

Research Article

A Two-Stage Method for UCAV TF/TA Path Planning Based on Approximate Dynamic Programming

Hao-xiang Chen , Ying Nan , and Yi Yang 

Nanjing University of Aeronautics and Astronautics, Nanjing 20016, China

Correspondence should be addressed to Hao-xiang Chen; chenhaoxiang@nuaa.edu.cn

Received 21 February 2018; Revised 16 August 2018; Accepted 27 August 2018; Published 5 September 2018

Academic Editor: Mauro Gaggero

Copyright © 2018 Hao-xiang Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We present a two-stage method for solving the terrain following (TF)/terrain avoidance (TA) path-planning problem for unmanned combat air vehicles (UCAVs). The 1st stage of planning takes an optimization approach for generating a 2D path on a horizontal plane with no collision with the terrain. In the 2nd stage of planning, an optimal control approach is adopted to generate a 3D flyable path for the UCAV that is as close as possible to the terrain. An approximate dynamic programming (ADP) algorithm is used to solve the optimal control problem in the 2nd stage by training an action network to approximate the optimal solution and training a critical network to approximate the value function. Numerical simulations indicate that ADP can determine the optimal control variables for UCAVs; relative to the conventional optimization method, the optimal control approach with ADP shows a better performance under the same conditions.

1. Introduction

Unmanned combat air vehicles (UCAVs) constitute an experimental class of unmanned aerial vehicle. Since the U.S. Air Force and Department of Defense developed the first UCAV X245A, UCAVs have been a means for precise air strikes worldwide in modern warfare. Relative to manned fighters, UCAVs have potential strong advantages such as less risk to personnel safety, lighter weight due to the lack of need to carry equipment for a human pilot and greater payloads, range, and maneuverability [1]. However, it is necessary to plan a path for a UCAV on a battlefield with complex terrain conditions. This process has received considerable research attention because following the planned path maximizes the likelihood of mission success. The concept of terrain following (TF)/terrain avoidance (TA) path planning is presented to solve the path-planning problem over complex terrain. The fundamental purpose of TF/TA path planning is to calculate an optimal or approximately optimal route for a UCAV that enables the UCAV to penetrate the terrain without collisions and avoid enemy radar detection by taking advantage of topographic factors around the UCAV. Conventional approaches to solving this problem are divided into

two categories: optimization approaches and optimal control approaches. Optimization approaches typically formulate this problem as a matter of numerical cost minimization to be solved using algorithms such as genetic algorithms (GAs) [2], particle swarm optimization [2], and A^* search [3] and rapid-exploration random tree (RRT) [4] algorithms. A^* and RRTs are geometric variants of the shortest path algorithm, but these approaches encounter challenges associated with path constraints. Moreover, all of the aforementioned algorithms are strongly dependent on the cost function, and during processing, their cost function should be updated and restored, which is a time-consuming operation. Due to the slow execution of optimization approaches, researchers have developed the optimal control approach by formulating the optimal path-planning problem as a two-point boundary value problem. Pure optimal control approaches can accommodate flight constraints but are problematic due to the challenges posed by complex dynamics to obtain the nonlinear adjoint equations. Moreover, the solutions are sensitive to the initial values of the adjoint variables in the corresponding iterative procedures. Therefore, some researchers have investigated alternative approaches to solve the optimal control problem. For example, Chen et al. [5] introduced a GA-based approach

for 2D path planning with a chromosome of variable length, although this approach cannot be applied to 3D TF/TA missions, and Samar et al. [6] proposed integrated guidance and control for flight height, which can accommodate TF missions while considering robustness, but the solution is not optimal. In addition, some authors have used dynamic programming (DP) to solve the optimal control problem by calculating the control variables based on a path comprising discrete segments, which has provided optimal solutions with relatively strong robustness [1]; however, the large scales of intermediate data restoration and processing—the so called “curse of dimensionality”—has limited the calculation efficiency. In recent years, researchers have made improvements to the original DP, and one of the representative examples is approximate dynamic programming (ADP) [7]. ADP, as an effective intelligent control method, solves the curse of dimensionality using an approximate function to replace the actual cost function in DP and has been applied to optimal control problems in path planning [8].

According to the advantages of the optimization approach and optimal control approach, a novel two-stage planning method is proposed here that decomposes the 3D TF/TA path-planning problem into two subproblems: a 2D path optimization subproblem with the purpose of terrain (threat) avoidance and a 3D optimal control subproblem based on a pre-planned collision-free path. Therefore, the 1st stage of planning employs the DP algorithm to generate a 2D path on the horizontal plane with no collision with the terrain. In the 2nd stage of planning, ADP is adopted to generate a 3D flyable path for the UCAV as close as possible to the terrain. Numerical simulations of path planning in a real terrain environment are also presented to verify the accuracy and efficiency of the proposed method. An analysis of a comparison with the conventional method indicates the superiority of the proposed method in certain respects.

2. Problem Definition

Path planning for a fixed-wing UCAV in an actual battle environment with complex terrain conditions requires a novel low-altitude penetration technology for achieving TF and TA. In previous research, the TF/TA path-planning problem has been commonly addressed as an optimization problem with numerous performance functions (PFs) and flight restrictions, and the strategy of the planner and the characteristics of planned path are therefore implicit in these functions and restrictions. In addition, some researchers choose the optimal control approach to solve this problem by generating a time-based flyable path. Here, we present a novel two-stage path-planning methodology to achieve TF/TA path planning in 3D space, which decomposes the optimization problem into two stages: the 1st stage focuses on the optimization problem of planning a grid-based collision-free path to avoid non-flyover terrain, and the 2nd stage takes the optimal control approach to TF flight on the basis of the roughly planned path from the 1st stage.

3. The 1st Stage of TA Path Planning

Generally, terrain information extracted from a geographic information system (GIS) is provided in the form of sample points on a gridded digital map, and values between two grid points can be obtained by interpolation. The topographic height T of each sample point (x_T, z_T) is defined as

$$T = \text{Terrain}(x_T, z_T). \quad (1)$$

Through this process, the entire topographic space for path planning is rasterized into grids with the same spacing, and the flight path is discretized as a set of path points determined by the grid lattices. The starting point and the final point are designated P_0 and P_f , respectively. Denote this set of path points as a sequence $\{P_0, P_1, P_2 \cdots P_M, P_f\}$, where $P_i = (x_{T_i}, z_{T_i}, T_i)$, and then connect adjacent points from beginning to end. Therefore, the path can be described as

$$\text{Path}_{1s} = \{P_0, P_1, P_2 \cdots P_M, P_f\}. \quad (2)$$

The entire flight path is divided into $M + 1$ segments; the i th segment is designated $L_i = \{P_{i-1}, P_i\}$, $i = 1 \cdots M + 1$; and each segment L_i should never cross terrain higher than the UCAV maximum altitude. In the actual path-planning process, the length of each segment shall never be more than twice the sampling distance of the GIS (Δ_s), which is expressed as $\Delta_s \leq \|L_i\| \leq 4\Delta_s$.

For each segment, the rate of climb (ROC) is given as

$$K_i = \frac{T_i - T_{i-1}}{\|L_i\|}, \quad (3)$$

where the segment length is $\|L_i\| = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}$. Each candidate point should satisfy the constraint $K_{\min} \leq K_i \leq K_{\max}$, and K_{\min} , K_{\max} are converted by the flight characteristics of the UCAV. In the same way, the rate of deviation in the horizontal plane can be given as

$$Q_i = \frac{|z_i - z_{i-1}|}{x_i - x_{i-1}}, \quad (4)$$

$$0 \leq Q_i \leq Q_{\max} \quad (x_i - x_{i-1} > 0)$$

$$-Q_{\max} \leq Q_i \leq 0 \quad (x_i - x_{i-1} < 0),$$

where Q_{\max} is also determined by the flight characteristics of the UCAV.

During the process of path planning, it is expected that the UCAV will fly close to some part of the protruding terrain, which means that the UCAV will take advantage of topographic factors to escape detection. Thus, the height difference of the terrain points around the UCAV is proposed to measure that advantage such that the advantage for the path is larger than for other paths for larger height differences. Taking δ points (commonly $\delta = 15$) on each side

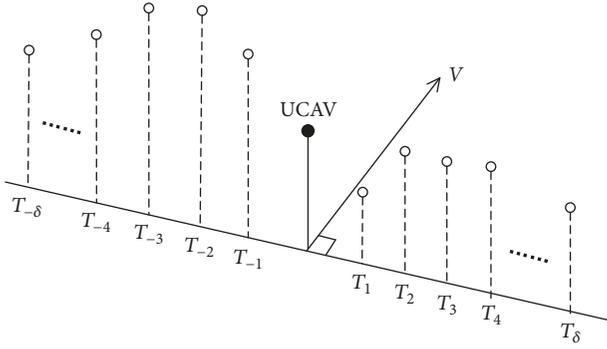


FIGURE 1: Terrain points on each side perpendicular to the speed.

perpendicular to the velocity vector as shown in Figure 1, the height difference can be given as the following:

$$T_j = T(x_U + j \cdot \Delta_s, z_U + j \cdot \Delta_s) \quad j = 1, 2, \dots, \delta \quad (5)$$

$$F_D = \sum_{j=1}^{\delta} |T_j - T_{-j}|; \quad \delta \in \mathbb{Z}. \quad (6)$$

In actual calculation, in order to facilitate the calculation and comparison of different path, the PFs of the planned path comprise normalized flight range indicator, terrain height indicator, ROC indicator, TA (no-collision) indicator, and terrain differences indicator, which can be expressed as follows:

(1) Flight range indicator is

$$\bar{J}_L = \sum \left(\frac{\|L_i\|}{L_{\max}} \right) \quad (7)$$

(2) Terrain height indicator is

$$\bar{J}_H = \sum \left(\frac{T_i}{H_{\max}} \right) \quad (8)$$

(3) ROC indicator is

$$\bar{J}_K = \sum \left(\frac{|K_i|}{K_{\max}} \right) = \sum \frac{|T_i - T_{i-1}|}{\|L_i\| \cdot K_{\max}} \quad (9)$$

(4) TA indicator is

$$\bar{J}_C = \sum Col_i$$

$$Col_i = \begin{cases} 0 & \text{no-collision} \\ 1 & \text{collision.} \end{cases} \quad (10)$$

(5) Terrain differences indicator is

$$\bar{J}_D = \sum \left(\frac{F_D}{F_{D\max}} \right) = \sum \sum_{j=1}^{\delta} \frac{-|T_j - T_{-j}|}{F_{D\max}} \quad (11)$$

$$F_{D\max} = \max_{m \in [-\delta, \delta]} (|T_m - T_{-m}|)$$

$L_{\max}, H_{\max}, K_{\max}$ are preset values determined by the flight characteristics of the UCAV. To summarize, the complete PF of the planned path from the 1st stage is written as follows:

$$\begin{aligned} C_{1s} &= \alpha_1 \bar{J}_L + \alpha_2 \bar{J}_H + \alpha_3 \bar{J}_K + \alpha_4 \bar{J}_C + \alpha_5 \bar{J}_D \\ &= \alpha_1 \sum_{Path_{1s}} \frac{\|L_i\|}{L_{\max}} + \alpha_2 \sum_{Path_{1s}} \frac{T_i}{H_{\max}} \\ &\quad + \alpha_3 \sum_{Path_{1s}} \frac{|T_i - T_{i-1}|}{\|L_i\| \cdot K_{\max}} + \alpha_4 \sum_{Path_{1s}} Col_i \\ &\quad + \alpha_5 \sum_{Path_{1s}} \sum_{j=1}^{\delta} \frac{-|T_j - T_{-j}|}{F_{D\max}} \end{aligned} \quad (12)$$

where $\alpha_1 \sim \alpha_5$ are the weight coefficients. In this form of cost function, various coefficient combinations reflect the strategy of the planner in this stage. As J_H and J_K are related to the vertical and horizontal the choice of planned waypoints, the values of α_2 and α_3 determine whether the UCAV is favored to fly over or around convex topography.

The optimization problem of the 1st stage in the 3D physical space \mathfrak{R} with terrain \mathfrak{T} is summarized as

$$\begin{aligned} Path_{1s} &= \{P_0, P_1, P_2 \cdots P_M, P_f\} \\ \{P_0, P_1, P_2 \cdots P_N\} &= \arg \min_{Path_{1s}} C_{1s} \\ \Delta_s &\leq \|L_i\| \leq 4\Delta_s \\ K_{\min} &\leq K_i \leq K_{\max} \\ 0 &\leq Q_i \leq Q_{\max} \quad (x_i - x_{i-1} > 0) \\ -Q_{\max} &\leq Q_i \leq 0 \quad (x_i - x_{i-1} < 0) \\ &i = 1, 2, 3 \cdots M + 1. \end{aligned} \quad (13)$$

Finally, the optimal discrete path is interpolated into a continuous smoothed path using B-spline cubic interpolation [9]:

$$Path_{1s}^{\sim} = \rho(x, z, T) = B^3(Path_{1s}) \quad (14)$$

4. The 2nd Stage of TF Path Planning

The planned path from the 1st stage is an approximate path that consists of terrain points that are taken as the input of 2nd-stage planning to generate a flyable path. A dynamic model of the UCAV is introduced in the 2nd stage for TF path planning and is used to generate a 4D time-based flight path.

4.1. Kinematic Model of the UCAV. To solve the TF path-planning problem for a fixed-wing UCAV, it is reasonable to

take 3-DOF motion into consideration rather than 6-DOF motion. The kinematic equations with airflow turbulence are

$$\begin{aligned}\dot{x} &= V \cos \theta \cos \psi + w_x \\ \dot{V} &= -g \sin \theta + \frac{T \cos \alpha - D}{m} - \dot{w}_x \cos \theta \cos \psi - \dot{w}_y \\ &\quad \cdot \sin \theta + \dot{w}_z \cos \theta \sin \psi \\ \dot{z} &= -V \cos \theta \sin \psi - w_z \\ \dot{\theta} &= -\frac{g \cos \theta}{V} + \frac{(T \sin \alpha + L) \sin \gamma}{mV} \\ &\quad + \frac{1}{V} (\dot{w}_x \sin \theta \cos \psi - \dot{w}_y \cos \theta - \dot{w}_z \sin \theta \sin \psi)\end{aligned}\quad (15)$$

$$\dot{h}_u = V \sin \theta + w_y$$

$$\dot{\psi} = \frac{1}{mV \cos \theta} ((T \sin \alpha + L) \sin \gamma + m\dot{w}_x \sin \psi + m\dot{w}_z \cos \psi)$$

$$D = 0.5 \rho_a S_r C_D V^2$$

$$L = 0.5 \rho_a S_r C_L V^2$$

$$-\theta_{\max} \leq \theta \leq \theta_{\max} \quad |\dot{\theta}| < \dot{\theta}_{\max} \quad (16)$$

$$-\psi_{\max} \leq \psi \leq \psi_{\max} \quad |\dot{\psi}| < |\dot{\psi}|_{\max}$$

$$-\alpha_{\max} \leq \alpha \leq \alpha_{\max} \quad |\dot{\alpha}| < \dot{\alpha}_{\max}$$

$$-\gamma_{\max} \leq \gamma \leq \gamma_{\max} \quad |\dot{\gamma}| < \dot{\gamma}_{\max}$$

where position variables x, z, h, V represent the longitude, latitude, height, and speed of the UCAV in a geographic coordinate system, respectively, and $\theta, \psi, \alpha, \gamma$ represent the path angle, heading angle, attack angle, and rolling angle of the UCAV, respectively. To simplify the dynamic model of the UCAV, mass m is assumed to be constant; T is the thrust of the UCAV's engine; L and D represent for the lift and drag, which are determined by the air density ρ_a , reference area S_r , pneumatic coefficient C_L/C_D , and speed. $W = (w_x, w_y, w_z, \dot{w}_x, \dot{w}_y, \dot{w}_z)$ is the variable of airflow turbulence, which consists of the wind speed and wind acceleration in the coordinate system of geography.

4.2. TF Performance Indicator. In this stage, the fundamental problem is to determine the optimal path for the UCAV to follow.

In order to ensure the reliability of the planned path for UCAV to fly, one of the fundamental indicator should be concerned is the total flight time, which can be expressed in an integral form:

$$t_{total} = \int_{t_0}^{t_f} dt = t_f - t_0 \quad (17)$$

The optimal path in the 2nd stage is a path closest to the terrain that satisfies all physical and flight constraints of

the UCAV. A general criterion of closeness that provides flexibility is defined in the form of simple clearance above the terrain:

$$e = h_u(x, z) - T(x, z) - c_0, \quad (18)$$

where h_u is the flight altitude of the UCAV and c_0 is the preset safe height in TF flight. In addition to the criterion of closeness, in the horizontal plane, with the purpose of tracking the path planned in the 1st stage, the tracking distance variable is

$$\Delta d = \sqrt{(x - x^{\sim})^2 + (z - z^{\sim})^2}, \quad (19)$$

where (x^{\sim}, z^{\sim}) represents the coordinate of the projection of $Path_{1s}^{\sim}$ on the horizontal plane. Thus, the performance indicator for TF path planning comprises the total flight time, the normalized closest criterion and tracking distance:

$$\begin{aligned}C_{2s} &= \int_{t_0}^{t_f} (\mu_1 \bar{t}_{total} + \mu_2 \bar{e} + \mu_3 \Delta \bar{d}) dt \\ \bar{t}_{total} &= \frac{1}{t_{\max}} \\ \bar{e} &= \frac{e}{e_s} \\ \Delta \bar{d} &= \frac{\Delta d}{\Delta d_s}\end{aligned}\quad (20)$$

where μ_1, μ_2 , and μ_3 are weight coefficients (with $\mu_1 + \mu_2 + \mu_3 = 1$), t_{\max} is the maximum flight time of the UCAV, and t_{\max}, e_s , and Δd_s are preset values used to calculate the normalized variables \bar{t}_{total}, \bar{e} , and $\Delta \bar{d}$, respectively.

4.3. TF Flight Path Generation. Instead of the curvature function, the kinematic equations and physical constraints of the UCAV are directly used in the 2nd stage to generate an optimal and flyable path. Accordingly, the path-planning problem is converted to a 4D time-based optimal control problem that can be formulated as

$$Path_{2s}(t, x, z, h_u) = \arg \min_{Path_{2s}} C_{2s}$$

$$C_{2s} = \int_{t_0}^{t_f} (\mu_1 \bar{t}_{total} + \mu_2 \bar{e} + \mu_3 \Delta \bar{d}) dt$$

$$\dot{X} = f(X, U, W)$$

$$X = (x, z, h_u, V, \theta, \psi)$$

$$U = (\alpha, \gamma) \quad (21)$$

$$W = (w_x, w_y, w_z, \dot{w}_x, \dot{w}_y, \dot{w}_z)$$

$$-\theta_{\max} \leq \theta \leq \theta_{\max} \quad |\dot{\theta}| < \dot{\theta}_{\max}$$

$$-\psi_{\max} \leq \psi \leq \psi_{\max} \quad |\dot{\psi}| < |\dot{\psi}|_{\max}$$

$$-\alpha_{\max} \leq \alpha \leq \alpha_{\max} \quad |\dot{\alpha}| < \dot{\alpha}_{\max}$$

$$-\gamma_{\max} \leq \gamma \leq \gamma_{\max} \quad |\dot{\gamma}| < \dot{\gamma}_{\max}$$

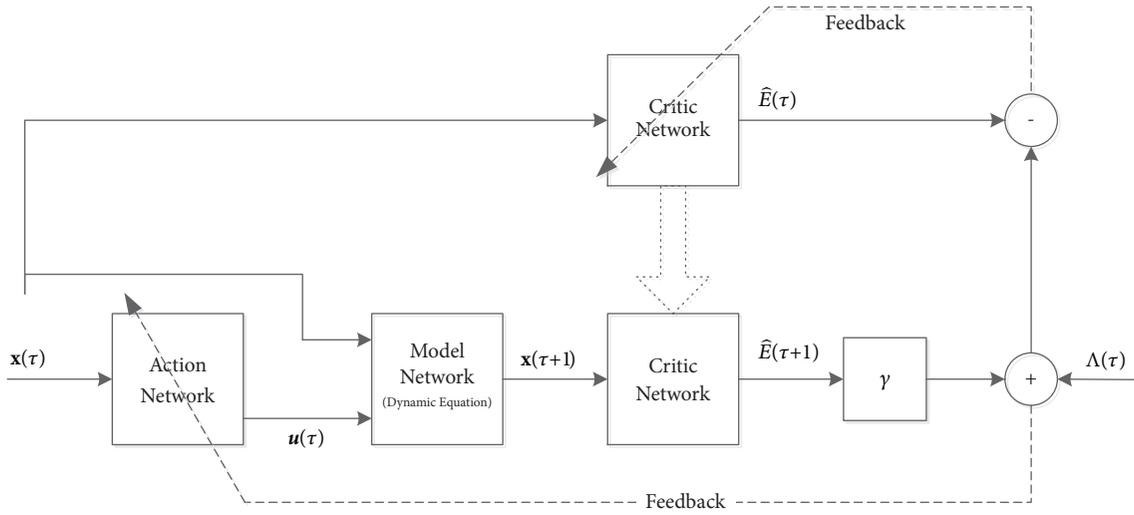


FIGURE 2: Composition of the ADP model.

5. ADP Algorithm

The DP algorithm has long been considered a feasible approach for solving the path-planning problem. However, the curse of dimensionality has restricted its maximum computational speed, which has motivated the development of several approximate techniques for composing the PF. Neural networks and kernel functions are considered effective approaches to achieve this idea in ADP. ADP reduces the dependence on the model through interaction with the system and can continuously improve the accuracy of the estimation and achieve the approximation of the optimal strategy.

The path-planning problem can be abstracted as a nonlinear continuous state model, and the model can be discretized into a nonlinear system with S subplanning steps:

$$\mathbf{x}(\tau + 1) = f(\mathbf{x}(\tau), \mathbf{u}(\tau)) \quad \tau = 1, 2, \dots, S. \quad (22)$$

The discrete PF of the planned path is posed as

$$E(\tau) = \sum_{j=\tau}^{\infty} \rho^{j-\tau} \Lambda(\tau); \quad 0 < \rho < 1 \quad (23)$$

$$\Lambda(\tau) = \Lambda(\mathbf{x}(\tau), \mathbf{u}(\tau)),$$

where ρ is the discount coefficient and $\Lambda(\tau)$ is the efficacy function of a single planning step determined by the related physical variables of theUCAV.

5.1. Modeling an ADP System. An approach called adaptive critic designs (ACDs) for ADP was first introduced by Werbos in 1977 and has recently received increasing attention [10]. After continuous improvements, a typical form of ACDs used in the modeling of ADP consists of three modules—the action network, model network and critic network—as shown in Figure 2. In the present work, each module is considered to be a backpropagation neural network

(BPNN). The motivations to adopt BPNN to the construction of ADP are mainly on its great potential for solving nonlinear problems and uncertain system control, which can be specifically explained as the following aspects: (1) its ability to fully approximate complex nonlinear mapping; (2) its superiority to learn and adapt to the dynamic characteristics of uncertain systems compared than other approaches; (3) BPNN is highly robust and fault tolerant; (4) distributed running processing is adopted in BPNN to make fast operations possible.

Thus, for the presented nonlinear dynamic model ofUCAV, in the critic network, evaluation value is calculated from the input of flight states by continuous training and threshold updating of BPNN; and in the action network, BPNN are continuously trained with the objective to approximate the optimal (or suboptimal) control variables based on the input of flight states, as shown in Figure 2.

5.2. Training and Threshold Update of the Critic Network. The structural scheme of the critic network is shown as Figure 3. There are 7 input layers and more than 13 hidden layers in the structure. ω_{c1} and ω_{c2} are the weight matrices of the neural network.

The flight state of theUCAV at time τ is taken as the input of the neural network. The critic network calculates and outputs the value $\hat{E}(\tau)$, which estimates the PF $E(\tau)$ in (23) by finding the minimal following error measures over time:

$$\|\xi_c\| = \sum_{\tau} \xi_c(\tau) = \frac{1}{2} \sum_{\tau} [\hat{E}(\tau) - \rho \hat{E}(\tau + 1) - \Lambda(\tau)]^2 \quad (24)$$

When $\xi_c(\tau) = 0$ for all τ , then

$$\begin{aligned} \hat{E}(\tau) &= \Lambda(\tau) + \rho \hat{E}(\tau + 1) \\ &= \Lambda(\tau) + \rho [\Lambda(\tau + 1) + \rho \hat{E}(\tau + 2)] = \dots \\ &= \sum_{i=k}^{\infty} \rho^{i-k} \Lambda(\tau), \end{aligned} \quad (25)$$

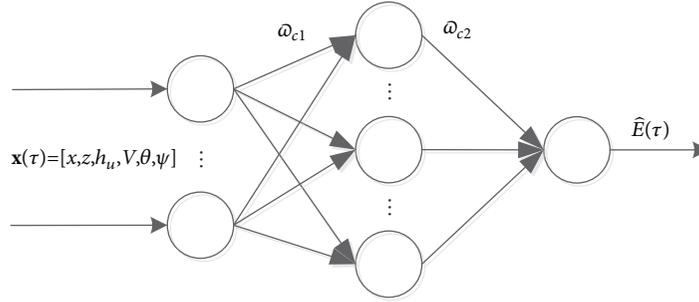


FIGURE 3: Structure of the critic network.

which is exactly the same as the PF in (22). Therefore, the training of the critic network is achieved by minimizing the following error defined by (23). The output of the critical network is calculated as

$$C_{h1j}(\tau) = \sum_{i=1}^{N_{in}} x_i(\tau) \cdot \omega_{c1ij}(\tau) + b_{c1j}(\tau) \quad (26)$$

$$C_{h2j}(\tau) = \frac{1 - e^{-2C_{h1j}(\tau)}}{1 + e^{-2C_{h1j}(\tau)}} \quad (27)$$

$$\hat{E}(\tau) = \sum_{j=1}^{N_h} C_{h2j}(\tau) \cdot \omega_{c2j}(\tau) + b_{c2}(\tau) \quad (28)$$

where $\omega_{c1ij}(\tau)$ is the i th row and j th column of the weight matrix ω_{c1} from the input layer to the hidden layer, b_{c1j} is the j th row of the threshold vector b_{c1} , $\omega_{c2j}(\tau)$ is the j th column of the weight matrix ω_{c2} from the hidden layer to the output layer, and b_{c2} is the threshold of the output layer. C_{h1j} and C_{h2j} are the input and output of the hidden layer, respectively. The update law of the weight matrix and threshold with the learning rate of the critical network $l_c(\tau)$ is given by the following equations:

$$\omega_{c2j}(\tau + 1) = \omega_{c2j}(\tau) - l_c(\tau) \frac{\partial \xi_c}{\partial \hat{E}(\tau)} \cdot \frac{\partial \hat{E}(\tau)}{\partial \omega_{c2j}(\tau)} \quad (29)$$

$$= \omega_{c2j}(\tau) - l_c(\tau) \cdot \xi_c(\tau) \cdot c_{h2j}(\tau),$$

$$b_{c2}(\tau + 1) = b_{c2}(\tau) - l_c(\tau) \frac{\partial \xi_c(\tau)}{\partial \hat{E}(\tau)} \cdot \frac{\partial \hat{E}(\tau)}{\partial b_{c2}(\tau)}, \quad (30)$$

$$\omega_{c1ij}(\tau + 1) = \omega_{c1ij}(\tau) - l_c(\tau) \frac{\partial \xi_c}{\partial \hat{E}(\tau)} \cdot \frac{\partial \hat{E}(\tau)}{\partial C_{h2j}(\tau)}$$

$$\cdot \frac{\partial C_{h2j}(\tau)}{\partial C_{h1j}(\tau)} \cdot \frac{\partial C_{h1j}(\tau)}{\partial \omega_{c1ij}(\tau)} \quad (31)$$

$$= \omega_{c1ij}(\tau) - l_c(\tau) \cdot \xi_c(\tau) \cdot \omega_{c2j}(\tau)$$

$$\cdot \frac{4e^{-2C_{h1j}(\tau)}}{(e^{-2C_{h1j}(\tau)} + 1)^2} \cdot x_i(\tau), \text{ and}$$

$$\begin{aligned} b_{c1j}(\tau + 1) &= b_{c1j}(\tau) - l_c(\tau) \frac{\partial \xi_c}{\partial \hat{E}(\tau)} \cdot \frac{\partial \hat{E}(\tau)}{\partial C_{h2j}(\tau)} \\ &\cdot \frac{\partial C_{h2j}(\tau)}{\partial C_{h1j}(\tau)} \cdot \frac{\partial C_{h1j}(\tau)}{\partial b_{c1j}(\tau)} \\ &= b_{c1j}(\tau) - l_c(\tau) \cdot \xi_c(\tau) \cdot \omega_{c2j}(\tau) \\ &\cdot \frac{4e^{-2C_{h1j}(\tau)}}{(e^{-2C_{h1j}(\tau)} + 1)^2}. \end{aligned} \quad (32)$$

5.3. Training and Threshold Updating of the Action Network. After training the critic network, the action network is trained with the objective of approximating the optimal control variable $\mathbf{u}(\tau)$ of the UCAV. Developed in the same way as the critic network, the structural scheme of the critic network is shown in Figure 4. In the scheme of the ADP model, the action network employs a BPNN to calculate the control variable $\mathbf{u}(\tau)$ of the UCAV on the basis of the motion state $\mathbf{x}(\tau)$.

The output calculation is defined as

$$A_{h1j} = \sum_{i=1}^{N_{in}} x_i(\tau) \cdot \omega_{a1ij}(\tau) + b_{a1j}(\tau), \quad (33)$$

$$A_{h2j}(\tau) = \frac{1 - e^{-2A_{h1j}(\tau)}}{1 + e^{-2A_{h1j}(\tau)}}, \text{ and} \quad (34)$$

$$\mathbf{u}(\tau) = \sum_{j=1}^{N_h} A_{h2j}(\tau) \cdot \omega_{a2j}(\tau) + b_{a2j}(\tau), \quad (35)$$

where $\mathbf{x}_i(k)$ are the input variables of the action network and ω_{a1ij} is the i th row and j th column of the weight matrix ω_{a1} from the input layer to the hidden layer. b_{a1j} is the j th row of the threshold vector b_{a1} , ω_{a2j} is the j th column of the weight matrix ω_{a2} from the hidden layer to the output layer, and b_{a2j} is the j th column of the threshold b_{a2} of the output layer. A_{h1j} and A_{h2j} are the input and output of the hidden layer, respectively. The update law of the weight matrix and

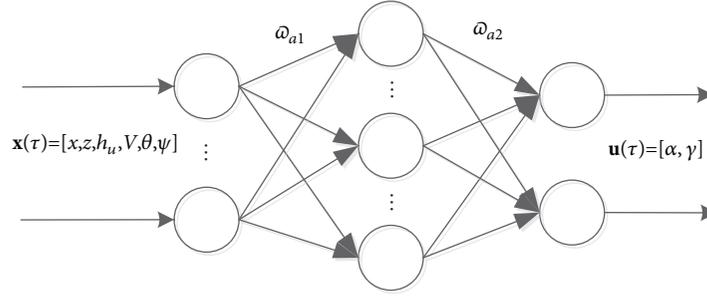


FIGURE 4: Action network structure.

threshold with the learning rate of the critical network $l_a(\tau)$ is given by the following equations:

$$\begin{aligned} \bar{\omega}_{a2ij}(\tau+1) &= \bar{\omega}_{a2ij}(\tau) - l_a(\tau) \frac{\partial \hat{E}(\tau)}{\partial u_i(\tau)} \cdot \frac{\partial u_i(\tau)}{\partial \bar{\omega}_{a2j}(\tau)} \\ &= \bar{\omega}_{a2ij}(\tau) - l_a(\tau) \frac{\partial \hat{E}(\tau)}{\partial u_i(\tau)} \cdot A_{h2j}(\tau), \end{aligned} \quad (36)$$

$$\begin{aligned} b_{a2j}(\tau+1) &= b_{a2j}(\tau) - l_a(\tau) \frac{\partial \hat{E}(\tau)}{\partial u_j(\tau)} \cdot \frac{\partial u_j(\tau)}{\partial b_{a2j}(\tau)} \\ &= b_{a2j}(\tau) - l_a(\tau) \frac{\partial \hat{E}(\tau)}{\partial u_j(\tau)}, \end{aligned} \quad (37)$$

$$\begin{aligned} \bar{\omega}_{a1ij}(\tau+1) &= \bar{\omega}_{a1ij}(\tau) - l_a(\tau) \frac{\partial \hat{E}(\tau)}{\partial a_{h2j}(\tau)} \cdot \frac{\partial a_{h2j}(\tau)}{\partial a_{h1j}(\tau)} \\ &\quad \cdot \frac{\partial a_{h1j}(\tau)}{\partial \bar{\omega}_{a1ij}(\tau)} \\ &= \bar{\omega}_{a2j}(\tau) - l_a(\tau) \frac{\partial \hat{E}(\tau)}{\partial a_{h2j}(\tau)} \\ &\quad \cdot \frac{-4e^{-2h_{a2j}(\tau)}}{(e^{-2h_{a2j}(\tau)} + 1)^2} \cdot x_i(\tau), \text{ and} \end{aligned} \quad (38)$$

$$\begin{aligned} b_{a1j}(\tau+1) &= b_{a1j}(\tau) - l_a(\tau) \frac{\partial \hat{E}(\tau)}{\partial a_{h2j}(\tau)} \cdot \frac{\partial a_{h2j}(\tau)}{\partial a_{h1j}(\tau)} \\ &\quad \cdot \frac{\partial a_{h1j}(\tau)}{\partial b_{a1j}(\tau)} \\ &= b_{a2j}(\tau) - l_a(\tau) \frac{\partial \hat{E}(\tau)}{\partial a_{h2j}(\tau)} \\ &\quad \cdot \frac{-4e^{-2h_{a2j}(\tau)}}{(e^{-2h_{a2j}(\tau)} + 1)^2}. \end{aligned} \quad (39)$$

5.4. Calculation Steps for ADP in 2nd-Stage Planning. (1) System initialization: load the smoothed collision-free $Path_{1s}^*$ from the 1st-stage, initialize the weight coefficients of the action network and critic network, set the initial values of the

learning rates $l_a(\tau)$ and $l_c(\tau)$, and define the iteration error ξ_c and the discount coefficient ρ .

(2) Load the initial flight state of the UCAV, input this state into the action network, and calculate the control variables of the UCAV.

(3) Take the control variables and flight state as the input of the model network and generate the state at the next time using the kinematic equations of the UCAV.

(4) Input the control variables and flight state into the critic network, and calculate the PF.

(5) Input the state for the next time into the action network, and calculate the control variables for the next time; then, input the new state and control variables into the critic network to calculate the PF for the next time.

(6) Calculate the error of the action network, train the action network according to the steps described in Section 5.3, and update the weight coefficient.

(7) Calculate the error of the critic network, train the critic network according to the steps described in Section 5.2, and update the weight coefficient.

(8) Loop over steps (2) to (7) until the maximum number of iterations is reached or the iteration error for stopping is satisfied.

6. Simulation Results and Analysis

To verify the performance of the discussed path-planning method, numerical simulation results and analysis are presented in this section. The planning system and UCAV object were built using Visual C++ (ver. 10.0), and the results figures were created using MATLAB. The entire simulation was performed on a workstation consisting of a 3.5 GHz Intel Core-i7 CPU and 16 GB of physical RAM running 64-bit Microsoft Windows 10. Some of the flight characteristic parameters of the UCAV are taken from that of the MQ-9 "Reaper", and the simulation parameters and initial conditions are listed in Table 1.

The air density can be calculated using

$$\rho_a = 1.225 \cdot e^{(-h/9300)} \quad (40)$$

The digital map used in this study was built on GIS data from the National Geomatics Center of China (NGCC), covered an area of $33 \times 33 \text{ km}^2$ and had a sample distance of 0.03 km. Suppose that the space higher than 2000 m is covered by enemy radar detection and that the UCAV can

TABLE 1: Parameter values for the simulation model.

Mass/kg	Reference area/m ²	Max flt. height/m	Avg. trust/N
1030	19.4	2000	2.1×10 ⁴
$[V_{\min}, V_{\max}]/\text{m}\cdot\text{s}^{-1}$	$[\theta_{\max}, \psi_{\max}]/\text{deg}$	$[\dot{\theta}_{\max}, \dot{\psi}_{\max}]/\text{deg}\cdot\text{s}^{-1}$	$[\alpha_{\max}, \gamma_{\max}]/\text{deg}$
[100, 200]	[30, 90]	[5, 5]	[10, 30]
$[\dot{\alpha}_{\max}, \dot{\gamma}_{\max}]/\text{deg}\cdot\text{s}^{-1}$	$[w_x, w_y, w_z]/\text{m}\cdot\text{s}^{-1}$	$[\dot{w}_x, \dot{w}_y, \dot{w}_z]/\text{m}\cdot\text{s}^{-2}$	Avg. $[C_D, C_L]$
[5, 5]	[3, 3, 3]	[1, 1, 1]	[0.272, 0.316]
$P_0 = [x_0, z_0]/\text{km}$	$P_f = [x_f, z_f]/\text{km}$	c_0/m	Max overload
[6.19, 24.23]	[31.35, 14.75]	60	5 g

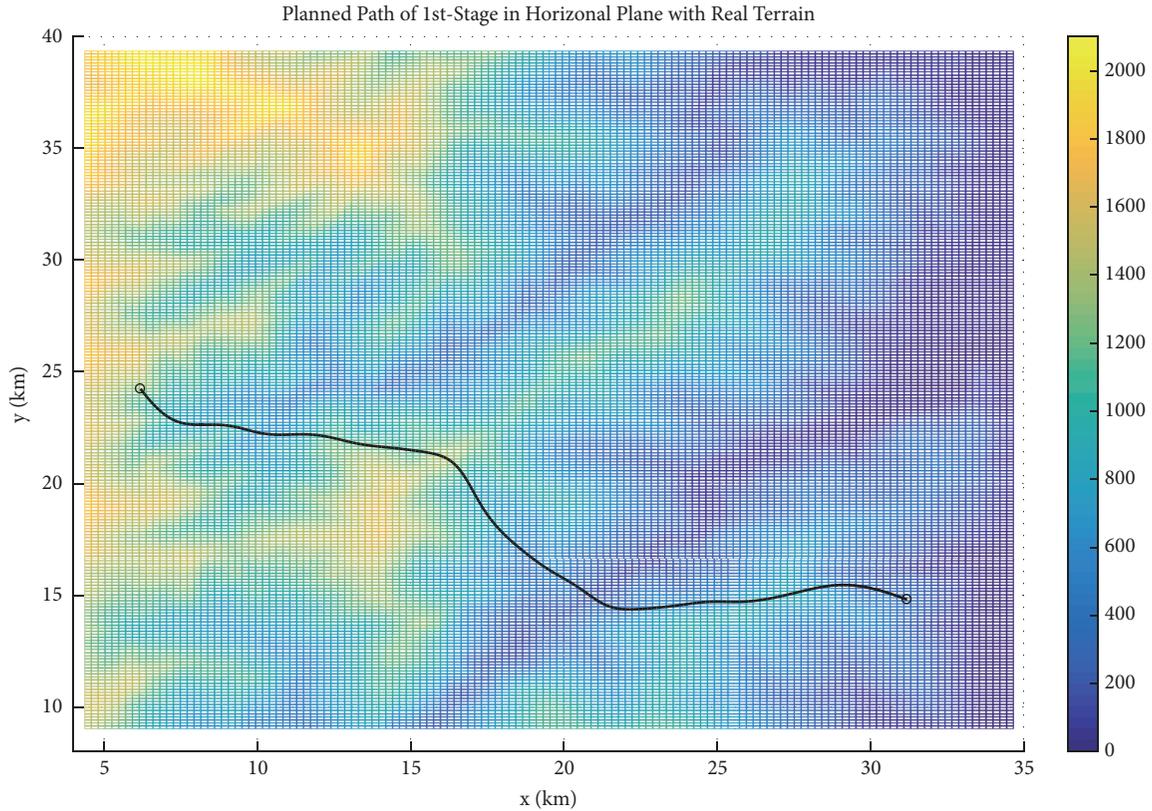


FIGURE 5: Planned path of the 1st stage in a horizontal plane with real terrain.

never fly higher than that altitude. In the 1st stage of planning for TA, the conventional DP algorithm was used to solve the optimization problem, and the maximum permission number of iterations was 500. The other coefficients were

$$(\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5) = (0.05, 0.14, 0.32, 0.27, 0.22)$$

$$(\mu_1, \mu_2, \mu_3) = (0.3, 0.4, 0.3)$$

$$K_{\max} = \tan \theta_{\max} = 0.577 \quad (41)$$

$$K_{\min} = -K_{\max} = -0.577$$

$$Q_{\max} = \tan \psi_{\max} = 1.192.$$

The results from the 1st stage of planning are shown in Figure 5; the attached color bar represents the terrain height

of the selected area, red represents the higher terrain, and blue is used for the lower terrain. Because the weight coefficients α_1 - α_5 were set to prioritize bypassing the terrain rather than flying over it, the optimal planned path avoided the UCAV-prohibited area as much as possible. In addition, because the collision punishment coefficient was far larger than the other indicators, the paths with existing or potential collisions were eliminated earlier in the DP operation process.

After obtaining the approximate path from the 1st stage, the 2nd stage of planning for TF generated the optimal flyable path shown in Figure 6. In Figure 6, the blue dots represent the profile of the terrain, and the red line represents the optimal 3D path.

To further verify the feasibility of flight for the planned path, the flight states and control variables are shown in

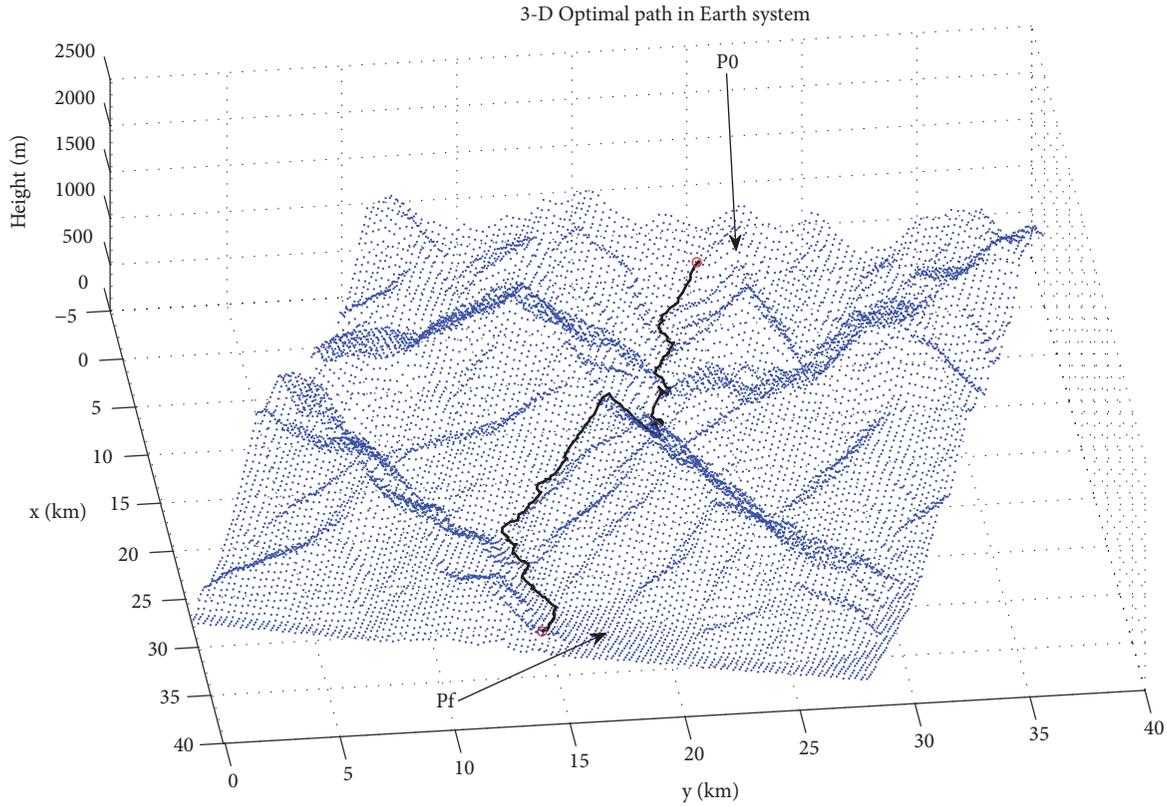


FIGURE 6: Optimal 3D path in the Earth system.

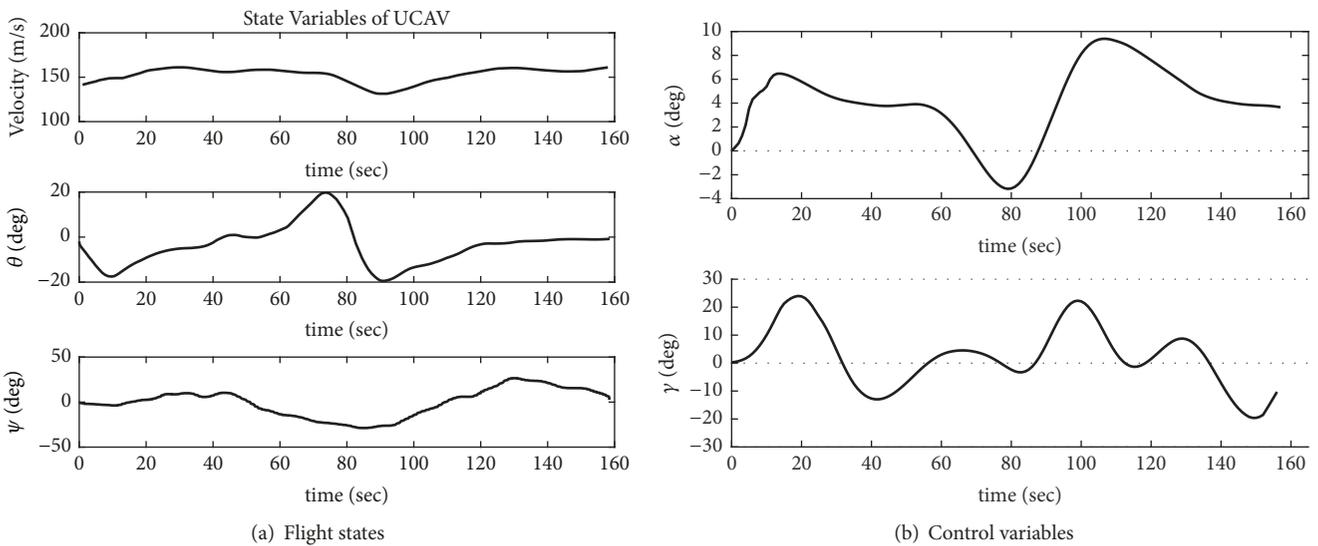


FIGURE 7: Flight states and control variables of the UCAV.

Figure 7. The presented variables satisfy the flight constraints and characteristics mentioned in Table 2. In addition, the iteration errors for the ADP algorithm in the 2nd stage are shown in Figure 8; the maximum permissible number of iterations was the same as for the 1st stage, and the convergence error was set to 10^{-5} . After 468 iterations, the

error reached the convergence requirement, indicating that the result was optimal under these conditions. Therefore, we conclude that the planned path is an optimal flyable path for the proposed UCAV.

At the end, to verify the superiority of the proposed method used in the 2nd stage, an optimization approach

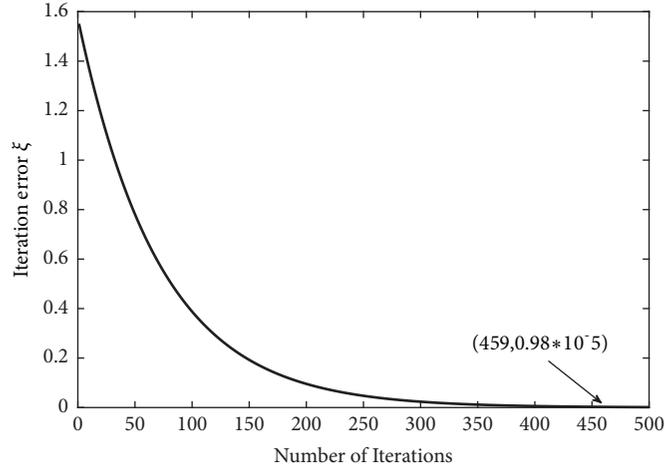
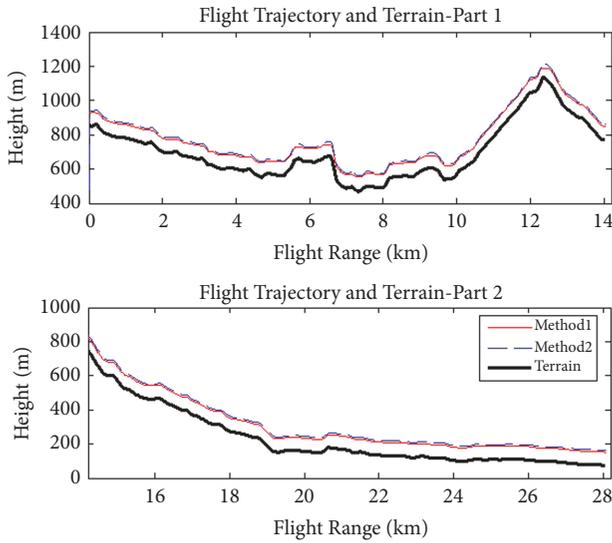
FIGURE 8: Iteration error for the ADP algorithm in the 2nd stage.

FIGURE 9: Flight trajectory and terrain.

with a GA with the same maximum permitted iteration number and convergence error is implemented for comparison. Figure 9 shows the heights of the planned trajectories using the 2 methods and the terrain profile beneath the trajectories; the distances between the flight trajectories and terrain are presented in Figure 10. Method 1 refers to the method proposed here, and Method 2 represents the GA optimization method. The computational efficiencies of the two methods are also listed in Table 2.

By cross-comparing the two graphs, we found that the trajectory generated by Method 1 was smoother than that for Method 2 (in which the height changes slowly) due to the consideration of flyability. Thus, regarding the same terrain height, the distance of Method 1 changed more severely than that of Method 2. In practical engineering, the inclination would be to let the UCAV track a smoother trajectory, but, in the optimization approach, a smoother trajectory could degrade the performance index. Regarding

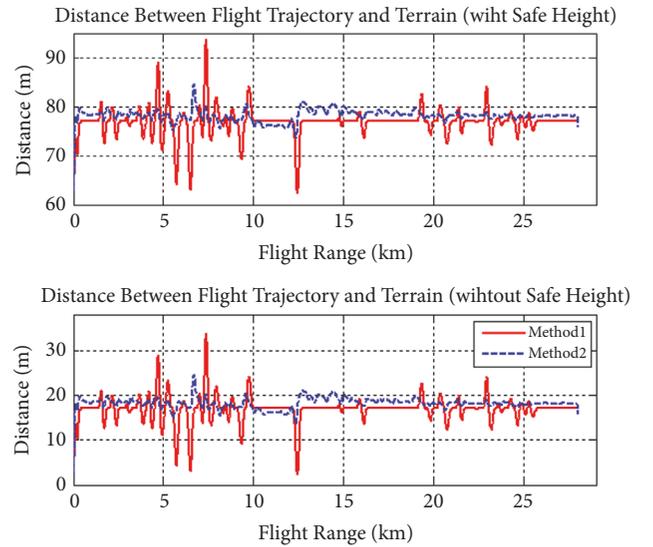


FIGURE 10: Distances between the flight trajectories and terrain.

TABLE 2: Computational efficiencies for methods 1 and 2.

Method	1	2
Calculation time	0.291 s	0.310 s
Iteration number	459	419
Max CPU/RAM used	55%/41%	58%/47%

computational efficiency, the two methods were comparable, although Method 1 was slightly faster. Therefore, the optimal control approach using the ADP algorithm enjoys a certain degree of superiority over the optimization approach in path-planning problems.

7. Conclusion

We have introduced a novel two-stage path-planning methodology for TF/TA path planning in 3D space that

decomposes the original planning problem into two phases: a 3D optimization problem and an optimal control problem. This proposed method is based on the ADP algorithm and can obtain optimal or approximately optimal solutions through iterative optimization. The simulation results indicate that this two-stage method can solve TF/TA path-planning problems better than the conventional optimization method in some respects. The superiority of the proposed method is that it accounts for the flyability of the planned path and the optimal performance indicators.

Therefore, more reasonable critical indicators should be taken into account to evaluate the characteristics of planned paths in future research. The next step will consist of modifying the modeling aspects and algorithms for online planning.

Data Availability

The values of some of the simulation parameters are available online, and other data is original from the authors of this paper.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by funding of Jiangsu Innovation Program for Graduate Education, KYLX15 0319, “the Fundamental Research Funds for the Central Universities.”

References

- [1] C. Goerzen, Z. Kong, and B. Mettler, “A survey of motion planning algorithms from the perspective of autonomous UAV guidance,” *Journal of Intelligent & Robotic Systems*, vol. 57, no. 1-4, pp. 65–100, 2010.
- [2] V. Roberge, M. Tarbouchi, and G. Labonte, “Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning,” *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 132–141, 2013.
- [3] H. Wang, Q. Li, and N. Cheng, “Real-time path planning for low altitude flight based on A* algorithm and TF/TA algorithm,” in *Proceedings of the 2012 IEEE International Conference on Automation Science and Engineering: Green Automation Toward a Sustainable Society, CASE 2012*, pp. 837–842, Republic of Korea, August 2012.
- [4] D. Americo and J. Caves, *Human-automation collaborative RRT for UAV mission path planning*, Institute of Technology, 2010.
- [5] G. X. Guang, X. W. Fu, and D. Q. Chen, “A genetic-algorithm-based approach to UAV path planning problem,” in *Proceedings of the Wseas International Conference on Simulation, Modelling and Optimization World Scientific and Engineering Academy and Society (WSEAS)*, pp. 523–527, 2005.
- [6] R. Samar and A. Rehman, “Autonomous terrain-following for unmanned air vehicles,” *Mechatronics*, vol. 21, no. 5, pp. 844–860, 2011.
- [7] X. Xu, C. Lian, L. Zuo, and H. He, “Kernel-based approximate dynamic programming for real-time online learning control: an experimental study,” *IEEE Transactions on Control Systems Technology*, vol. 22, no. 1, pp. 146–156, 2014.
- [8] C. Novoa and R. Storer, “An approximate dynamic programming approach for the vehicle routing problem with stochastic demands,” *European Journal of Operational Research*, vol. 196, no. 2, pp. 509–515, 2009.
- [9] T. Maekawa, T. Noda, S. Tamura, T. Ozaki, and K. Machida, “Curvature continuous path generation for autonomous vehicle using B-spline curves,” *Computer-Aided Design*, vol. 42, no. 4, pp. 350–359, 2010.
- [10] B. Hérissé, T. Hamel, R. Mahony, and F.-X. Rusotto, “A terrain-following control approach for a VTOL Unmanned Aerial Vehicle using average optical flow,” *Autonomous Robots*, vol. 29, no. 3-4, pp. 381–399, 2010.



Hindawi

Submit your manuscripts at
www.hindawi.com

