Nonlinear adaptive flight control using incremental approximate dynamic programming and output feedback

Zhou, Y; van Kampen, EJ; Chu, QP

DOI
10.2514/1.G001762

Publication date
2017

Document Version
Accepted author manuscript

Published in
Journal of Guidance, Control, and Dynamics: devoted to the technology of dynamics and control

Citation (APA)
Zhou, Y., van Kampen, EJ., & Chu, QP. (2017). Nonlinear adaptive flight control using incremental approximate dynamic programming and output feedback. *Journal of Guidance, Control, and Dynamics: devoted to the technology of dynamics and control*, *40*, 493-500. https://doi.org/10.2514/1.G001762

This is an Accepted Manuscript of an article published in: Journal of Guidance, Control, and Dynamics

Available online: https://arc.aiaa.org/doi/10.2514/1.G001762

Nonlinear Adaptive Flight Control Using Incremental Approximate Dynamic Programming and Output Feedback

Ye Zhou*, Erik-Jan van Kampen† and QiPing Chu‡

Delft University of Technology, 2629HS Delft, The Netherlands

I. Introduction

Model-free adaptive control approaches are worthwhile to be investigated for fault tolerant flight control due to many unsolved challenges in model-based strategies.[1–8] Reinforcement Learning (RL) controllers have been proposed to solve nonlinear, optimal control problems without using accurate system models.[9,10] Traditional RL, for solving optimality problems, is an off-line method using an n-dimensional look-up table for all possible state vectors, which may cause the "curse of dimensionality".[11,12]

To tackle the "curse of dimensionality", numerical methods, such as Approximate Dynamic Programming (ADP), have been developed to solve the optimality problem,[12,13] by applying a function approximator with parameters to approximate the value/cost function. Searching for an applicable structure and for the parameters of the function approximator is a global optimization problem as these approximators are in general highly nonlinear. For the special case when the dynamics of the system are linear, Dynamic Programming (DP) gives a complete and explicit solution, because the one-step state cost and the cost function in this case are quadratic.[13] For general nonlinear control problems, DP is difficult to carry out and ADP designs are not systematic.[11]

Considering the design challenges mentioned above, trade-off solutions which may lead to simple and systematic designs are extremely attractive. Some successful approaches have been reported lately.[14–17] In this paper, an incremental ADP (iADP) model-free adaptive control approach is developed for nonlinear systems. This control approach is inspired by the ideas and solutions given by several articles[13,17–20] . It starts with the selection of the cost function in a systematic way,[13] and follows with the Linear ADP (LADP) model-free adaptive control approach.[17] As the plant to be controlled in this paper is nonlinear, the iADP is developed based on the linearized incremental model of the original nonlinear system.[18–20]

The incremental form of a nonlinear dynamic system is actually a linear time-varying approximation of the original system assuming sufficiently high sample rate for the discretization.[18–20] Combining LADP and the incremental form of the system to be controlled leads to a new nonlinear adaptive control algorithm iADP. It retains the advantages of LADP with a systematic formulation of cost function approximations for nonlinear systems, while keeping the closed-loop system optimized.

Classical ADP methods assume that the system is fully observable and that the observed states obey a Markov process. The problems of partial/imperfect information and unmeasurable state vector estimation are very challenging and demanded to be solved in numerous applications.[21] Many researches have already taken presence of stochastic, time-varying wind disturbance into account as a general problem in practical navigation and guidance control.[22,23] Despite that, parametrized output feedback controllers have been designed to deal with problems without full state information and to achieve finite time stability based on observers.[24–29] However, these methods still need a priori knowledge or/and assumption of the system model structure.

*PhD student, Control and Operation Department, Aerospace Engineering, Delft University of Technology, AIAA student member.

†Assistant Professor, Control and Operation Department, Aerospace Engineering, Delft University of Technology, AIAA member.

‡Associate Professor, Control and Operation Department, Aerospace Engineering, Delft University of Technology, AIAA member.

Other than that, output feedback approximate dynamic programming algorithms[17] have been proposed, as opposed to full state feedback, to tackle problems without direct state observations. These algorithms do not require any a priori knowledge of the system or engineering knowledge to design control parameters, or even a separate observer. However, these algorithms are derived for affine in control input linear time-invariant (LTI) systems. This paper starts with an algorithm development combining ADP and the incremental approach assuming direct availability of full state observation.[30] Following is the core contribution of the paper, in which an iADP algorithm based on output feedback is designed by applying the output and input measurement to reconstruct the full state.

II. Incremental Approximate Dynamic Programming

Incremental methods are able to deal with nonlinear systems. These methods compute the required control increment at a certain moment using the conditions of the system in the instant before.[19] Aircraft models are highly nonlinear and can be generally given as follows:

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

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

where Eq. (1) is the system dynamic equation, in which $f[\mathbf{x}(t), \mathbf{u}(t)] \in \mathcal{R}^n$ provides the physical evaluation of n states over time, Eq. (2) is the output (observation) equation, which can be measured using sensors, and $h[\mathbf{x}(t)] \in \mathcal{R}^p$ is a vector denoting p measured outputs.

The system dynamics around the condition of the system at time t_0 can be linearized by using the first-order Taylor series expansion:

$$\dot{\mathbf{x}}(t) \simeq \dot{\mathbf{x}}(t_0) + F[\mathbf{x}(t_0), \mathbf{u}(t_0)][\mathbf{x}(t) - \mathbf{x}(t_0)] + G[\mathbf{x}(t_0), \mathbf{u}(t_0)][\mathbf{u}(t) - \mathbf{u}(t_0)], \tag{3}$$

where $F[\mathbf{x}(t), \mathbf{u}(t)] = \frac{\partial f[\mathbf{x}(t), \mathbf{u}(t)]}{\partial \mathbf{x}(t)} \in \mathcal{R}^{n \times n}$ is the system matrix of the linearized model at time t, and $G[\mathbf{x}(t), \mathbf{u}(t)] = \frac{\partial f[\mathbf{x}(t), \mathbf{u}(t)]}{\partial \mathbf{u}(t)} \in \mathcal{R}^{n \times m}$ is the control effectiveness matrix of the linearized model at time t.

It is assumed that the control inputs, states, and state derivatives of the system are measurable. Under this assumption, the model around time t_0 can be written in an incremental form:

$$\Delta\dot{\mathbf{x}}(t) \simeq F[\mathbf{x}(t_0), \mathbf{u}(t_0)]\Delta\mathbf{x}(t) + G[\mathbf{x}(t_0), \mathbf{u}(t_0)]\Delta\mathbf{u}(t). \tag{4}$$

This linearized incremental model is identifiable by using least squares (LS) techniques.

A. Incremental Approximate Dynamic Programming Based on Full State Feedback

Physical systems are generally continuous, but the collected data are discrete samples. It is assumed that the control system has a constant high sampling frequency. Thus, the nonlinear system can be written in a discrete form:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t), \tag{5}$$

$$\mathbf{y}_t = h(\mathbf{x}_t). \tag{6}$$

When the system has a direct availability of full state observation, the output equation can be written as

$$\mathbf{y}_t = \mathbf{x}_t. \tag{7}$$

By taking the Taylor expansion, the linearized discrete model of this nonlinear system around $\mathbf{x}_{t-1}$, which approximates $\mathbf{x}_t$, can also be written in an incremental form:

$$\Delta\mathbf{x}_{t+1} \simeq F_{t-1}\Delta\mathbf{x}_t + G_{t-1}\Delta\mathbf{u}_t, \tag{8}$$

where $\Delta\mathbf{x}_t = \mathbf{x}_t - \mathbf{x}_{t-1}$, $\Delta\mathbf{u}_t = \mathbf{u}_t - \mathbf{u}_{t-1}$, $F_{t-1} = \frac{\partial f(\mathbf{x},\mathbf{u})}{\partial \mathbf{x}}|_{\mathbf{x}_{t-1},\mathbf{u}_{t-1}} \in \mathcal{R}^{n \times n}$ is the system matrix, and $G_{t-1} = \frac{\partial f(\mathbf{x},\mathbf{u})}{\partial \mathbf{u}}|_{\mathbf{x}_{t-1},\mathbf{u}_{t-1}} \in \mathcal{R}^{n \times m}$ is the control effectiveness matrix at time step $t-1$. Because of the high frequency sample data and slow-varying system, the current linearized model (F_{t-1}, G_{t-1}) can be identified from M different data points using a piecewise sequential LS method.[30,31] Because there are $n+m$ parameters in the ith row, M needs to satisfy $M \geq (n+m)$.

To minimize the cost of the system to reach its goal, the *one-step cost function* is defined quadratically:

$$c_t = c(\mathbf{y}_t, \mathbf{u}_t, \mathbf{d}_t) = (\mathbf{y}_t - \mathbf{d}_t)^T Q(\mathbf{y}_t - \mathbf{d}_t) + \mathbf{u}_t^T R\mathbf{u}_t, \tag{9}$$

where Q and R are positive definite matrices, and $\mathbf{d}_t$ denotes the desired output. Considering a stabilizing control problem, the one-step cost function at time t can be written as

$$c_t = c(\mathbf{y}_t, \mathbf{u}_t) = \mathbf{y}_t^T Q\mathbf{y}_t + \mathbf{u}_t^T R\mathbf{u}_t. \tag{10}$$

For infinite horizons, the cost-to-go function is the cumulative future reward from any initial state $\mathbf{x}_t$:

$$\begin{aligned} J^\mu(\mathbf{x}_t) &= \sum_{i=t}^{\infty} \gamma^{i-t}(\mathbf{y}_i^T Q\mathbf{y}_i + \mathbf{u}_i^T R\mathbf{u}_i) \\ &= \mathbf{y}_t^T Q\mathbf{y}_t + (\mathbf{u}_{t-1} + \Delta\mathbf{u}_t)^T R(\mathbf{u}_{t-1} + \Delta\mathbf{u}_t) + \gamma J^\mu(\mathbf{x}_{t+1}), \end{aligned} \tag{11}$$

where μ is the current *policy* (control law) for this iADP algorithm, $\gamma \in [0, 1]$ is a parameter called the *discounted rate* or the *forgetting factor*. The cost-to-go function for the *optimal policy* μ^* is defined as follows:

$$J^*(\mathbf{x}_t) = \min_{\Delta\mathbf{u}_t} \left[\mathbf{y}_t^T Q\mathbf{y}_t + (\mathbf{u}_{t-1} + \Delta\mathbf{u}_t)^T R(\mathbf{u}_{t-1} + \Delta\mathbf{u}_t) + \gamma J^*(\mathbf{x}_{t+1})\right]. \tag{12}$$

And the policy μ is defined as feedback control in an incremental form:

$$\Delta\mathbf{u}_t = \mu(\mathbf{u}_{t-1}, \mathbf{x}_t, \Delta\mathbf{x}_t). \tag{13}$$

The optimal policy at time t is given by

$$\mu^* = \arg \min_{\Delta\mathbf{u}_t} \left[\mathbf{y}_t^T Q\mathbf{y}_t + (\mathbf{u}_{t-1} + \Delta\mathbf{u}_t)^T R(\mathbf{u}_{t-1} + \Delta\mathbf{u}_t) + \gamma J^*(\mathbf{x}_{t+1})\right]. \tag{14}$$

When the dynamics of the system are linear, this problem is known as the linear-quadratic regulator (LQR) control problem. For this nonlinear case, the cost-to-go is the sum of quadratic values in the outputs and inputs with a forgetting factor. Thus, the cost-to-go $J^\mu(\mathbf{x}_t)$ should always be positive. In general, ADP uses a surrogate cost function approximating the true cost-to-go. The goal is to capture its key features instead of accurately approximating the true cost-to-go. In many practical cases, even for time-varying systems, simple quadratic cost function approximations are chosen so that the evaluation step can be exactly carried out and the optimization problem is reduced to be tractable.[13] A systematic cost function approximation applied in this paper is chosen to be quadratic in $\mathbf{x}_t$ for some symmetric, positive definite matrix P:

$$\widehat{J}^\mu(\mathbf{x}_t) = \mathbf{x}_t^T P\mathbf{x}_t. \tag{15}$$

This quadratic cost function approximation has an additional, important benefit for this approximately convex state-cost system with a fixed minimum value. To be specific, this system has an optimal state when it reaches the desired state and keeps it. The true cost function has many local minima elsewhere because of the nonlinearity. On the other hand, this quadratic cost function has only one local minimum, which is also the global one. Therefore, this quadratic form helps to prevent the policy from going into any other local minimum. The learned symmetric, positive definite P matrix guarantees progressive optimization of the policy.

The LQR Bellman equation for $\widehat{J}^\mu$ in the incremental form becomes

$$\begin{aligned} \widehat{J}^\mu(\mathbf{x}_t) \& &= \mathbf{y}_t^T Q\mathbf{y}_t + (\mathbf{u}_{t-1} + \Delta\mathbf{u}_t)^T R(\mathbf{u}_{t-1} + \Delta\mathbf{u}_t) \\ \& \quad &+ \gamma(\mathbf{x}_t + F_{t-1}\Delta\mathbf{x}_t + G_{t-1}\Delta\mathbf{u}_t)^T P(\mathbf{x}_t + F_{t-1}\Delta\mathbf{x}_t + G_{t-1}\Delta\mathbf{u}_t). \end{aligned} \tag{16}$$

By setting the derivative with respect to $\Delta\mathbf{u}_t$ to zero, the *optimal control* can be obtained:

$$\Delta\mathbf{u}_t = -(R + \gamma G_{t-1}^T P G_{t-1})^{-1}[R\mathbf{u}_{t-1} + \gamma G_{t-1}^T P\mathbf{x}_t + \gamma G_{t-1}^T P F_{t-1}\Delta\mathbf{x}_t]. \tag{17}$$

From Eq. (17), it can be concluded that the policy is in the form of system variables $(\mathbf{u}_{t-1}, \mathbf{x}_t, \Delta\mathbf{x}_t)$ feedback, and the gains are functions of the dynamics of the current linearized system (F_{t-1}, G_{t-1}).

Opposite to the model-based control algorithms with on-line identification of nonlinear systems, availability of these local linear models is sufficient for iADP algorithms. Furthermore, the determination of the linear model structure is much simpler than the identification of the nonlinear model structure. If the nonlinear model is unknown, while the full state is measurable, the iADP algorithm, as shown below, can be applied to improve the policy online.

iADP algorithm based on full state feedback (iADP-FS)

Evaluation. The cost function kernel matrix P under policy μ can be evaluated and updated recursively to Bellman equation for each iteration $j = 0, 1, ...$ until convergence:

$$\mathbf{x}_t^T P^{(j+1)} \mathbf{x}_t = \mathbf{y}_t^T Q \mathbf{y}_t + \mathbf{u}_t^T R \mathbf{u}_t + \gamma \mathbf{x}_{t+1}^T P^{(j)} \mathbf{x}_{t+1}. \tag{18}$$

Policy improvement. Policy improves for the new kernel matrix $P^{(j+1)}$:

$$\Delta \mathbf{u}_t = -(R + \gamma G_{t-1}^T P^{(j+1)} G_{t-1})^{-1} [R \mathbf{u}_{t-1} + \gamma G_{t-1}^T P^{(j+1)} \mathbf{x}_t + \gamma G_{t-1}^T P^{(j+1)} F_{t-1} \Delta \mathbf{x}_t]. \tag{19}$$

When Δt approximates to 0, the identified incremental model F_{t-1}, G_{t-1} and the prediction of the next state approximate the true values. With this linearized model, this problem locally becomes an LQR problem. Referring to optimal control problems, the policy designed above approaches the optimal policy as $\gamma = 1$. However, in ADP, the discount factor γ is usually chosen as $\gamma \in (0, 1)$, so that the infinite sum has a finite value as long as the cost sequence is bounded, and the agent is not 'myopic' in being concerned only with maximizing immediate cost.[9]

B. Incremental Approximate Dynamic Programming Based on Output Feedback

The full state of a system, such as an air vehicle system, is often not available. In addition, agents often try to control a system without enough information to infer its real states.[21] The partially observable Markov decision process (POMDP) framework can be used to deal with stochastic systems. For deterministic systems, these types of methods are often referred to as output feedback. The systems still need to be observable, which means that the unmeasurable internal states (full states) can be reconstructed with the observations over a long enough time horizon. For model-free methods, the system is observable when the observability matrix has a full column rank.

Considering the nonlinear system again, see Eq. (5) and (6), the *output (observation)* around $\mathbf{x}_{t-1}$ can also be linearized with Taylor expansion:

$$\Delta \mathbf{y}_t \simeq H_{t-1} \Delta \mathbf{x}_t, \tag{20}$$

where $H_{t-1} = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}}|_{\mathbf{x}_{t-1}} \in \mathcal{R}^{p \times n}$ is the *observation matrix* at time step $t-1$. The nonlinear system incremental dynamics, see Eq. (8) and (20), at current time t can be represented by the previously measured data on time horizon [t-N, t]:

$$\Delta \mathbf{x}_t \simeq \widetilde{F}_{t-2,t-N-1} \cdot \Delta \mathbf{x}_{t-N} + U_N \cdot \overline{\Delta \mathbf{u}}_{t-1,t-N}, \tag{21}$$

$$\overline{\Delta \mathbf{y}}_{t,t-N+1} \simeq V_N \cdot \Delta \mathbf{x}_{t-N} + T_N \cdot \overline{\Delta \mathbf{u}}_{t-1,t-N}, \tag{22}$$

where symbol $\widetilde{F}_{t-a,t-b} = \prod_{i=t-a}^{t-b} F_i = F_{t-a} \cdot \ \cdots \ \cdot F_{t-b}$,

$$\overline{\Delta \mathbf{u}}_{t-1,t-N} = \begin{bmatrix} \Delta \mathbf{u}_{t-1} \\ \Delta \mathbf{u}_{t-2} \\ \vdots \\ \Delta \mathbf{u}_{t-N} \end{bmatrix} \in \mathcal{R}^{mN}, \qquad \overline{\Delta \mathbf{y}}_{t,t-N+1} = \begin{bmatrix} \Delta \mathbf{y}_t \\ \Delta \mathbf{y}_{t-1} \\ \vdots \\ \Delta \mathbf{y}_{t-N+1} \end{bmatrix} \in \mathcal{R}^{mN},$$

$U_N = \begin{bmatrix} G_{t-2} & F_{t-2} G_{t-3} & ... & \widetilde{F}_{t-2,t-N} \cdot G_{t-N-1} \end{bmatrix} \in \mathcal{R}^{n \times mN}$ is the *controllability matrix*,

$$V_N = \begin{bmatrix} H_{t-1} \widetilde{F}_{t-2,t-N-1} \\ H_{t-2} \widetilde{F}_{t-3,t-N-1} \\ \vdots \\ H_{t-N} F_{t-N-1} \end{bmatrix} \in \mathcal{R}^{pN \times n} \text{ is the } \textit{observability matrix},$$

$$T_N = \begin{bmatrix} H_{t-1}G_{t-2} & H_{t-1}F_{t-2}G_{t-3} & H_{t-1}\widetilde{F}_{t-2,t-3}G_{t-4} & \cdots & H_{t-2}\widetilde{F}_{t-3,t-N}\cdot G_{t-N-1} \\ 0 & H_{t-2}G_{t-3} & H_{t-2}F_{t-3}G_{t-4} & \cdots & H_{t-2}\widetilde{F}_{t-3,t-N}\cdot G_{t-N-1} \\ 0 & 0 & H_{t-3}G_{t-4} & \cdots & H_{t-3}\widetilde{F}_{t-4,t-N}\cdot G_{t-N-1} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & H_{t-N}\cdot G_{t-N-1} \end{bmatrix} \in \mathcal{R}^{pN\times mN}.$$

When the system is fully observable, the left inverse of V_N, which has a full column rank, can be obtained:

$$V_N^{left} = (V_N^T V_N)^{-1} V_N^T. \tag{23}$$

To have a full column rank for observability matrix V_N, N needs to satisfy $N \geq n/p$. Making the number of parameters to be identified as small as possible, the smallest value for N which meets $N \geq n/p$ is usually selected.

By left-multiplying V_N^{left} to Eq. (22), and then substituting the equation of $\Delta\mathbf{x}_{t-N}$ into Eq. (21), the incremental state can be reconstructed uniquely as a function of the past input/output:

$$\begin{aligned} \Delta\mathbf{x}_t &\simeq \widetilde{F}_{t-2,t-N-1}\cdot V_N^{left}\cdot\overline{\Delta\mathbf{y}}_{t,t-N+1} + (U_N - \widetilde{F}_{t-2,t-N-1}\cdot V_N^{left}\cdot T_N)\cdot\overline{\Delta\mathbf{u}}_{t-1,t-N} \\ &= \begin{bmatrix} M_{\Delta u} & M_{\Delta y} \end{bmatrix}\begin{bmatrix} \overline{\Delta\mathbf{u}}_{t-1,t-N} \\ \overline{\Delta\mathbf{y}}_{t,t-N+1} \end{bmatrix} \\ &= M_{t-1}\overline{\Delta\mathbf{z}}_{t,t-N}, \end{aligned} \tag{24}$$

where $M_{\Delta y}$ denotes $\widetilde{F}_{t-2,t-N-1}\cdot V_N^{left} \in \mathcal{R}^{n\times pN}$, $M_{\Delta u}$ denotes $U_N - M_{\Delta y}T_N \in \mathcal{R}^{n\times mN}$, and $M_{t-1} = [M_{\Delta u}\ \ M_{\Delta y}] \in \mathcal{R}^{n\times(m+p)N}$. The matrix M_{t-1} is identifiable by using previous $\widehat{M}$ steps with $\widehat{M} \geq (m+p)N$.

The nonlinear incremental output equation, Eq. (20), can be represented by a history of measured input/output data on time horizon [t-N, t-1] in another form:

$$\overline{\Delta\mathbf{y}}_{t-1,t-N} \simeq \overline{V}_N\cdot\Delta\mathbf{x}_{t-N} + \overline{T}_N\cdot\overline{\Delta\mathbf{u}}_{t-1,t-N}, \tag{25}$$

where $\overline{V}_N = \begin{bmatrix} H_{t-2}\widetilde{F}_{t-3,t-N-1} \\ H_{t-3}\widetilde{F}_{t-3,t-N-1} \\ \vdots \\ H_{t-N-1} \end{bmatrix} \in \mathcal{R}^{pN\times n}$,

$$\overline{T}_N = \begin{bmatrix} 0 & H_{t-2}G_{t-3} & H_{t-2}F_{t-3}G_{t-4} & \cdots & H_{t-2}\widetilde{F}_{t-3,t-N}\cdot G_{t-N-1} \\ 0 & 0 & H_{t-3}G_{t-4} & \cdots & H_{t-3}\widetilde{F}_{t-4,t-N}\cdot G_{t-N-1} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & H_{t-N}\cdot G_{t-N-1} \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \in \mathcal{R}^{pN\times mN}.$$

When the system is fully observable, the left inverse of $\overline{V}_N$, which also has a full column rank, can be obtained:

$$\overline{V}_N^{left} = (\overline{V}_N^T\overline{V}_N)^{-1}\overline{V}_N^T. \tag{26}$$

Left-multiplying $\overline{V}_N^{left}$ to Eq. (25) and substituting the resulted $\Delta\mathbf{x}_{t-N}$ into Eq. (21) and then the resulted $\Delta\mathbf{x}_t$ into Eq. (20), the dynamics of the output and of previous measured data can be obtained:

$$\begin{aligned} \Delta\mathbf{y}_t &\simeq (H_{t-1}U_N - H_{t-1}\widetilde{F}_{t-2,t-N-1}\cdot\overline{V}_N^{left}\overline{T}_N)\cdot\overline{\Delta\mathbf{u}}_{t-1,t-N} \\ &\quad + H_{t-1}\widetilde{F}_{t-2,t-N-1}\overline{V}_N^{left}\cdot\overline{\Delta\mathbf{y}}_{t-1,t-N} \\ &= \underline{F}_{t-1}\cdot\overline{\Delta\mathbf{u}}_{t-1,t-N} + \underline{G}_{t-1}\cdot\overline{\Delta\mathbf{y}}_{t-1,t-N}. \end{aligned} \tag{27}$$

The output increment $\Delta\mathbf{y}_{t+1}$ can also be reconstructed uniquely as a function of the measured input/output data of N previous steps:

$$\begin{aligned} \Delta\mathbf{y}_{t+1} &\simeq \underline{F}_t\cdot\overline{\Delta\mathbf{u}}_{t,t-N+1} + \underline{G}_t\cdot\overline{\Delta\mathbf{y}}_{t,t-N+1} \\ &= \underline{F}_{t,11}\cdot\Delta\mathbf{u}_t + \underline{F}_{t,12}\cdot\overline{\Delta\mathbf{u}}_{t-1,t-N+1} + \underline{G}_t\cdot\overline{\Delta\mathbf{y}}_{t,t-N+1}, \end{aligned} \tag{28}$$

where $\underline{F}_t \in \mathcal{R}^{p\times Nm}$ is the extended system matrix, $\underline{G}_t \in \mathcal{R}^{p\times Np}$ is the extended control effectiveness matrix, $\underline{F}_{t,11} \in \mathcal{R}^{p\times m}$ and $\underline{F}_{t,12} \in \mathcal{R}^{p\times(N-1)m}$ are partitioned matrices from $\underline{F}_t$. $\underline{F}_t$ and $\underline{G}_t$ are identifiable by using the piecewise sequential LS method.[30,31] In this case, there are $(m+p)N$ parameters in each row. Therefore, the number of previous data samples M needs to satisfy $M \geq (m+p)N$.

It is assumed that the cost-to-go of the system state at time t can be written as a function of a symmetric expended kernel matrix $\overline{P}$ in the quadratic form in terms of a history of observation vectors $\overline{\mathbf{z}}_{t,t-N} = [\overline{\mathbf{u}}^T_{t-1,t-N}, \overline{\mathbf{y}}^T_{t,t-N+1}]^T$:

$$\widehat{J}^{\mu}(\mathbf{z}_{t,t-N}) = \overline{\mathbf{z}}^T_{t,t-N}\overline{P}\overline{\mathbf{z}}_{t,t-N}. \tag{29}$$

The optimal policy under the estimation of $\overline{P}$ in terms of $\overline{\mathbf{z}}_{t,t-N}$ is rewritten to be

$$\mu^* = \arg \min_{\Delta\mathbf{u}_t} (\mathbf{y}_t^T Q\mathbf{y}_t + \mathbf{u}_t^T R\mathbf{u}_t + \gamma\overline{\mathbf{z}}^T_{t+1,t-N+1}\ \overline{P}\ \overline{\mathbf{z}}_{t+1,t-N+1}), \tag{30}$$

where

$$\overline{\mathbf{z}}^T_{t+1,t-N+1}\ \overline{P}\ \overline{\mathbf{z}}_{t+1,t-N+1} = \begin{bmatrix} \mathbf{u}_{t-1}+\Delta\mathbf{u}_t \\ \overline{\mathbf{u}}_{t-1,t-N+1} \\ \mathbf{y}_t+\Delta\mathbf{y}_{t+1} \\ \overline{\mathbf{y}}_{t,t-N+2} \end{bmatrix}^T \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{12}^T & P_{22} & P_{23} & P_{24} \\ P_{13}^T & P_{23}^T & P_{33} & P_{34} \\ P_{14}^T & P_{24}^T & P_{34}^T & P_{44} \end{bmatrix} \begin{bmatrix} \mathbf{u}_{t-1}+\Delta\mathbf{u}_t \\ \overline{\mathbf{u}}_{t-1,t-N+1} \\ \mathbf{y}_t+\Delta\mathbf{y}_{t+1} \\ \overline{\mathbf{y}}_{t,t-N+2} \end{bmatrix}. \tag{31}$$

By differentiating with respect to $\Delta\mathbf{u}_t$, the policy improvement step can be obtained in terms of the measured data:

$$\begin{aligned} &-[R+\gamma P_{11}+\gamma(\underline{F}_{t,11})^T\cdot P_{33}\cdot\underline{F}_{t,11}+\gamma P_{13}\underline{F}_{t,11}+\gamma(P_{13}\underline{F}_{t,11})^T]\cdot\Delta\mathbf{u}_t \\ &=[R+\gamma P_{11}+\gamma(\underline{F}_{t,11})^T\cdot P_{13}^T]\mathbf{u}_{t-1}+\gamma[(\underline{F}_{t,11})^T P_{33}+P_{13}]\mathbf{y}_t \\ &\quad+\gamma[P_{12}+(\underline{F}_{t,11})^T\cdot P_{23}^T]\overline{\mathbf{u}}_{t-1,t-N+1}+\gamma[P_{14}+(\underline{F}_{t,11})^T\cdot P_{34}]\overline{\mathbf{y}}_{t,t-N+2} \\ &\quad+\gamma[(\underline{F}_{t,11})^T P_{33}+P_{13}](\underline{F}_{t,12}\cdot\overline{\Delta\mathbf{u}}_{t-1,t-N+1}+\underline{G}_t\cdot\overline{\Delta\mathbf{y}}_{t,t-N+1}). \end{aligned} \tag{32}$$

If the nonlinear model is unknown, and only partial information about the states is accessible, the output feedback ADP algorithm combined with the incremental method can be applied to improve the policy online.

iADP algorithm based on output feedback (iADP-OP)

Evaluation. The cost function kernel matrix $\overline{P}$ under policy μ can be evaluated and updated recursively according to Bellman equation for each iteration $j = 0, 1, ...$ until convergence:

$$\mathbf{z}'^T_{t,t-N+1}\overline{P}^{(j+1)}\mathbf{z}'_{t,t-N+1} = \mathbf{y}_t^T Q\mathbf{y}_t + \mathbf{u}_t^T R\mathbf{u}_t + \gamma\mathbf{z}'^T_{t+1,t-N+2}\overline{P}^{(j)}\mathbf{z}'_{t+1,t-N+2}. \tag{33}$$

Policy improvement. The policy improves for the new kernel matrix $\overline{P}^{(j+1)}$ according to the derived optimal control policy:

$$\begin{aligned} \Delta\mathbf{u}_t = &-[R+\gamma P_{11}+\gamma(\underline{F}_{t,11})^T\cdot P_{33}\cdot\underline{F}_{t,11}+\gamma P_{13}\underline{F}_{t,11}+\gamma(P_{13}\underline{F}_{t,11})^T]^{-1}\cdot \\ &\{[R+\gamma P_{11}+\gamma(\underline{F}_{t,11})^T\cdot P_{13}^T]\mathbf{u}_{t-1}+\gamma[(\underline{F}_{t,11})^T P_{33}+P_{13}]\mathbf{y}_t \\ &+\gamma[P_{12}+(\underline{F}_{t,11})^T\cdot P_{23}^T]\overline{\mathbf{u}}_{t-1,t-N+1}+\gamma[P_{14}+(\underline{F}_{t,11})^T\cdot P_{34}]\overline{\mathbf{y}}_{t,t-N+2} \\ &+\gamma[(\underline{F}_{t,11})^T P_{33}+P_{13}](\underline{F}_{t,12}\cdot\overline{\Delta\mathbf{u}}_{t-1,t-N+1}+\underline{G}_t\cdot\overline{\Delta\mathbf{y}}_{t,t-N+1})\}. \end{aligned} \tag{34}$$

Approximating Δt to 0, the policy designed above approaches the optimal policy.

III. Numerical Experiments and Results

This section applies both iADP-FS and iADP-OF algorithms on a simulation of controlling an aerospace related model. This is to show how the algorithms perform in stabilizing and regulating the system in presence of input disturbances and an initial offset.

A. Air vehicle model

A nonlinear air vehicle simulation model is used in this section. Air vehicle models can be highly nonlinear and are generally given as follows:

$$\dot{\mathbf{x}}(t) = f[\mathbf{x}(t), \mathbf{u}(t) + \mathbf{w}(t)], \tag{35}$$

$$\mathbf{y}(t) = h[\mathbf{x}(t)], \tag{36}$$

where Eq. (35) is the kinematic state equation, $\mathbf{w}(t)$ is the external disturbance, which is set to be caused only by the input noise, Eq. (36) is the output (observation) equation.

As an application, only the elevator deflection will be regulated as pitch control to stabilize the air vehicle. Thus, two longitudinal states, angle of attack α and pitch rate q (i.e. $\mathbf{x} = [\alpha \;\; q]$), and one control input, the elevator deflection angle δ_e, are concerned.

The nonlinear model in the pitch plane is simulated around a steady wings-level flight condition:

$$\dot{\alpha} = q + \frac{\bar{q}S}{m_a V_T} \, C_z(\alpha, q, M_a, \delta_e), \tag{37}$$

$$\dot{q} = \frac{\bar{q}Sd}{I_{yy}} \, C_m(\alpha, q, M_a, \delta_e), \tag{38}$$

where $\bar{q}$ is dynamic pressure, S is reference area, m_a is mass, V_T is speed, d is reference length, and I_{yy} is pitching moment of inertia. C_z and C_m are the aerodynamic force and moment coefficients, which are highly nonlinear functions. As a preliminary test, an air vehicle model[32,33] is taken in the pitch plane for $-10^o < \alpha < 10^o$.

When the input is $\mathbf{u}(t) = 0$, $\alpha = 0$ and $q = 0$ form an equilibrium of the system. The flight control task is to stabilize the system (i.e., a regulator problem), if there is any input disturbance or any offset from this condition. Specifically, an optimal policy μ^* and the associated optimum performance need to be found by minimizing the state-cost function J.[34]

B. Results

1. *IADP algorithm based on full state feedback*

As with other ADP methods, good state-cost estimation depends heavily on the exploration of the state space, which is represented by persistent excitation in this case. An amplitude varying multiple doublet disturbance was used this numerical experiment to test the performance of the proposed controllers. Fig. 1 shows the response when a 3211 input disturbance is introduced. The control system trained with iADP algorithm rejects the disturbance compared to the response with the initial policy.

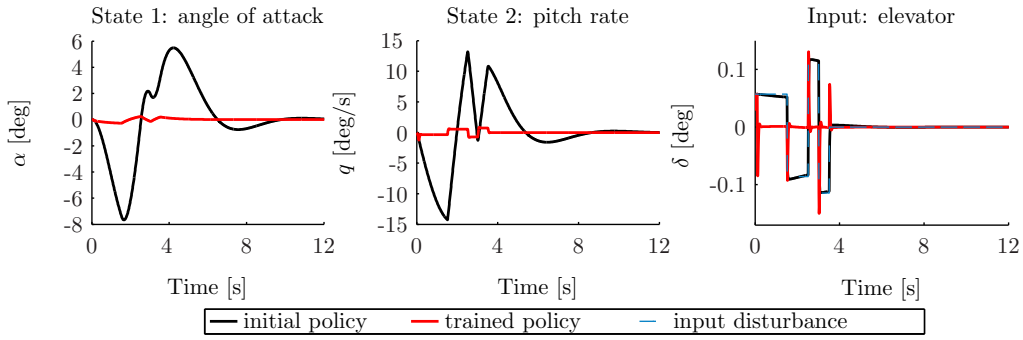

Figure 1. IADP-FS applied to nonlinear aircraft model with 3211 input disturbance

Fig. 2 shows the control performance when the initial state is an offset after a simulated gust. After training, the information of $G(\mathbf{x}, \mathbf{u})$ and $F(\mathbf{x}, \mathbf{u})$ can be used to estimate the current linearized system when the system cannot be identified using online identification without persistent excitation. Because the iADP method uses a simple quadratic cost function, the policy parameters of kernel matrix P converge after only 2 iterations.

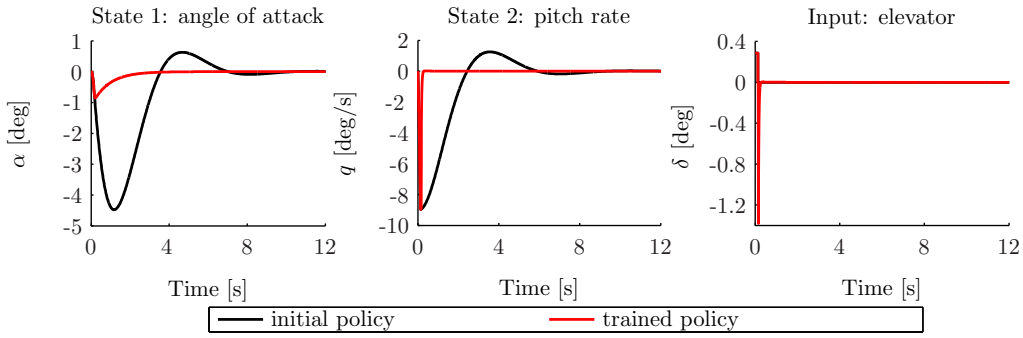

Figure 2. IADP-FS applied to nonlinear aircraft model with an initial offset

This control method does not need the model of the nonlinear system, but still needs the full state to estimate the cost function and the control effectiveness matrix. If the model of the nonlinear system is unknown, and only coupled state information (observations) can be obtained, the iADP algorithm based on output feedback can be used.

2. IADP algorithm based on output feedback

In practice, vane measurement techniques are cost effective in measuring the angle of attack α.[35] Vanes are usually mounted on the aircraft in a location x_{vane} that allows for relatively undisturbed air flow to be measured:

$$\alpha_{measure} \simeq C_c(\alpha + \frac{x_{vane} - x_{cg}}{V} \cdot q), \tag{39}$$

where C_c denotes the calibration coefficient, and x_{cg} is the aircraft center of gravity. As a consequence, the kinematic position error induced by angular velocities q at the vane location has to be considered.

According to this practical case, the output/sensor measurement is set to be a combination of α and q with coefficients. Considering a practical case, which is to regulate α, a big portion of α (0.9) and a small portion of q (0.1) are selected:

$$\mathbf{y}(t) = [c_1 \ c_2] \cdot \mathbf{x}(t) = [0.9 \ 0.1] \cdot \begin{bmatrix} \alpha \\ q \end{bmatrix}. \tag{40}$$

Fig. 3 shows the disturbance response when a 3211 input disturbance was introduced; Fig. 4 shows the control performance when the initial state is an offset from the stable condition after a simulated gust; Fig. 5 shows that the policy parameters of the kernel matrix converge quickly. After only 4 training iterations, the nonlinear system can be regulated, as shown in Fig. 3 and Fig. 4.

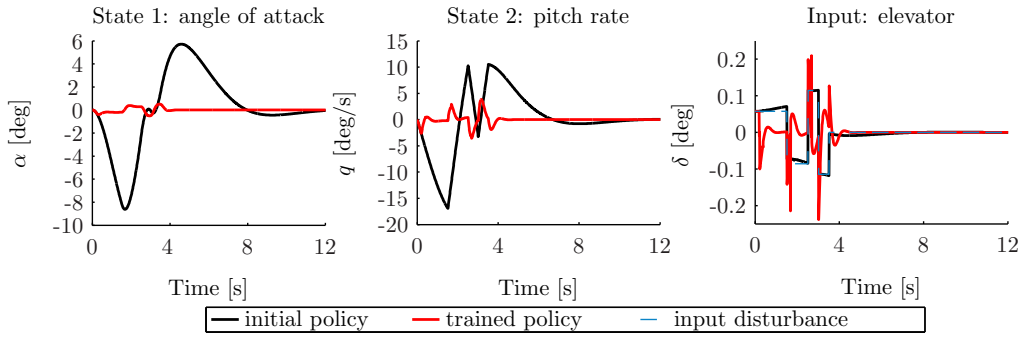

Figure 3. IADP-OP applied to nonlinear aircraft model with 3211 input disturbance

Note that when information of α is available, we can calculate q by using the identified model and enough previously measured observations. With some knowledge or assumptions on the model, the aircraft pitch plane system, α and q, is observable with only information of α. However, the proposed iADP algorithm is a model-free method, i.e., no assumptions about the model are needed, and the observations are from only two

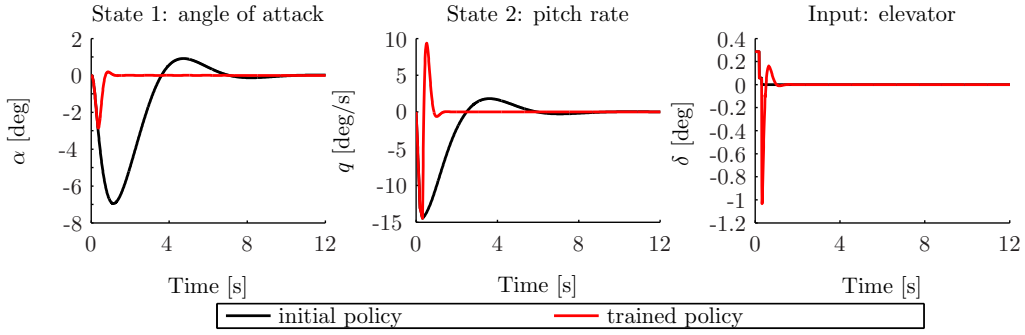

Figure 4. IADP-OP applied to nonlinear aircraft model with an initial offset

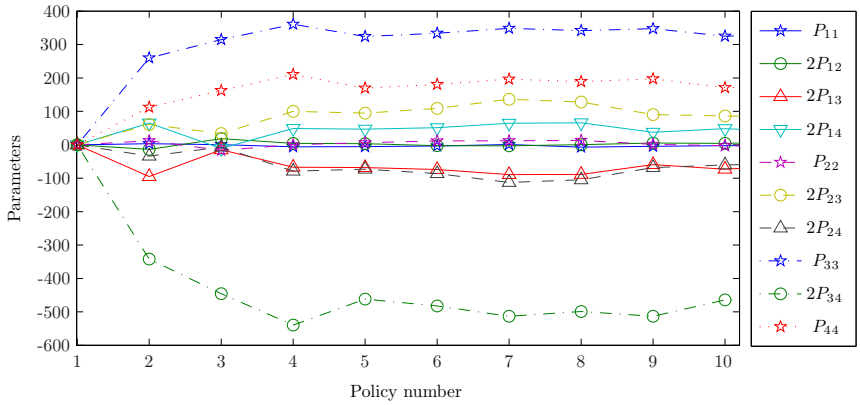

Figure 5. Kernel matrix parameters during training with IADP-OP

samples. Therefore, the observability is defined in terms of whether V_N in Eq. (22) has full column rank. If no information about one of the states can be provided, the iADP algorithm might not be beneficial.

Fig. 6 and Fig. 7 show a comparison of disturbance response and natural response, respectively, among 3 policies. The initial policy is what the original system follows. It cannot compensate for the undesired inputs, such as gusts and ground effects. When the full state is available, the iADP-FS algorithm improves the closed-loop performance, lowers the disturbance response, and stabilizes the system from an offset much quicker. When the full state is unavailable, but the system is observable, the iADP-OP algorithm generates a policy. This policy has an almost equal ability to stabilize and regulate the system to the policy of iADP-FS.

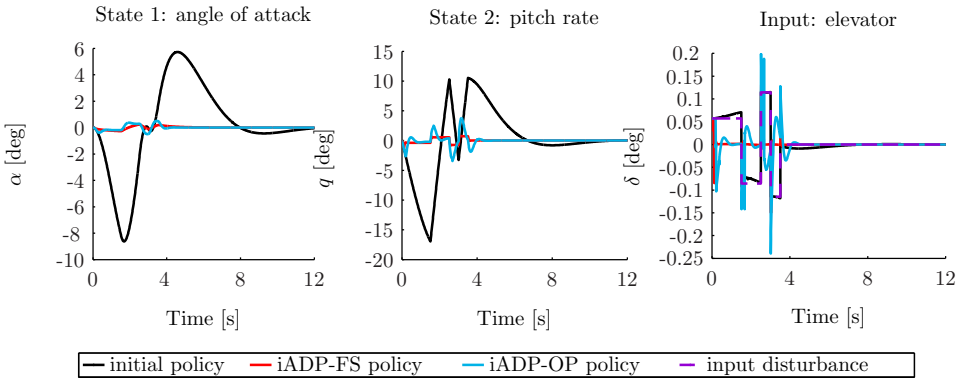

Figure 6. Comparison of policies applied to nonlinear aircraft model with 3211 input disturbance

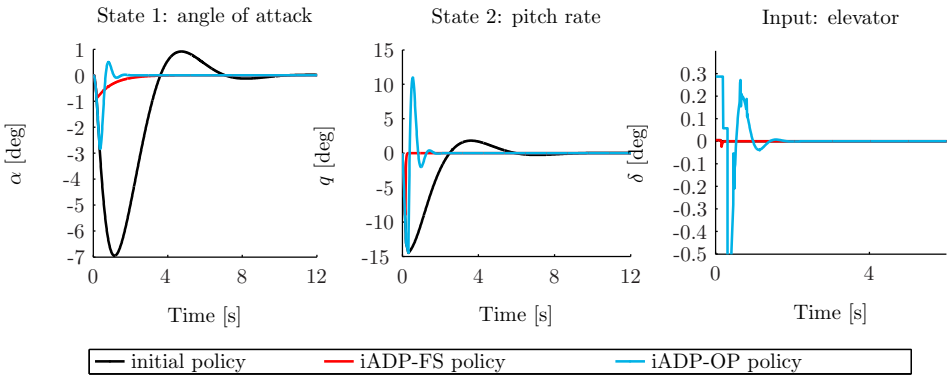

Figure 7. Comparison of policies applied to nonlinear aircraft model with an initial offset

IV. Conclusion

This paper proposes a novel adaptive control method for nonlinear systems, called incremental Approximate Dynamic Programming (iADP). It systematically applies a quadratic cost-to-go function and greatly simplifies the design process of ADP. In addition, the incremental approach can deal with the nonlinearity of systems. The iADP method combines the advantages of both the LADP method and the incremental approach, and provides a model-free, effective adaptive flight controller for nonlinear systems. In addition to the iADP algorithm based on full state feedback (iADP-FS), an iADP algorithm based on output feedback (iADP-OP) is developed. IADP-OP uses only a history of measured input and output data from a dynamical nonlinear system to reconstruct the local model.

Both the iADP-FS algorithm and the iADP-OP algorithm are applied to an aerospace related model. The simulation results show that the trained policy with either algorithm rejects the disturbance compared to the response with the initial policy. This demonstrates that both model-free adaptive control algorithms improve the closed-loop performance of the nonlinear system, while keeping the design process simple and systematic as compared to conventional ADP algorithms.

This new method can potentially design a near-optimal controller for nonlinear systems without a priori knowledge nor full state measurements of the dynamic model. Although still no theoretical guarantees on the nonlinear system performance can be offered, the performance of systems with approximately convex cost functions is observed to be very promising. For general nonlinear systems and more complex tasks, real applications and other possibilities such as piecewise quadratic cost functions will be studied in the future.

Acknowledgement. The first author is financially supported for this Ph.D. research by China Scholarship Council with the project reference number of 201306290026.

References

[1]Lombaerts, T., Oort, E. V., Chu, Q., Mulder, J., and Joosten, D., "Online aerodynamic model structure selection and parameter estimation for fault tolerant control," *Journal of guidance, control, and dynamics*, Vol. 33, No. 3, 2010, pp. 707–723.

[2]Tang, L., Roemer, M., Ge, J., Crassidis, A., Prasad, J., and Belcastro, C., "Methodologies for adaptive flight envelope estimation and protection," *AIAA Guidance, Navigation, and Control Conference*, 2009, p. 6260.

[3]Van Oort, E., Sonneveldt, L., Chu, Q.-P., and Mulder, J., "Full-envelope modular adaptive control of a fighter aircraft using orthogonal least squares," *Journal of guidance, control, and dynamics*, Vol. 33, No. 5, 2010, pp. 1461–1472.

[4]Sghairi, M., De Bonneval, A., Crouzet, Y., Aubert, J., and Brot, P., "Challenges in Building Fault-Tolerant Flight Control System for a Civil Aircraft," *IAENG International Journal of Computer Science*, Vol. 35, No. 4, 2008.

[5]Sonneveldt, L., Van Oort, E., Chu, Q., and Mulder, J., "Nonlinear adaptive trajectory control applied to an F-16 model," *Journal of Guidance, control, and Dynamics*, Vol. 32, No. 1, 2009, pp. 25–39.

[6]Farrell, J., Sharma, M., and Polycarpou, M., "Backstepping-based flight control with adaptive function approximation," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 6, 2005, pp. 1089–1102.

[7]Sonneveldt, L., Van Oort, E., Chu, Q., and Mulder, J., "Comparison of inverse optimal and tuning functions designs for adaptive missile control," *Journal of guidance, control, and dynamics*, Vol. 31, No. 4, 2008, pp. 1176–1182.

[8]Sonneveldt, L., Chu, Q., and Mulder, J., "Nonlinear flight control design using constrained adaptive backstepping," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 2, 2007, pp. 322–336.

[9]Sutton, R. S. and Barto, A. G., *Introduction to reinforcement learning*, MIT Press, 1998.

[10]Bellman, R., *Dynamic Programming*, Princeton University Press, 1957.

[11]Khan, S. G., Herrmann, G., Lewis, F. L., Pipe, T., and Melhuish, C., "Reinforcement learning and optimal adaptive control: An overview and implementation examples," *Annual Reviews in Control*, Vol. 36, No. 1, 2012, pp. 42–59.

[12]Si, J., *Handbook of learning and approximate dynamic programming*, Vol. 2, John Wiley & Sons, 2004.

[13]Keshavarz, A. and Boyd, S., "Quadratic approximate dynamic programming for input-affine systems," *International Journal of Robust and Nonlinear Control*, Vol. 24, No. 3, 2014, pp. 432–449.

[14]Todorov, E. and Li, W., "A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems," *Proceedings of the 2005, American Control Conference, 2005.*, IEEE, 2005, pp. 300–306.

[15]Vrabie, D. and Lewis, F., "Integral reinforcement learning for online computation of feedback Nash strategies of nonzero-sum differential games," *49th IEEE Conference on Decision and Control (CDC)*, IEEE, 2010, pp. 3066–3071.

[16]Morimoto, J. and Atkeson, C. G., "Minimax differential dynamic programming: An application to robust biped walking," 2003.

[17]Lewis, F. L. and Vamvoudakis, K. G., "Reinforcement learning for partially observable dynamic processes: Adaptive dynamic programming using measured output data," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, Vol. 41, No. 1, 2011, pp. 14–25.

[18]Sieberling, S., Chu, Q. P., and Mulder, J. A., "Robust flight control using incremental nonlinear dynamic inversion and angular acceleration prediction," *Journal of guidance, control, and dynamics*, Vol. 33, No. 6, 2010, pp. 1732–1742.

[19]Simplício, P., Pavel, M. D., van Kampen, E., and Chu, Q. P., "An acceleration measurements-based approach for helicopter nonlinear flight control using Incremental Nonlinear Dynamic Inversion," *Control Engineering Practice*, Vol. 21, No. 8, 2013, pp. 1065–1077.

[20]Acquatella, P. J., van Kampen, E., and Chu, Q. P., "Incremental Backstepping for Robust Nonlienar Flight Control," *Proceedings of the EuroGNC 2013*, 2013.

[21]Sigaud, O. and Buffet, O., *Markov decision processes in artificial intelligence*, John Wiley & Sons, 2013.

[22]Bakolas, E. and Tsiotras, P., "Feedback navigation in an uncertain flowfield and connections with pursuit strategies," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 4, 2012, pp. 1268–1279.

[23]Anderson, R. P., Bakolas, E., Milutinović, D., and Tsiotras, P., "Optimal feedback guidance of a small aerial vehicle in a stochastic wind," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 4, 2013, pp. 975–985.

[24]Zou, A.-M. and Kumar, K. D., "Quaternion-based distributed output feedback attitude coordination control for spacecraft formation flying," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 2, 2013, pp. 548–556.

[25]Hu, Q., Jiang, B., and Friswell, M. I., "Robust saturated finite time output feedback attitude stabilization for rigid spacecraft," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 6, 2014, pp. 1914–1929.

[26]Ulrich, S., Sasiadek, J. Z., and Barkana, I., "Nonlinear Adaptive Output Feedback Control of Flexible-Joint Space Manipulators with Joint Stiffness Uncertainties," *Journal of Guidance, Control, and Dynamics*, Vol. 37, No. 6, 2014, pp. 1961–1975.

[27]Mazenc, F. and Bernard, O., "Interval observers for linear time-invariant systems with disturbances," *Automatica*, Vol. 47, No. 1, 2011, pp. 140–147.

[28]Efimov, D., Raïssi, T., Chebotarev, S., and Zolghadri, A., "Interval state observer for nonlinear time varying systems," *Automatica*, Vol. 49, No. 1, 2013, pp. 200–205.

[29]Akella, M. R., Thakur, D., and Mazenc, F., "Partial Lyapunov Strictification: Smooth Angular Velocity Observers for Attitude Tracking Control," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 3, 2015, pp. 442–451.

[30]Zhou, Y., van Kampen, E., and Chu, Q. P., "Incremental Approximate Dynamic Programming for Nonlinear Flight Control Design," *Proceedings of the EuroGNC 2015*, 2015.

[31]Zhou, Y., van Kampen, E., and Chu, Q. P., "Nonlinear Adaptive Flight Control Using Incremental Approximate Dynamic Programming and Output Feedback," *Proc AIAA Guidance Navigation Control Conference*, 2016.

[32]Sonneveldt, L., *Adaptive backstepping flight control for modern fighter aircraft*, TU Delft, Delft University of Technology, 2010.

[33]Kim, S.-H., Kim, Y.-S., and Song, C., "A robust adaptive nonlinear control approach to missile autopilot design," *Control engineering practice*, Vol. 12, No. 2, 2004, pp. 149–154.

[34]Anderson, B. D. and Moore, J. B., *Optimal control: linear quadratic methods*, Courier Corporation, 2007.

[35]Laban, M., *On-line aircraft aerodynamic model identification*, TU Delft, Delft University of Technology, 1994.