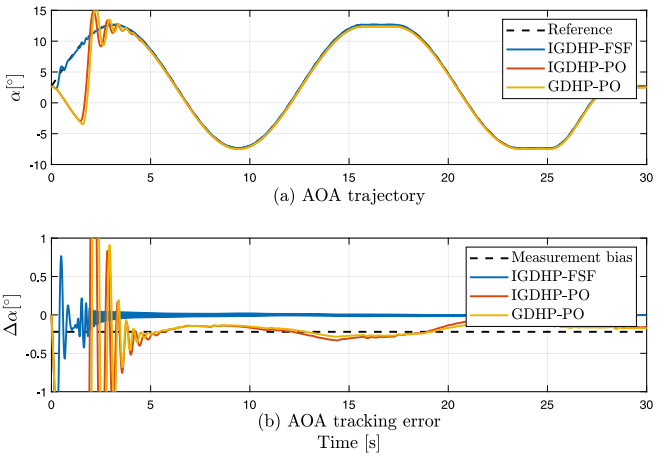

Fig. 4. Online AOA tracking control with initially benchmark condition.

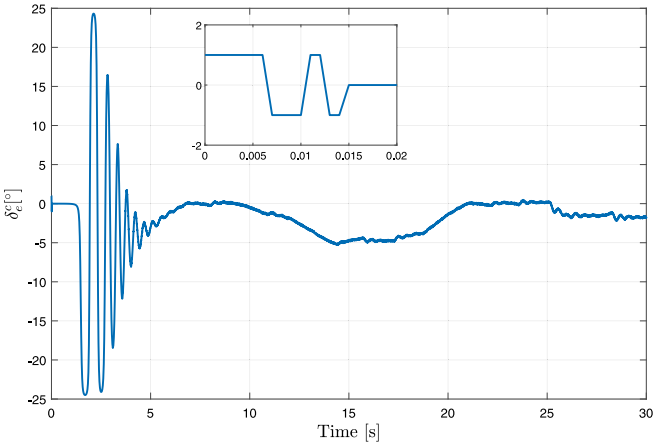

Fig. 5. Elevator deflection command produced by the IGDHP-PO method starting from the benchmark condition.

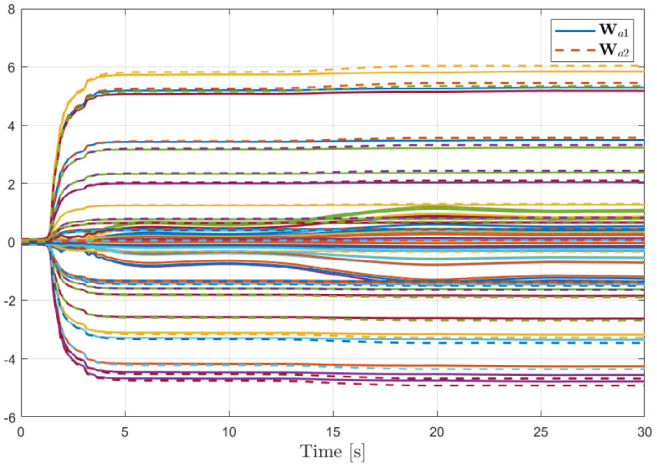

Fig. 6. Convergence of the actor weights using the IGDHP-PO approach with initially benchmark condition.

is applied to kick off the learning process. The control input command turns to be nearly 0 after the probing signal vanishes since the initial weights of the actor network are tiny random values, as presented in Fig. 6. Then both critic and actor networks are excited and the actor–critic scheme behaves as a high-gain

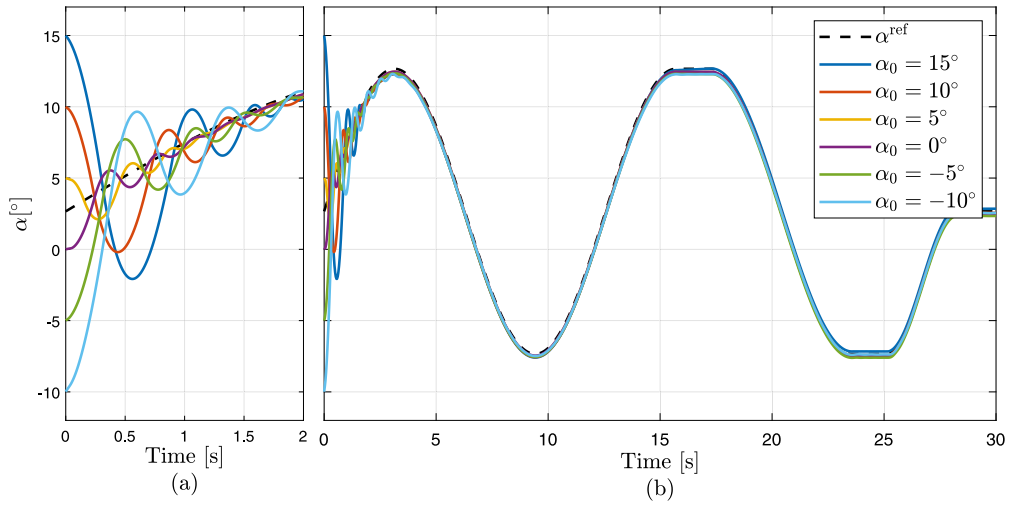

Fig. 7. Online AOA tracking control with different initial states using the IGDHP-PO approach.

controller during the exploration stage until the weights converge. The learning process is performed totally online and will continue even after the policy has converged.

In addition, the initial condition can have an impact on the controller while the proposed IGDHP-PO approach can deal with a wide range of initial states within [−10 deg, 15 deg] without loss of precision. As presented in Fig. 7, the AOA can track the given reference signal α^{ref} in less than 2 s for all initial conditions using the IGDHP-PO approach, which is indicative of its competent adaptability and robustness.

Nevertheless, only when the task is successfully carried out, can the results presented above make sense. Random factors, such as initial weights of ANNs and measurement noises, can have impact on the performance and occasionally even trigger divergence and failure. To compare the robustness of these algorithms to various aspects, a concept of success ratio is introduced as a performance index. For a successful implementation, the amplitude of the AOA trajectory cannot transcend a range of [−15 deg, 20 deg] over the whole control period, the system is supposed to be able to track the reference signal after 2π seconds, and the tracking errors will not exceed 1 deg hereafter. Remaining all parameters unchanged except for probing signal, 1000 Monte Carlo simulations are implemented to assess the performance of these approaches.

The experiment are executed with 7 different initial AOAs and equal reference to evaluate the robustness of these approaches towards initial tracking errors and the results regarding success ratio are illustrated in Table 3. Both IGDHP-FSF and IGDHP-PO have a success ratio of 100% in the benchmark condition, which implies these approaches are stable for this tracking control problem. On the other hand, the success ratio of GDHP-PO in the benchmark condition is merely 51.2% and for all initial states, the success ratios of IGDHP-FSF and IGDHP-PO outclass those of GDHP-PO, demonstrating the incremental technique can significantly improve system identification compared to the traditional global ANN-based model in this online application. The success ratios of IGDHP-PO are generally smaller than those of IGDHP-FSF, which is intuitively predictable, while the differences are less than 1% for more than half initial states and this shows that the IGDHP-PO can cope with PO situations. It is shown that the success ratios with $\alpha_0 = -5$ deg and $\alpha_0 = 0$ deg decrease by 35.5% and 21.5% for IGDHP-FSF and by 53.7% and 26.5% for IGDHP-PO compared to the benchmark condition, respectively. Besides the algorithms themselves, this reduction of success ratio can also be caused by the collective effect of the nominal

Table 3
Success ratio comparison for different initial states with 1000 times of Monte Carlo simulation.

α_0[deg]	−10	−5	0	2.6638[a]	5	10	15
IGDHP-FSF	100%	64.5%	78.5%	100%	100%	99.1%	99.4%
IGDHP-PO	92.7%	46.3%	73.5%	100%	99.3%	99.2%	99.6%
GDHP-PO	25.7%	38.2%	45.3%	51.2%	44.3%	41.4%	50.5%

[a]AOA value in the benchmark condition.

reference signal and inherent dynamics of the system. Besides, it should also be noted that in multiple cases the success ratio is not 100%, which is owing to the fact that it is arduous to accomplish optimal PE condition due to the circular argument between PE condition, accurate system information and stable control policy [25]. Although some cases can achieve a success ratio of 100%, random factors prevent the controller from consistent perfect tracking. Nevertheless, there is still prospect of full success and the results presented in Table 3 are obtained based on current settings. The development of various aspects can benefit the stability and improve success ratio, such as sensor precision, probing signal, parameter initialization and learning rates. It is still an open problem to improve these factors and therefore this paper concentrates on the assessment of robustness between IGHDP-FSF, IGDHP-PO and GDHP-PO.

The adaptability is one of the most significant aspects through which ACD approaches show their superiority over other conventional optimal control methods. Sound polices can be learned automatically by updating the weights of the networks, which enables ACDs be applied to FTC problems. Therefore, the following part investigates and compares the capability to adapt of IGDHP-FSF, IGDHP-PO and GDHP-PO by applying them to two fault scenarios, where the system is supposed to track a sinusoidal reference signal. Considering the moment when the faults happen can have an influence to the results, the faults are designed to take place at the instants of 4π seconds and 5π seconds, respectively.

The first fault scenario examined is that the elevator actuator is partially damaged. Specifically, the elevator suddenly loses 30% of its effectiveness and 50% of its bandwidth, and the bound of deflection rate degrades by 1/3 to [−60 deg/s, 60 deg/s]. Besides, there is a constant disturbance acting on the actuator, making the real deflection 2 deg larger than the produced commands. The results are illustrated in Figs. 8 and 9, where the malfunction of the actuator does not exceedingly affect the performance of

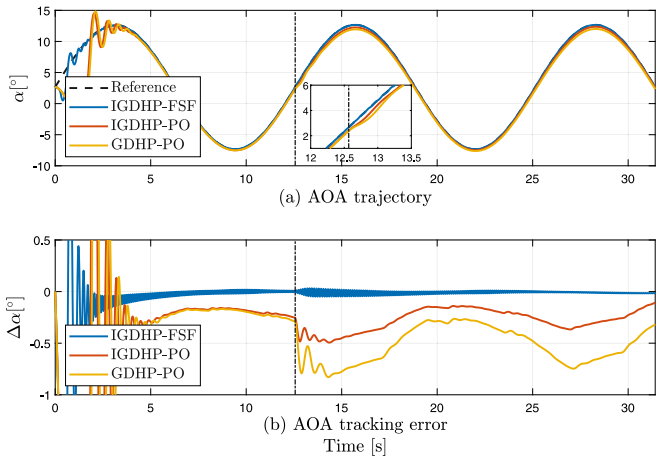

Fig. 8. Online AOA tracking control with a malfunction of the actuator occurring at the 4π seconds.

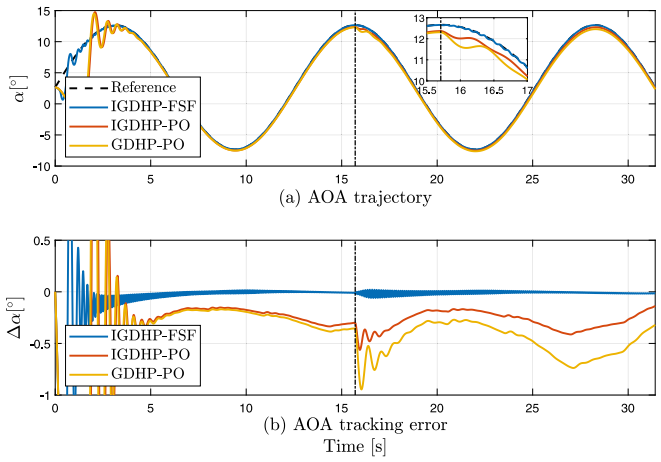

Fig. 9. Online AOA tracking control with a malfunction of the actuator occurring at the 5π seconds.

these adaptive controllers, especially for IGDHP-FSF, which acts completely to normal after a small oscillation, without significant increase of tracking errors. The impacts of the sudden damage on IGDHP-PO and GDHP-PO are more obvious, which escalate their tracking errors in varying degrees. It can be seen that the tracking errors of both approaches increase after encountering instantaneous damage, while IGDHP-PO relatively surpasses GDHP-PO since the latter has larger errors. Nevertheless, all approaches are capable of handling this fault scenario.

The second fault scenario investigated is the damage of the left horizontal stabilizer. With an intact right horizontal stabilizer, the center of gravity inevitably shifts from the normal position, which will lead to an increment of pitching moment. This structural damage can also affect the closed-loop performance by degrading longitudinal damping and stability margin. As presented in Figs. 10 and 11, the static structural fault caused by damage of the horizontal stabilizer demonstrates a bigger impact compared to the considered actuator fault. At large positive values of AOA, the tracking performance of all three approaches significantly degrades to different extents. On the whole, all approaches can adapt to this sudden fault in that their performance is improving with the time. IGDHP-PO still outperforms GDHP-PO in adaptation since after one period of reference signal, the tracking errors of IGDHP-PO has declined to less than 1.5 deg while tracking errors of GDHP are beyond 2 deg. Synthesizing two fault scenarios,

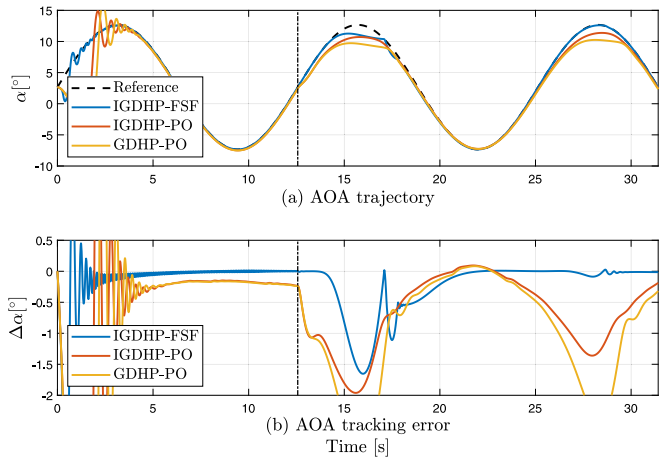

Fig. 10. Online AOA tracking control with the left horizontal stabilizer damaged at the 4π seconds.

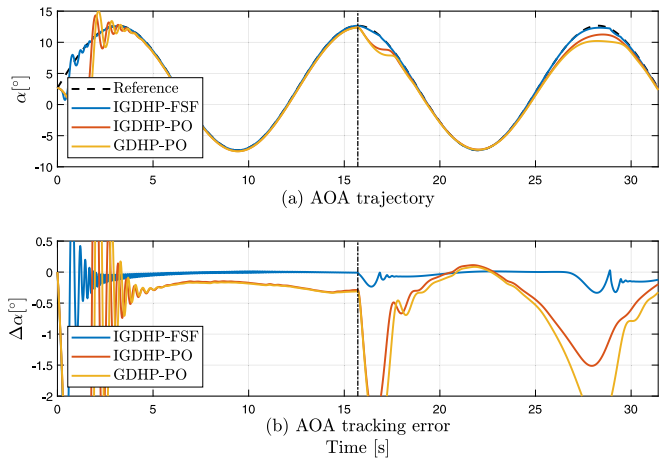

Fig. 11. Online AOA tracking control with the left horizontal stabilizer damaged at the 5π seconds.

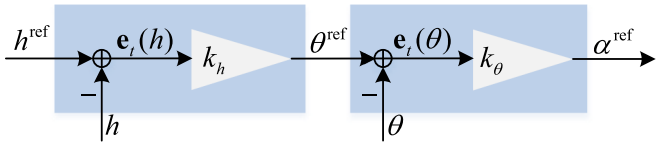

Fig. 12. Altitude and pitch angle control loops used to generate AOA reference signal.

it can be concluded that incremental technique has an advantage of quick adaptation over conventional ANN-based global model.

Then, more practical application scenarios are investigated, where the altitude control loop assisted with PID controller is introduced to generate a more realistic reference signal. The pitch angle of the system is selected as the controlled variable of the intermediate loop, i.e., from outer loop to inner loop, the controlled variables are altitude, pitch angle and AOA, respectively, as illustrated in Fig. 12. Other system states are regarded as the unmeasured inner states. Although flight path angle is more widely used as the intermediate variable, in this experiment, choosing pitch angle show a better performance, which is also feasible in the real world. In practice, proportional control is often applied alone, and in this paper, $k_h = 3$, $k_\theta = 5$, while all other parameters are kept unchanged.

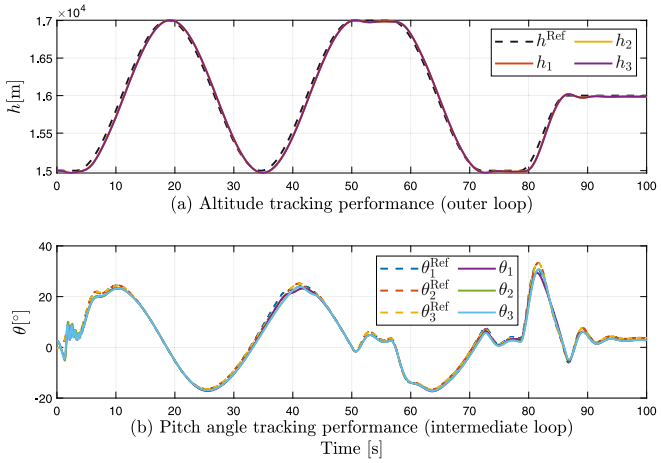

Fig. 13. Altitude and pitch angle control loops used to generate AOA reference signal.

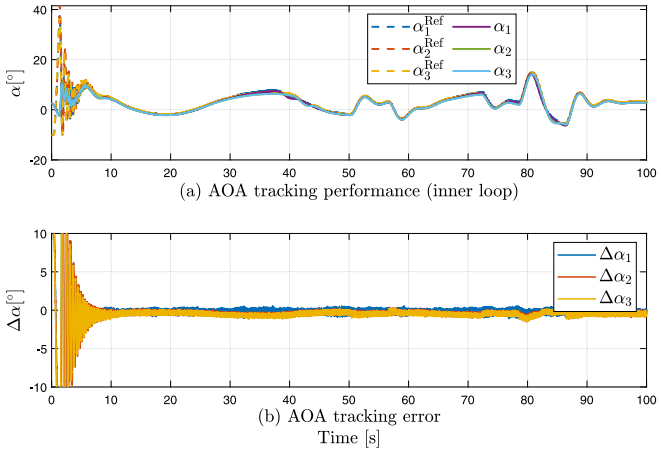

Fig. 14. Online AOA tracking control with the reference provided by altitude tracking control task.

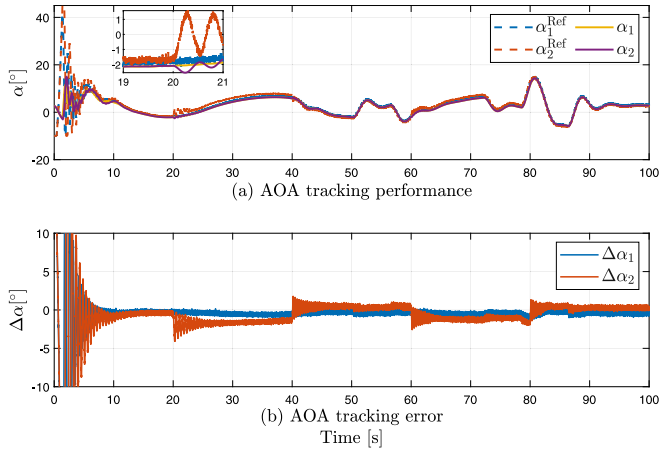

Fig. 15. Online AOA tracking control with the presence of sudden load disturbance.

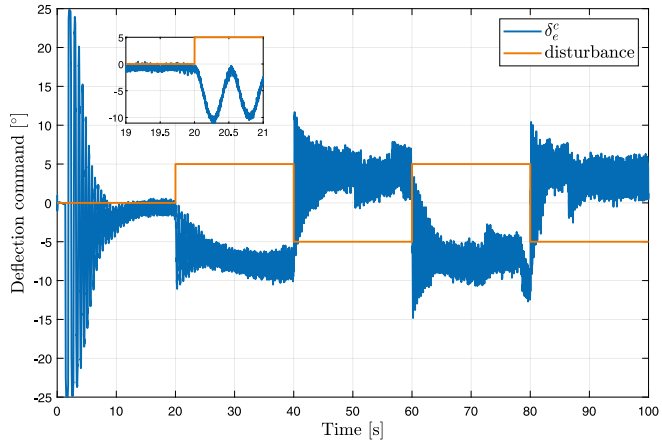

Fig. 16. Evolution of the elevator deflection command and the sudden load disturbance.

Parameter variations can affect the dynamic response of control system. Hence, different discounting factors are examined to further verify the robustness of the developed IGDHP-PO algorithm. Figs. 13 and 14 demonstrate the tracking performance with different discounting factors, where the subscript 1, 2, and 3 respectively stands for the case of $\gamma = 0.99995$, $\gamma = 0.9$, and $\gamma = 0.85$. As can be seen, if successfully implemented, comparable performance can be obtained with different discounting factors. Because of initially random policy and totally online learning, the tracking performance at the beginning is also imperfect. Due to the measurement uncertainties and proportional controllers, the generated AOA reference oscillates over the altitude control task. Despite this, the developed approach manages to keep the tracking errors mostly within 1 deg and controls the system to track the designed altitude reference. Nevertheless, it is also observed that with other parameters unchanged, different discounting factors can lead to different success ratios, specifically 99.1% for $\gamma = 0.99995$, 97.3% for $\gamma = 0.9$, and 70.7% for $\gamma = 0.85$. This shows that the developed IGDHP-PO approach is robust to the forgetting factor to a certain extent.

In addition, load disturbance is often encountered in flight control due to turbulence. Therefore, the tracking performance with the presence of sudden load disturbance is examined. Although the turbulence can act on the whole aircraft, out of the purpose of simplicity and reproducibility, the considered load disturbance is set as an equivalent square wave disturbance acting on the real elevator deflection, i.e. δ_e. As presented in Figs. 15 and 16, although sudden disturbance has an impact on tracking performance, the proposed IGHDP-PO approach can adapt with a fast changing deflection command. The largest impact happens at the instant when the disturbance load appears for the first time. When an equivalent 5 deg deflection disturbance is encountered, the controller tries to generate an opposite action to stabilize it. Due to the overshoot, oscillations are initiated, but the AOA signal manages to track the reference. After the onset, the subsequent changing disturbance becomes less influential, which demonstrates the robustness and the online learning property of the proposed method.

6. Conclusion

This paper develops an incremental model-based global dual heuristic programming (IGDHP) approach by combining global dual heuristic programming (GDHP) and augmented incremental techniques, which solves the partial observability problem. Moreover, the input saturation constraint is overcome by utilizing a symmetrical sigmoid function as the output layer activation function of the actor network, which frees the matrix **R** in the reward from having to be positive definite.

Various numerical simulation studies are conducted with an aerospace system, including attitude control problems with different initial states and sudden structural changes, and an altitude control problem combined with PID controller and hierarchy technique. The results uniformly demonstrate that the developed IGDHP algorithm can effectively deal with partial observability and surpass conventional GDHP in online stability, robustness to different initial states, and adaptability when encountering unforeseen faults. The applications to altitude control demonstrate that the developed IGDHP algorithm is robust to parameter variations and load disturbance, and has the potential to be applied to realistic complex scenarios combined with other techniques.

Future research should continue working on bridging the gap between the algorithms and real world systems, and approaches and skills to better satisfy the persistence excitation (PE) condition are specially recommended.

CRediT authorship contribution statement

Bo Sun: Conceptualization, Methodology, Software, Validation, Investigation, Writing - original draft, Visualization, Funding acquisition. **Erik-Jan van Kampen:** Investigation, Resources, Writing - review & editing, Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Suresh, N. Kannan, Direct adaptive neural flight control system for an unstable unmanned aircraft, Appl. Soft Comput. 8 (2008) 937–948.

[2] D.P. Bertsekas, Dynamic Programming and Optimal Control, Vol. 1, Athena scientific Belmont, MA, 1995.

[3] H.J. Kappen, Optimal Control Theory and the Linear Bellman Equation, Cambridge University Press, 2011.

[4] W.B. Powell, Approximate Dynamic Programming: Solving the Curses of Dimensionality, Vol. 703, John Wiley & Sons, 2007.

[5] Y. Zhou, E.-J. van Kampen, Q.P. Chu, Nonlinear adaptive flight control using incremental approximate dynamic programming and output feedback, J. Guid. Control Dyn. 40 (2016) 493–496.

[6] Y. Zhou, E.-J. van Kampen, Q.P. Chu, Incremental approximate dynamic programming for nonlinear adaptive tracking control with partial observability, J. Guid. Control Dyn. 41 (2018) 2554–2567.

[7] B. Sun, E.-J. van Kampen, Incremental model-based heuristic dynamic programming with output feedback applied to aerospace system identification and control, in: 2020 IEEE Conference on Control Technology and Applications (CCTA), IEEE, 2020, pp. 366–371.

[8] P. Hušek, K. Narenathreyas, Aircraft longitudinal motion control based on takagi–sugeno fuzzy model, Appl. Soft Comput. 49 (2016) 269–278.

[9] S.A.A. Rizvi, Z. Lin, Output feedback q-learning control for the discrete-time linear quadratic regulator problem, IEEE Trans. neural Netw. Learn. Syst. 30 (2018) 1523–1536.

[10] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT press, 2018.

[11] R.S. Sutton, A.G. Barto, R.J. Williams, Reinforcement learning is direct adaptive optimal control, IEEE Control Syst. Mag. 12 (1992) 19–22.

[12] F.L. Lewis, D. Vrabie, Reinforcement learning and adaptive dynamic programming for feedback control, IEEE Circuits Syst. Mag. 9 (2009) 32–50.

[13] D. Wang, H. He, D. Liu, Adaptive critic nonlinear robust control: A survey, IEEE Trans. Cybern. 47 (2017) 3429–3451.

[14] X. Liu, B. Zhao, D. Liu, Fault tolerant tracking control for nonlinear systems with actuator failures through particle swarm optimization-based adaptive dynamic programming, Appl. Soft Comput. 97 (2020) 106766.

[15] F.L. Lewis, K.G. Vamvoudakis, Reinforcement learning for partially observable dynamic processes: Adaptive dynamic programming using measured output data, IEEE Trans. Syst. Man Cybern. B 41 (2010) 14–25.

[16] B. Kiumarsi, F.L. Lewis, M.-B. Naghibi-Sistani, A. Karimpour, Optimal tracking control of unknown discrete-time linear systems using input–output measured data, IEEE Trans. Cybern. 45 (2015) 2770–2779.

[17] A. Keshavarz, S. Boyd, Quadratic approximate dynamic programming for input-affine systems, Internat. J. Robust Nonlinear Control 24 (2014) 432–449.

[18] D. Liu, X. Yang, D. Wang, Q. Wei, Reinforcement-learning-based robust controller design for continuous-time uncertain nonlinear systems subject to input constraints, IEEE Trans. Cybern. 45 (2015) 1372–1385.

[19] H. Modares, F.L. Lewis, Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning, Automatica 50 (2014) 1780–1792.

[20] H. Modares, F.L. Lewis, M.-B. Naghibi-Sistani, Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems, Automatica 50 (2014) 193–202.

[21] F.A. Yaghmaie, D.J. Braun, Reinforcement learning for a class of continuous-time input constrained optimal control problems, Automatica 99 (2019) 221–227.

[22] K. Zhang, H. Zhang, X. Liang, Z. Wang, Neurodynamic programming and tracking control scheme of constrained-input systems via a novel event-triggered pi algorithm, Appl. Soft Comput. 83 (2019) 105629.

[23] Y. Zhou, E.-J. van Kampen, Q.P. Chu, Incremental model based online dual heuristic programming for nonlinear adaptive control, Control Eng. Pract. 73 (2018) 13–25.

[24] B. Sun, E.-J. van Kampen, Incremental model-based global dual heuristic programming for flight control, IFAC-PapersOnLine 52 (2019) 7–12.

[25] B. Sun, E.-J. van Kampen, Incremental model-based global dual heuristic programming with explicit analytical calculations applied to flight control, Eng. Appl. Artif. Intell. 89 (2020) 103425.

[26] B. Kiumarsi, K.G. Vamvoudakis, H. Modares, F.L. Lewis, Optimal and autonomous control using reinforcement learning: A survey, IEEE Trans. Neural Netw. Learn. Syst. 29 (2017) 2042–2062.

[27] B. Depraetere, M. Liu, G. Pinte, I. Grondman, R. Babuška, Comparison of model-free and model-based methods for time optimal hit control of a badminton robot, Mechatronics 24 (2014) 1021–1030.

[28] D.V. Prokhorov, D.C. Wunsch, Adaptive critic designs, IEEE Trans. Neural Netw. 8 (1997) 997–1007.

[29] B. Sun, E.-J. van Kampen, Launch vehicle discrete-time optimal tracking control using global dual heuristic programming, in: 2020 IEEE Conference on Control Technology and Applications (CCTA), IEEE, 2020, pp. 162–167.

[30] D. Liu, D. Wang, D. Zhao, Q. Wei, N. Jin, Neural-network-based optimal control for a class of unknown discrete-time nonlinear systems using globalized dual heuristic programming, IEEE Trans. Autom. Sci. Eng. 9 (2012) 628–634.

[31] D. Wang, D. Liu, Q. Wei, D. Zhao, N. Jin, Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming, Automatica 48 (2012) 1825–1832.

[32] E.-J. van Kampen, Q.P. Chu, J. Mulder, Continuous adaptive critic flight control aided with approximated plant dynamics, in: AIAA Guidance, Navigation, and Control Conference and Exhibit, 2006, p. 6429.

[33] D. Vrabie, F. Lewis, Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems, Neural Netw. 22 (2009) 237–246.

[34] S. Bhasin, R. Kamalapurkar, M. Johnson, K.G. Vamvoudakis, F.L. Lewis, W.E. Dixon, A novel actor–critic–identifier architecture for approximate optimal control of uncertain nonlinear systems, Automatica 49 (2013) 82–92.

[35] J. Na, G. Herrmann, Online adaptive approximate optimal tracking control with simplified dual approximation structure for continuous-time unknown nonlinear systems, IEEE/CAA J. Autom. Sin. 1 (2014) 412–422.

[36] D. Liu, Y. Huang, D. Wang, Q. Wei, Neural-network-observer-based optimal control for unknown nonlinear systems using adaptive dynamic programming, Internat. J. Control 86 (2013) 1554–1566.

[37] I. Grondman, M. Vaandrager, L. Busoniu, R. Babuska, E. Schuitema, Efficient model learning methods for actor–critic control, IEEE Trans. Syst. Man Cybern. B 42 (2012) 591–602.

[38] Y. Huang, D.M. Pool, O. Stroosma, Q. Chu, Long-stroke hydraulic robot motion control with incremental nonlinear dynamic inversion, IEEE/ASME Trans. Mechatronics 24 (2019) 304–314.

[39] X. Wang, E.-J. van Kampen, Q.P. Chu, P. Lu, Incremental sliding-mode fault-tolerant flight control, J. Guid. Control Dyn. 42 (2018) 244–259.

[40] X. Wang, S. Sun, E.-J. van Kampen, Q.P. Chu, Quadrotor fault tolerant incremental sliding mode control driven by sliding mode disturbance observers, Aerosp. Sci. Technol. 87 (2019) 417–430.

[41] Y. Zhou, E.-J. Van Kampen, Q. Chu, Incremental model based online heuristic dynamic programming for nonlinear adaptive tracking control with partial observability, Aerosp. Sci. Technol. 105 (2020) 106013.

[42] N. Szanto, V. Narayanan, S. Jagannathan, Event-sampled direct adaptive NN output- and state-feedback control of uncertain strict-feedback system, IEEE Trans. Neural Netw. Learn. Syst. 29 (2017) 1850–1863.

[43] Z. Wang, R. Lu, F. Gao, D. Liu, An indirect data-driven method for trajectory tracking control of a class of nonlinear discrete-time systems, IEEE Trans. Ind. Electron. 64 (2017) 4121–4129.

[44] S. Ragi, E.K. Chong, UAV path planning in a dynamic environment via partially observable Markov decision process, IEEE Trans. Aerosp. Electron. Syst. 49 (2013) 2397–2412.
[45] Y. Zhou, E.-J. van Kampen, Q.P. Chu, Hybrid hierarchical reinforcement learning for online guidance and navigation with partial observability, Neurocomputing 331 (2019) 443–457.
[46] X. Wang, E.-J. van Kampen, Q.P. Chu, P. Lu, Stability analysis for incremental nonlinear dynamic inversion control, J. Guid. Control Dyn. 42 (2019) 1116–1129.
[47] F. Grondman, G. Looye, R.O. Kuchar, Q.P. Chu, E.-J. van Kampen, Design and flight testing of incremental nonlinear dynamic inversion-based control laws for a passenger aircraft, in: 2018 AIAA Guidance, Navigation, and Control Conference, 2018, p. 0385.
[48] B. Farhang-Boroujeny, Adaptive Filters: Theory and Applications, second ed., John Wiley & Sons, 2013.
[49] L. Nguyen, M. Ogburn, W. Gilbert, K. Kibler, P. Brown, P. Deal, Nasa Technical Paper 1538-Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane with Relaxed Longitudinal Static Stability, Tech. rep. NASA, 1979.
[50] R. Van't Veld, E.-J. van Kampen, Q.P. Chu, Stability and robustness analysis and improvements for incremental nonlinear dynamic inversion control, in: 2018 AIAA Guidance, Navigation, and Control Conference, 2018, p. 1127.
[51] D. Simon, Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches, John Wiley & Sons, 2006.