# Adaptive Optimal Flight Control for a Fixed-wing Unmanned Aerial Vehicle using Incremental Value Iteration

Li, Y.; van Kampen, E.

**DOI**
10.1109/ICM54990.2023.10101984

**Publication date**
2023

**Document Version**
Final published version

**Published in**
2023 IEEE International Conference on Mechatronics

# Adaptive Optimal Flight Control for a Fixed-wing Unmanned Aerial Vehicle using Incremental Value Iteration

1st Yifei Li
*Department of Control & Operations*
*Delft University of Technology*
Delft, the Netherlands
Y.Li-34@tudelft.nl

2nd Erik-Jan van Kampen
*Department of Control & Operations*
*Delft University of Technology*
Delft, the Netherlands
E.van Kampen@tudelft.nl

2023 IEEE International Conference on Mechatronics (ICM) | 978-1-6654-6661-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICM54990.2023.10101984

***Abstract*—This paper deals with the design of an adaptive optimal controller for a fixed-wing Unmanned Aerial Vehicle(UAV) using an incremental value iteration algorithm. The incremental model is firstly introduced to linearize a nonlinear system. The recursive least squares(RLS) identification algorithm is then used to identify the incremental model. Based on incremental control, the incremental value iteration algorithm is developed for a nonlinear optimal control problem. Moreover, this algorithm is applied to longitudinal attitude tracking of a fixed-wing unmanned aerial vehicle. Simulation results show that the designed adaptive flight controller is robust to variations in initial value of the angle of attack.**

***Index Terms*—adaptive control, optimal control, UAV, value iteration.**

## I. Introduction

Value iteration is an effective algorithm to solve the Hamilton-Jacobi-Bellman (HJB) equation for infinite-time horizon optimal control problems [1]. Value iteration performs a recursive computation between policy evaluation and policy improvement, which is proved to converge to the optimal value function and policy [2]. Compared to policy iteration [3], value iteration requires less computational load, and the initial admissible control law is not needed. Value iteration algorithms have been successfully applied in satellite attitude control [4] and orbital maneuver problems [5].

Incremental control [6] can deal with nonlinear system control through linearization. The incremental model is obtained by taking the Taylor Expansion of the original nonlinear system. Current works have focused on Incremental Nonlinear Dynamic Inversion [7], Incremental Backstepping [8], Incremental Sliding Mode Control [9], and Incremental Approximate Dynamic Programming(IADP) [10]. IADP can learn an optimal control policy, as well as the optimal value function, without knowing the system dynamics. A recursive least squares identification is commonly used in IADP to identify the incremental model.

Because IADP is a model-free method, it can be used in control problems with faults and uncertainties. In [4], a satellite attitude control problem is considered with sloshing liquid fuel, which can be modelled as unknown internal dynamics. The designed controller can stabilize the attitude, while rejecting the effects of changing model parameters. In [11], the IADP algorithm is used to design a controller for an F-16 aircraft longitudinal model, with actuator faults and structural damages. In [12], the IADP algorithm is used to design an adaptive flight controller for a longitudinal airplane model, considering partial observability of measured states.

The main contributions of this paper are summarized as follows:

- The incremental value iteration is developed as a model-free approach for a fixed-wing UAV attitude tracking problem.
- The uncertainties of wing lift and drag moments, caused by the change of center of pressure, is estimated by using RLS identification algorithm.
- The kernel matrix $P$ in incremental value iteration is used to learn the optimal cost function of the flight controller, which is associated with the angle of attack tracking error and control effort.

The remainder of this paper is structured as follows. Section II introduces the incremental control method and the RLS identification algorithm, Section III develops the incremental value iteration algorithm for nonlinear optimal control problems. Section IV introduces the longitudinal attitude dynamical model of a fixed-wing UAV. In Section V, the effectiveness of the adaptive optimal flight controller of the fixed-wing UAV using incremental value iteration algorithm is numerically verified. The conclusion of this paper is provided in Section VI.

## II. Problem Formulation

### A. Incremental Control

The discrete-time nonlinear system driven by control input is presented as

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{u}_k), k \in \mathbb{N} \tag{1}$$

where $f:\mathbb{R}^n\times\mathbb{R}^m\to\mathbb{R}^n$ is a smooth nonlinear function associated with state vector $\boldsymbol{x}_k$ and input vector $\boldsymbol{u}_k$. $n,m$ are positive integers denoting the dimensions of the state and control spaces. $k$ represents the discrete-time index. $\mathbb{N}$ represents the set of nonnegative integers.

Taking the Taylor Expansion of (1) at state $\boldsymbol{x}_k$ as

$$\begin{aligned}\boldsymbol{x}_{k+1}=&\boldsymbol{x}_k+\boldsymbol{F}_{k-1}(\boldsymbol{x}_k-\boldsymbol{x}_{k-1})+\boldsymbol{G}_{k-1}(\boldsymbol{u}_k-\boldsymbol{u}_{k-1})\\&+O\left[(\boldsymbol{x}_k-\boldsymbol{x}_{k-1})^2,(\boldsymbol{u}_k-\boldsymbol{u}_{k-1})^2\right]\end{aligned}\tag{2}$$

where $\boldsymbol{F}_{k-1}=\partial f(\boldsymbol{x},\boldsymbol{u})/\partial\boldsymbol{x}|_{\boldsymbol{x}_{k-1},\boldsymbol{u}_{k-1}}\in\mathbb{R}^{n\times n}$ is the system transition matrix, and $\boldsymbol{G}_{k-1}=\partial f(\boldsymbol{x},\boldsymbol{u})/\partial\boldsymbol{u}|_{\boldsymbol{x}_{k-1},\boldsymbol{u}_{k-1}}\in\mathbb{R}^{n\times m}$ is the input distribution matrix at time step $k-1$ for discretized systems. $O\left[(\boldsymbol{x}_k-\boldsymbol{x}_{k-1})^2,(\boldsymbol{u}_k-\boldsymbol{u}_{k-1})^2\right]$ are the higher order terms of the Taylor Expansion.

Eq.(2) can be rewritten in an incremental formulation as

$$\Delta\boldsymbol{x}_{k+1}=\boldsymbol{F}_{k-1}\Delta\boldsymbol{x}_k+\boldsymbol{G}_{k-1}\Delta\boldsymbol{u}_k+O(\Delta\boldsymbol{x}_k^2,\Delta\boldsymbol{u}_k^2)\tag{3}$$

where $\Delta\boldsymbol{x}_{k+1}=\boldsymbol{x}_{k+1}-\boldsymbol{x}_k$ is the state increment at time index $k+1$ with respect to $k$. $\Delta\boldsymbol{x}_k=\boldsymbol{x}_k-\boldsymbol{x}_{k-1},\Delta\boldsymbol{u}_k=\boldsymbol{u}_k-\boldsymbol{u}_{k-1}$ are the state and control increments at time index $k$ with respect to $k-1$.

By assuming a high sample frequency, the higher-order terms $O\left[(\boldsymbol{x}_k-\boldsymbol{x}_{k-1})^2,(\boldsymbol{u}_k-\boldsymbol{u}_{k-1})^2\right]$ can be omitted [6], such that

$$\Delta\boldsymbol{x}_{k+1}\approx\boldsymbol{F}_{k-1}\Delta\boldsymbol{x}_k+\boldsymbol{G}_{k-1}\Delta\boldsymbol{u}_k\tag{4}$$

### B. Recursive Least Squares Identification

The augmented system state and system matrices are defined as

$$\begin{cases}\boldsymbol{X}_k=\begin{bmatrix}\Delta\boldsymbol{x}_k\\ \Delta\boldsymbol{u}_k\end{bmatrix}\\ \hat{\boldsymbol{\Theta}}_{k-1}=\begin{bmatrix}\hat{\boldsymbol{F}}_{k-1} & \hat{\boldsymbol{G}}_{k-1}\end{bmatrix}^T\end{cases}\tag{5}$$

where $\hat{\boldsymbol{F}}_{k-1},\hat{\boldsymbol{G}}_{k-1}$ are the approximations of $\boldsymbol{F}_{k-1},\boldsymbol{G}_{k-1}$.

The one-step prediction of $\Delta\hat{\boldsymbol{x}}_{k+1}^T$ is

$$\Delta\hat{\boldsymbol{x}}_{k+1}^T=\boldsymbol{X}_k^T\hat{\boldsymbol{\Theta}}_{k-1}\tag{6}$$

The error between $\Delta\boldsymbol{x}_{k+1}^T$ and $\Delta\hat{\boldsymbol{x}}_{k+1}^T$ is defined as

$$\boldsymbol{\varepsilon}_k=\Delta\boldsymbol{x}_{k+1}^T-\Delta\hat{\boldsymbol{x}}_{k+1}^T\tag{7}$$

The estimate of the augmented system matrices $\hat{\boldsymbol{\Theta}}_{k-1}$ is updated as

$$\hat{\boldsymbol{\Theta}}_k=\hat{\boldsymbol{\Theta}}_{k-1}+\frac{\Lambda_{k-1}\boldsymbol{X}_k}{\kappa+\boldsymbol{X}_k^T\Lambda_{k-1}\boldsymbol{X}_k}\boldsymbol{\varepsilon}_k\tag{8}$$

where $\Lambda_{k-1}$ is the equal weighted estimation of covariance matrix $\mathrm{Cov}(\hat{\boldsymbol{\Theta}}_k-\hat{\boldsymbol{\Theta}}_{k-1})$, which describes the confidence of the estimated $\hat{\boldsymbol{\Theta}}_k$. $\Lambda_{k-1}$ is updated by

$$\Lambda_k=\frac{1}{\kappa}\left[\Lambda_{k-1}-\frac{\Lambda_{k-1}\boldsymbol{X}_k\boldsymbol{X}_k^T\Lambda_{k-1}}{\kappa+\boldsymbol{X}_k^T\Lambda_{k-1}\boldsymbol{X}_k}\right]\tag{9}$$

where $\kappa\in(0,1)$ is the forgetting factor. The value of $\kappa$ provides a balance between noise rejection and time-varying parameter estimation. When $\kappa\to1$, the RLS algorithm becomes equally weighted and behaves better at noise rejection; when $\kappa\to0$, the RLS algorithm shows more adaptation to new measurements, and thus adapts better to time-varying parameters. For a satisfying performance in practice, $\kappa$ is suggested to be assigned as $0.9<\kappa<0.995$.

## III. Adaptive Optimal Control

This section develops the incremental value iteration algorithm for the optimal control problem of a nonlinear system. The approximated cost function is firstly derived which is a function associated with the control increment. As a result, the optimal control increment in an analytical form is obtained. Finally, the incremental value iteration algorithm is provided.

### A. Incremental Value Iteration

Incremental value iteration algorithm combines incremental control and value iteration to solve the HJB equation for nonlinear system infinite-time horizon optimal control problems. Incremental control utilizes RLS identification algorithm to obtain a linear system model online, which is used in value iteration to derive an optimal control solution.

The utility function is defined as

$$r(\boldsymbol{x}_k,\boldsymbol{u}_k)=(\boldsymbol{x}_k-\boldsymbol{x}_k^{\mathrm{ref}})^T\boldsymbol{Q}(\boldsymbol{x}_k-\boldsymbol{x}_k^{\mathrm{ref}})+\boldsymbol{u}_k^T\boldsymbol{R}\boldsymbol{u}_k\tag{10}$$

where $Q$ and $R$ are positive definite matrices, and $\boldsymbol{x}_k^{\mathrm{ref}}$ is the reference signal for the system state. The cost function is the cumulative sum of utility function starting from state $\boldsymbol{x}_k$ driven by a policy

$$V(k)=\sum_{l=k}^{\infty}\gamma^{l-k}r(\boldsymbol{x}_l,\boldsymbol{u}_l)\tag{11}$$

where the discount factor $\gamma\in(0,1)$ represents the importance of future utility functions.

The Bellman equation [1] is then derived as

$$V(k)=r(\boldsymbol{x}_k,\boldsymbol{u}_k)+\gamma V(k+1)\tag{12}$$

**Remark 1:** The discount factor $\gamma<1$ makes sure that the sequences of discounted future utility functions converge to 0 as $l\to\infty$, which is a finite-horizon optimal control problem. Intuitively, the future utility values do not have the same importance as near-horizon utility values. When $\gamma=1$, as in typical infinite-horizon optimal control problems, the bound of $V(k)$ goes to infinity and the stability result fails. The value of $\gamma$ affects the convergence rate of value iteration. The smaller $\gamma$ is, the faster the value iteration algorithm converges.

The reconstruction of the exact cost functions $V(k)$ and $V(k+1)$ in Eq.(12) is a challenge in approximate value iteration. For this purpose, a parameterized approximator is commonly used, at the cost of introducing an approximation

error. More details on approximation errors are discussed in Section III. This paper adopts a quadratic cost function to approximate the exact cost function as follows:

$$\hat{V}(k) = \boldsymbol{e}_k^T P \boldsymbol{e}_k \tag{13}$$

Because $\hat{V}(k+1) = \boldsymbol{e}_{k+1}^T P \boldsymbol{e}_{k+1}$, where $\boldsymbol{e}_{k+1}$ is not available at time index $k$, one has to predict $\boldsymbol{e}_{k+1}$ using the constructed linear incremental model. To this end, the exact $\boldsymbol{e}_{k+1}$ is derived as

$$\begin{aligned}\boldsymbol{e}_{k+1} &= \boldsymbol{x}_{k+1} - \boldsymbol{x}_{k+1}^{\text{ref}} \\ &= \boldsymbol{x}_k + \hat{\boldsymbol{F}}_{k+1}\Delta\boldsymbol{x}_k + \hat{\boldsymbol{G}}_{k-1}\Delta\boldsymbol{u}_k + \Delta_{IME} - \boldsymbol{x}_k^{\text{ref}} - \Delta\boldsymbol{x}_{k+1}^{\text{ref}} \\ &= (\boldsymbol{x}_k - \boldsymbol{x}_k^{\text{ref}}) + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k + \hat{\boldsymbol{G}}_{k-1}\Delta\boldsymbol{u}_k + \Delta_{IME} - \Delta\boldsymbol{x}_{k+1}^{\text{ref}} \\ &= \boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k + \hat{\boldsymbol{G}}_{k-1}\Delta\boldsymbol{u}_k + \Delta_{IME} - \Delta\boldsymbol{x}_{k+1}^{\text{ref}}\end{aligned} \tag{14}$$

where $\Delta_{IME} = (\boldsymbol{F}_{k-1} - \hat{\boldsymbol{F}}_{k-1})\Delta\boldsymbol{x}_k + (\boldsymbol{G}_{k-1} - \hat{\boldsymbol{G}}_{k-1})\Delta\boldsymbol{u}_k + O(\Delta\boldsymbol{x}_k^2, \Delta\boldsymbol{u}_k^2)$ is the total error of using incremental model approximation and RLS estimation.

Omitting $\Delta_{IME}$ and the increment of reference signal $\Delta\boldsymbol{x}_{k+1}^{\text{ref}}$, the prediction of $\boldsymbol{e}_{k+1}$ is calculated as

$$\hat{\boldsymbol{e}}_{k+1} = \boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k + \hat{\boldsymbol{G}}_{k-1}\Delta\boldsymbol{u}_k \tag{15}$$

Then, the modified approximated cost function, denoted as $\hat{V}'(k+1)$, is defined as

$$\hat{V}'(k+1) = \hat{\boldsymbol{e}}_{k+1}^T P \hat{\boldsymbol{e}}_{k+1} \tag{16}$$

Using $\hat{V}'(k+1)$ in Eq.(16) to construct exact $V(k+1)$ in Eq.(12) , one has

$$\begin{aligned}\hat{V}(k) &\approx r(\boldsymbol{x}_k, \boldsymbol{u}_k) + \gamma\hat{V}'(\boldsymbol{e}_{k+1}) \\ &= \boldsymbol{e}_k^T Q \boldsymbol{e}_k + \boldsymbol{u}_k^T R \boldsymbol{u}_k + \gamma\hat{\boldsymbol{e}}_{k+1}^T P \hat{\boldsymbol{e}}_{k+1} \\ &= \boldsymbol{e}_k^T Q \boldsymbol{e}_k + (\boldsymbol{u}_{k-1} + \Delta\boldsymbol{u}_k)^T R(\boldsymbol{u}_{k-1} + \Delta\boldsymbol{u}_k) \\ &\quad + \gamma(\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k + \hat{\boldsymbol{G}}_{k-1}\Delta\boldsymbol{u}_k)^T P(\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k + \hat{\boldsymbol{G}}_{k-1}\Delta\boldsymbol{u}_k)\end{aligned} \tag{17}$$

**Remark 2:** In Eq.(17), the quadratic function $\hat{\boldsymbol{e}}_{k+1}^T P \hat{\boldsymbol{e}}_{k+1}$ is used to construct the exact cost function $V(k+1)$, i.e.$\hat{V}'(k+1) = \hat{\boldsymbol{e}}_{k+1}^T P \hat{\boldsymbol{e}}_{k+1}$. This approximation can be divided into two parts: the first part is using $\boldsymbol{e}_{k+1}^T P \boldsymbol{e}_{k+1}$ to approximate $V(k+1) = \sum_{l=k+1}^{\infty} \gamma^{l-k} r(\boldsymbol{x}_l, \boldsymbol{u}_l)$, the second part is using $\hat{\boldsymbol{e}}_{k+1}$ to approximate $\boldsymbol{e}_{k+1}$. The approximation error of $\hat{V}(k)$ decreases at each step of value iteration. The approximation error of $\hat{\boldsymbol{e}}_{k+1}$ depends on the RLS identification algorithm.

The optimal approximated cost function $\hat{V}^*(k)$ is defined as

$$\hat{V}^*(k) = \min_{\Delta\boldsymbol{u}_k} \left[\boldsymbol{e}_k^T Q \boldsymbol{e}_k + (\boldsymbol{u}_{k-1} + \Delta\boldsymbol{u}_k)^T R(\boldsymbol{u}_{k-1} + \Delta\boldsymbol{u}_k) + \gamma\hat{V}^*(k+1)\right] \tag{18}$$

The optimal control increment $\Delta\boldsymbol{u}^*(k)$ is given as

$$\Delta\boldsymbol{u}^*(k) = \underset{\Delta\boldsymbol{u}_k}{\text{argmin}} \left[\boldsymbol{e}_k^T Q \boldsymbol{e}_k + (\boldsymbol{u}_{k-1} + \Delta\boldsymbol{u}_k)^T R(\boldsymbol{u}_{k-1} + \Delta\boldsymbol{u}_k) + \gamma\hat{V}^*(k+1)\right] \tag{19}$$

The best estimate for the optimal control increment is given by $\partial\hat{V}(k)/\partial(\Delta\boldsymbol{u}_k) = 0$

$$\begin{aligned}\frac{\partial\hat{V}(k)}{\partial\Delta\boldsymbol{u}_k} &\approx 2R(\boldsymbol{u}_{k-1} + \Delta\boldsymbol{u}_k) + 2\gamma\hat{\boldsymbol{G}}_{k-1}^T P(\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k + \hat{\boldsymbol{G}}_{k-1}\Delta\boldsymbol{u}_k) \\ &= 2(R + \gamma\hat{\boldsymbol{G}}_{k-1}^T P\hat{\boldsymbol{G}}_{k-1})\Delta\boldsymbol{u}_k + 2\left[R\boldsymbol{u}_{k-1} + \gamma\hat{\boldsymbol{G}}_{k-1}^T P(\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k)\right] \\ &= 0\end{aligned} \tag{20}$$

From Eq.(20), one has

$$\begin{aligned}2(R + \gamma\hat{\boldsymbol{G}}_{k-1}^T P\hat{\boldsymbol{G}}_{k-1})\Delta\boldsymbol{u}_k + 2\left[R\boldsymbol{u}_{k-1} + \gamma\hat{\boldsymbol{G}}_{k-1}^T P(\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k)\right] &= 0 \\ (R + \gamma\hat{\boldsymbol{G}}_{k-1}^T P\hat{\boldsymbol{G}}_{k-1})\Delta\boldsymbol{u}_k = R\boldsymbol{u}_{k-1} + \gamma\hat{\boldsymbol{G}}_{k-1}^T P(\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k) \\ \Delta\boldsymbol{u}_k = (R + \gamma\hat{\boldsymbol{G}}_{k-1}^T P\hat{\boldsymbol{G}}_{k-1})^{-1}\left[R\boldsymbol{u}_{k-1} + \gamma\hat{\boldsymbol{G}}_{k-1}^T P(\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k)\right]\end{aligned} \tag{21}$$

Therefore, the optimal incremental control $\Delta\boldsymbol{u}_k^*$ is given as

$$\Delta\boldsymbol{u}_k^* = (R + \gamma\hat{\boldsymbol{G}}_{k-1}^T P\hat{\boldsymbol{G}}_{k-1})^{-1}\left[R\boldsymbol{u}_{k-1} + \gamma\hat{\boldsymbol{G}}_{k-1}^T P(\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k)\right] \tag{22}$$

**a.Policy Improvement.** The policy improves for the current kernel matrix $P^i$:

$$\begin{aligned}\Delta\boldsymbol{u}_k^i &= -(R + \gamma\hat{\boldsymbol{G}}_{k-1}^T P^i\hat{\boldsymbol{G}}_{k-1})^{-1}\left[R\boldsymbol{u}_{k-1} + \gamma\hat{\boldsymbol{G}}_{k-1}^T P^i(\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1}\Delta\boldsymbol{x}_k)\right] \\ \boldsymbol{u}_k^i &= \boldsymbol{u}_{k-1} + \Delta\boldsymbol{u}_k^i\end{aligned} \tag{23}$$

**b.Policy Evaluation.** The cost function kernel matrix $P$ can be evaluated and updated recursively with the Bellman equation for each iteration $i = 0, 1, \ldots$ until convergence:

$$\boldsymbol{e}_k^T P^{i+1}\boldsymbol{e}_k \approx \boldsymbol{e}_k^T Q\boldsymbol{e}_k + (\boldsymbol{u}_k^i)^T R\boldsymbol{u}_k^i + \gamma\hat{\boldsymbol{e}}_{k+1}^T P^i\hat{\boldsymbol{e}}_{k+1} \tag{24}$$

**Remark 3:** The optimality of the control policy in Eq.(23) is partially achieved by adopting a changing kernel matrix $P^i$. Using Eq.(24) improves the precision of value function approximation, resulting into an improved matrix $P^{i+1}$, which makes $\hat{V}(\boldsymbol{e}_k)$ closer to $V(\boldsymbol{e}_k)$. Therefore, the control derived by $\hat{V}(\boldsymbol{e}_k)$ in Eq.(20) is closer to the optimal control derived by $V(\boldsymbol{e}_k)$. Meanwhile, there exists various approximation errors between exact value iteration and incremental value iteration.

## IV. Fixed-wing UAV Dynamical Model

### A. Longitudinal attitude dynamics

The longitudinal dynamical model is given as [13]

$$\begin{aligned}I\ddot{\theta} = &-F_f d_f + d_{\text{cp}} L_w \cos(\theta - \gamma_w) + d_{\text{cp}} D_w \sin(\theta - \gamma_w) \\ &- d_s L_s \cos(\theta - \gamma_s) - d_s D_s \sin(\theta - \gamma_s)\end{aligned} \tag{25}$$

where $I$ is the moment of inertia about center of gravity, $\theta$ is aircraft pitch angle, $F_f$ is fuselage aerodynamic force, $d_{\text{cp}}$ is center of pressure moment arm, $L_s, D_s$ are stabilator lift and drag forces. $d_f$ is fuselage moment arm, $d_s$ is stabilator moment arm. $\gamma_w$ is angle made by wing velocity and wing horizontal velocity, $\gamma_s$ is angle made by stabilator velocity and stabilator horizontal velocity. $\gamma_w, \gamma_s$ are defined as

$$\gamma_w = \arctan\frac{\dot{y}_w}{\dot{x}_w} \tag{26}$$

**Algorithm 1:** Incremental Value Iteration Algorithm

**Required Input:**
state $\boldsymbol{x}_k, \boldsymbol{x}_{k+1}$, state reference $\boldsymbol{x}_k^{\text{ref}}, \boldsymbol{x}_{k+1}^{\text{ref}}$
**Initialization:**
Choose maximum iteration number $i_{max}$
Choose discount factor $\gamma$, cost function matrices $Q, R$
Choose initial kernel matrix $P^0$, initial control $u_0$
Choose initial system matrices $\hat{\Theta}_0 = [\hat{F}_0, \hat{G}_0]^T$
Choose initial covariance matrix $\Lambda_0$, forgetting factor $\kappa$
**RLS Identification:**
1: $\Delta \hat{\boldsymbol{x}}_{k+1}^T = X_k^T \hat{\Theta}_{k-1}$
2: $\Delta \boldsymbol{x}_{k+1} = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$
3: $\boldsymbol{\varepsilon}_k = \Delta \boldsymbol{x}_{k+1}^T - \Delta \hat{\boldsymbol{x}}_{k+1}^T$
4: $\hat{\Theta}_k = \hat{\Theta}_{k-1} + \frac{\Lambda_{k-1} \boldsymbol{X}_k}{\kappa + \boldsymbol{X}_k^T \Lambda_{k-1} \boldsymbol{X}_k} \boldsymbol{\varepsilon}_k$
5: $\Lambda_k = \frac{1}{\kappa} \left[ \Lambda_{k-1} - \frac{\Lambda_{k-1} \boldsymbol{X}_k \boldsymbol{X}_k^T \Lambda_{k-1}}{\kappa + \boldsymbol{X}_k^T \Lambda_{k-1} \boldsymbol{X}_k} \right]$
**Value Iteration:**
**for** $i = 0$ to $i_{max}$
1: $\boldsymbol{e}_k \leftarrow \boldsymbol{x}_k - \boldsymbol{x}_k^{\text{ref}}$
2: $\boldsymbol{e}_{k+1} \leftarrow \boldsymbol{x}_{k+1} - \boldsymbol{x}_{k+1}^{\text{ref}}$
3: $\Delta \boldsymbol{u}_k^i \leftarrow -(R + \gamma \hat{\boldsymbol{G}}_{k-1}^T P^i \hat{\boldsymbol{G}}_{k-1})^{-1} \left[ R \boldsymbol{u}_{k-1} + \gamma \hat{\boldsymbol{G}}_{k-1}^T P^i (\boldsymbol{e}_k + \hat{\boldsymbol{F}}_{k-1} \Delta \boldsymbol{x}_k) \right]$
4: $\boldsymbol{u}_k^i \leftarrow \boldsymbol{u}_{k-1} + \Delta \boldsymbol{u}_k^i$
5: Solve $\boldsymbol{e}_k^T P^{i+1} \boldsymbol{e}_k = \boldsymbol{e}_k^T Q \boldsymbol{e}_k + (\boldsymbol{u}_k^i)^T R \boldsymbol{u}_k^i + \gamma \hat{\boldsymbol{e}}_{k+1}^T P^i \hat{\boldsymbol{e}}_{k+1}$, obtain $P^{i+1}$
**end for**

$$\gamma_s = \arctan \frac{\dot{y}_s}{\dot{x}_s} \tag{27}$$

and $(x_w, y_w)$ and $(x_s, y_s)$ are positions of wing and stabilator surface area centroids in a longitudinal body axis [13]:

$$(x_w, y_w) = (x + d_w \cos\theta, y + d_w \sin\theta) \tag{28}$$

$$(x_s, y_s) = (x - d_s \cos\theta, y - d_s \sin\theta) \tag{29}$$

The angles of attack of the wing ($\alpha_w$), the stabilator ($\alpha_s$), and the fuselage ($\alpha_f$) are:

$$\alpha_w = \theta - \gamma_w + \alpha_i \tag{30}$$

$$\alpha_s = \theta - \gamma_s + \phi \tag{31}$$

$$\alpha_f = \theta - \arctan(\frac{\dot{y}_f}{\dot{x}_f}) \tag{32}$$

where $\alpha_i$ is the wing incidence angle measured relative to the longitudinal body axis, $\phi$ is the stabilator deflection, and $(x_f, y_f)$ is the position of the fuselage centroid.

### B. Wing aerodynamics

The wing lift force $L_w$ is calculated as [14]

$$L_w = \frac{1}{2} C_{L_w} \rho (\dot{x}_w^2 + \dot{y}_w^2) S_w c_w \tag{33}$$

where $C_{L_w}$ is assumed to be linear:

$$C_{L_w} = a \alpha_w + C_{L_{a=0}} \tag{34}$$

where $a$ is given by Prandtl's lifting-line theory [14]:

$$a = \frac{a_0}{1 + a_0/(\pi e_w A R_w)} \tag{35}$$

and $a_0$ is the prestall lift coefficient slope for the airfoil. According to Ref. [13], the value $C_{L_{a=0}}$ is set to be 0.4.

The drag on the perching UAV wing, denoted as $D_w$ is presented as

$$D_w = \frac{1}{2} C_{D_w} \rho (\dot{x}_w^2 + \dot{y}_w^2) S_w c_w \tag{36}$$

where $C_{D_w}$ is a dimensional aerodynamic coefficient composed of profile drag and induced drag components for angles of attack between zero and the onset of stall [14]:

$$C_{D_w} = c_{d_w} + \frac{C_{L_w}^2}{\pi e_w A R_w} \tag{37}$$

Profile drag $c_{d_w}$ is assumed to be constant at 0.05. $C_{D_w}$ is modeled as a line extending to 1.2 at 15 deg angle of attack [13]. Parameters $e_w, AR_w$ are given in Table I. $C_{L_w}$ is calculated as in [13].

### C. Stabilator and fuselage aerodynamics

The lift and drag on the stabilator, $L_s, D_s$ are expressed in terms of dimensionless aerodynamic lift and drag coefficients, i.e., $C_{L_s}$ and $C_{D_s}$ [14]:

$$L_s = \frac{1}{2} C_{L_s} \rho (\dot{x}_s^2 + \dot{y}_s^2) S_s c_s \tag{38}$$

$$D_s = \frac{1}{2} C_{D_s} \rho (\dot{x}_s^2 + \dot{y}_s^2) S_s c_s \tag{39}$$

where $C_{L_s}$ is assumed to be a linear function of stabilator angle of attack, as follow

$$C_{L_s} = \eta \alpha_s \tag{40}$$

where $\eta$ is found using the classic equation based on Prandtl's lifting-line theory:

$$\eta = \frac{\eta_0}{1 + \eta_0/(\pi e_s A R_s)} \tag{41}$$

where $\eta_0$ is the lift coefficient slope in the prestall region for the NACA 0009 airfoil. $C_{D_s}$ is the drag coefficient composed of airfoil profile drag and induced drag components, as follow:

$$C_{D_s} = c_{d_s} + \frac{C_{L_s}^2}{\pi e_s A R_s} \tag{42}$$

where $c_d$ is assumed to be constant at 0.010 [15].

The aerodynamic force is calculated as

$$F_f = \frac{1.2}{\pi} \alpha_f \rho (\dot{x}_d^2 + \dot{y}_f^2) DL \tag{43}$$

where it is assumed that $F_f$ is linear with respect to angle of attac between 0 and 15 deg angle of attack [16].

## V. Numerical Simulation

The aircraft simulation model is trimmed such that its center of gravity is in the longitudinal plane during the flight. The reference $\alpha_f^{\text{ref}}$ is set to be $\alpha_f^{\text{ref}} = \frac{\pi}{12}\sin(0.4\pi t)$. In order to excite the RLS identification process, a persistent excitation signal is added into the control input as $n_k = 0.3e^{-t}[\sin(-20t)+\sin(10t)+\cos(30t)]$. The physical coefficients of the UAV arer provided in Table I.

TABLE I
Aircraft Physical Coefficients

| Parameter | Value |
|---|---|
| Mass $m$ | 0.88kg |
| Moment of inertia $I$ | 0.30 |
| Wingspan $S_w$ | 1.41m |
| Wing chord $c_w$ | 0.21m |
| Wing centroid to center of gravity | 0.29m |
| Wing aspect ratio $AR_w$ | 6.7 |
| Stabilator span $S_s$ | 0.40m |
| Stabilator chord $c_s$ | 0.11m |
| Stabilator aspect ratio $AR_s$ | 3.6 |
| Stabilator moment arm $d_s$ | 0.49m |
| Fuselage length $L$ | 0.38m |
| Fuselage diameter $D$ | 0.064m |
| Fuselage moment arm $d_f$ | 0.53m |
| Density of air $\rho$ | 1.20kg/m$^3$ |

### A. Robustness to Initial Values of $\alpha_0$

The robustness of the adaptive flight controller to different initial values of the angle of attack $\alpha_{f0}$ is investigated, in order to demonstrate the its performance in different real flight conditions. Specifically, the initial values are set to be $\alpha_{f0} = -10°, -5°, 0°, 5°, 10°$. Figures 1,2 show that the flight controller can track the reference signal with different initial $\alpha_{f0}$ in less than 2.5s, while keeping the pitch rate $\dot{\theta}$ stable. The tracking errors are large before 4s because the persistent excitation signal disturbs the control input. Although the PE signal $n_k$ is added into the control input, figures 1,2 show that it does not affect the tracking performance.

### B. Kernel Matrix P

The kernel matrix $P$ indicates the performance of the incremental value iteration algorithm. $P$ can learn the original nonlinear cost function and thus guide the controller to maximize the cost function. As shown in Figure 3, the elements $P_{11}, P_{12}, P_{21}, P_{22}$ at a fixed time $t = 3$s converge to the optimal values after 300 value iterations.

### C. Identification of System Matrix and Control Matrix

The online identification results using the RLS identification algorithm from Section II-B are shown in Figures 4, 5. $\hat{F}$ is the identified system matrix of the incremental model and $\hat{G}$ is the identified control matrix of the incremental model. Figures 4, 5 show that the estimated system matrix and control matrix converge in less than 0.3s. As a result, the derived control policy in Eq.(23) uses the estimation $\hat{F}, \hat{G}$ to calculate the optimal control command. The convergence

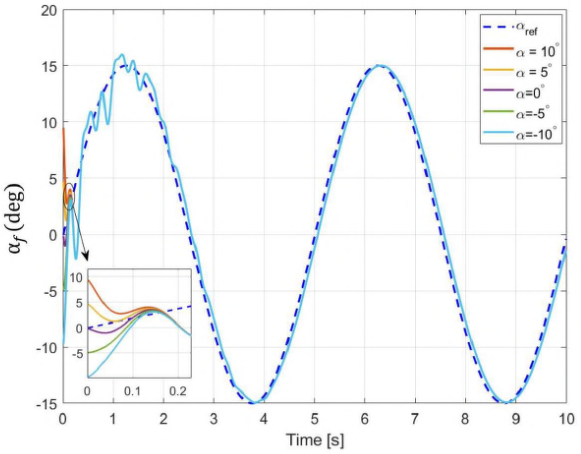


Fig. 1. Angle of Attack $\alpha_f$ Tracking.

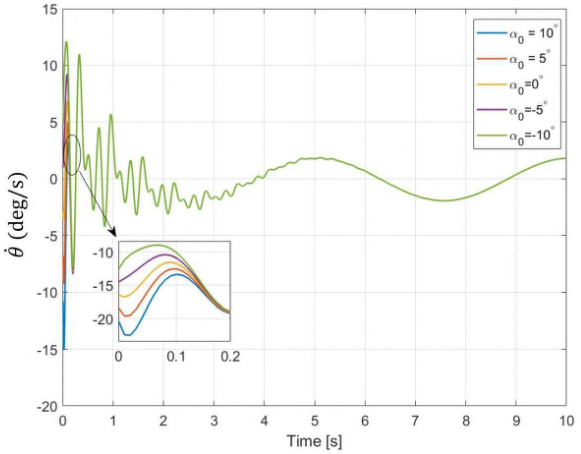


Fig. 2. Pitch Rate $\dot{\theta}$.

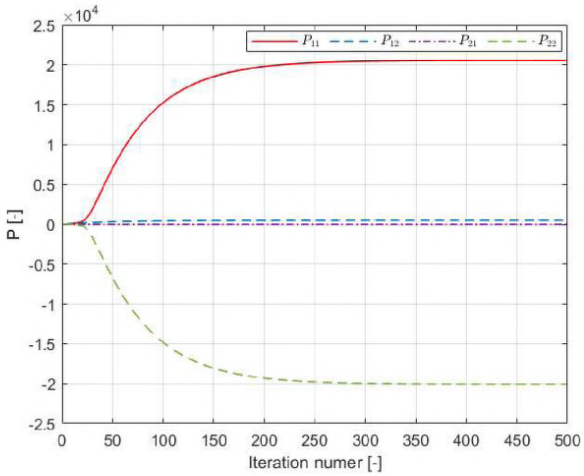


Fig. 3. Kernel Matrix $P$ at $t = 3$s.

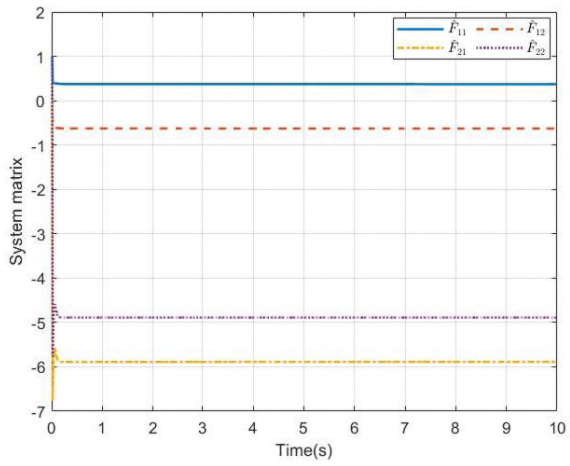


Fig. 4. Identified System Matrix $\hat{F}$.

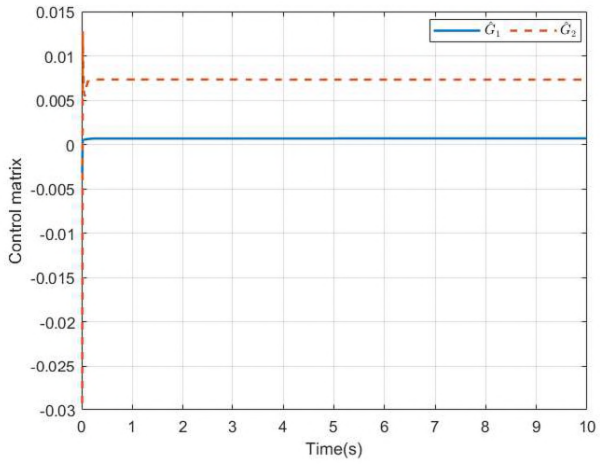


Fig. 5. Identified Control Matrix $\hat{G}$.

responses demonstrate the fast estimation performance of RLS identification algorithm.

## VI. CONCLUSIONS

The incremental value iteration is introduced to design an adaptive flight controller for fixed-wing unmanned aerial vehicles. A detailed derivation of the incremental value iteration algorithm is provided, leading to an approximated optimal controller in analytical form. Simulation results on a UAV longitudinal flight control problem verify that the designed optimal controller can track the angle of attack reference signal, while keeping the optimality of the cost function.

## REFERENCES


[1] R.S.Sutton, A.G.Barto, Reinforcement learning: An Introduction, Cambridge, Massachussetts, USA: The MIT Press, 1998..

[2] A,Al-Tamimi, F.L.Lewis, M.Abu-Khalaf, "Discrete-time Nonlinear HJB Solution Using Approximate Dynamic Programming: Convergence Proof," IEEE Transactions on System, Man and Cybernetics, 38(4),943-949,2008.


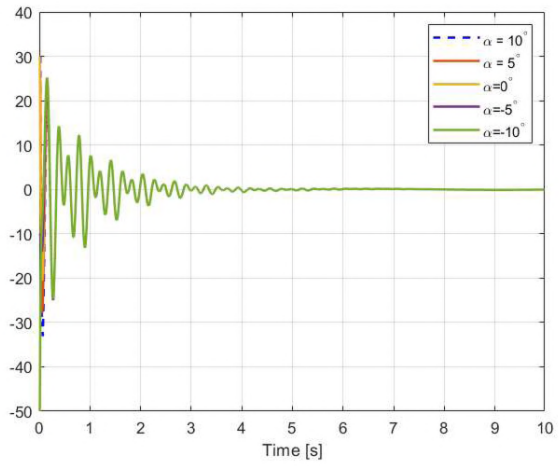


Fig. 6. Elevator Deflection.


[3] D.Liu, Q.Wei, "Policy Iteration Adaptive Dynamic Programming Algorithm for Discrete-time Nonlinear Systems," IEEE Transactions on Neural Networks and Learning Systems, 25(3), 621-634, 2014.

[4] Y.Zhou, E.van Kampen, Q.Chu, "Incremental Approximate Dynamic Programming for Nonlinear Adaptive Tracking Control with Partial Observability," Journal of Guidance, Control and Dynamics, 41(12), 2554-2567, 2018.

[5] A.Heydari, "Theoretical and Numerical Analysis of Approximate Dynamic Programming with Approximation Errors," Journal of Guidance, Control and Dynamics, 39(2), 301-311, 2016.

[6] S.Sieberling, "Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration prediction," Journal of Guidance, Control and Dynamics, 33(6), 1732-1742, 2010.

[7] P.Acquatella, W.Falkena, E.van Kampen and Q.Chu, "Robust Nonlinear Spacecraft Attitude Control Using Incremental Nonlinear Dynamic Inversion," AIAA Guidance, Navigation and Control Conference, 2012-4623, 2012.

[8] P.Acquatella, E.van Kampen, Q.Chu, "Incremental Backstepping for Robust Nonlinear Flight Control," Proceedings of the EuroGNC 2013: 2nd CEAS Specialist Conference on Guidance, Navigation and Control, CEAS, 2013, 1444-1463.

[9] X.Wang, T.Mkhoyan, R.Breuker, "Incremental Sliding-mode Fault-tolerant Flight Control," Journal of Guidance, Control and Dynamics, 42(2), 244-259, 2019.

[10] Y.Zhou, E.van Kampen, Q.Chu, "Nonlinear Adaptive Flight Control Using Incremental Approximate Dynamic Programming and Output Feedback," Journal of Guidance, Control and Dynamics, 40(2), 493-500, 2017.

[11] B.Sun, E.van.Kampen, "Incremental Model-based Global Dual Heuristic Programming With Explicit Analytical Calculations Applied to Flight Control," Engineering Applications of Artificial Intelligence, 89, 1-11, 2020.

[12] B.Sun, E.van.Kampen, "Intelligent Adaptive Optimal Control Using Incremental Model-based Global Dual Heuristic Programming Subject to Partial Observability," Applied Soft Computing, 103, 1-15, 2021.

[13] M.Puopolo, J.D.Jacob, "Model for Longitudinal Perch Maneuvers of a Fixed-wing Unmanned Aerial Vehicle," Journal of Aircraft, 52(6), 1-11, 2015.

[14] S.A.Brandt, R.J.Stiles, J.J.Bertin and R.Whitford, Introduction to Aeronautics: A Design Perspective. AIAA Reston, VA, 2004, pp.118-123, Chap.4.

[15] E.N.Jacobs, K.E.Ward, and R.M.Pinkerton, The characteristics of 78 related airfoil sections from tests in the variable-density wind tunnel. NACA Rept. 460, 1933.

[16] D.J.Tritton. Physical Fluid Dynamics, 2nd ed., Oxford University Press, New York, 1988, pp.32-33, Chapter 3.