doi:10.20944/preprints202210.0360.v1

Attitude Control of Highly Maneuverable Aircraft Using an Improved Q-learning

Mohsen. Zahmatkesh * **Seyyed. Ali. Emami** *
Afshin. Banazadeh * **Paolo. Castaldi** **

* *Aerospace Engineering Department, Sharif University of Technology, Tehran, Iran (e-mail: banazadeh@sharif.edu).*
** *Department of Electrical, Electronic and Information Engineering "Guglielmo Marconi", University of Bologna, Via Dell'Universit'a 50, Cesena, Italy (e-mail: paolo.castaldi@unibo.it)*

Abstract: Attitude control of a novel regional truss-braced wing aircraft with low stability characteristics is addressed in this paper using Reinforcement Learning (RL). In recent years, RL has been increasingly employed in challenging applications, particularly, autonomous flight control. However, a significant predicament confronting discrete RL algorithms is the dimension limitation of the state-action table and difficulties in defining the elements of the RL environment. To address these issues, in this paper, a detailed mathematical model of the mentioned aircraft is first developed to shape an RL environment. Subsequently, Q-learning, the most prevalent discrete RL algorithm will be implemented in both the Markov Decision Process (MDP), and Partially Observable Markov Decision Process (POMDP) frameworks to control the longitudinal mode of the air vehicle. In order to eliminate residual fluctuations that are a consequence of discrete action selection, and simultaneously track variable pitch angles, a Fuzzy Action Assignment (FAA) method is proposed to generate continuous control commands using the trained Q-table. Accordingly, it will be proved that by defining an accurate reward function, along with observing all crucial states (which is equivalent to satisfying the Markov Property), the performance of the introduced control system surpasses a well-tuned Proportional–Integral–Derivative (PID) controller.

Keywords: Reinforcement Learning, Q-learning, Fuzzy Q-learning, Attitude Control, Truss-braced Wing, Flight Control

1. INTRODUCTION

The aviation industry is expeditiously growing due to world demands such as reducing fuel burn, emissions, and cost, as well as providing the faster and safer flight. This motivates the advent of new airplanes with novel configurations. In addition, the scope clause agreement limits the number of seats in each aircraft and flight outsourcing to protect the union pilot jobs. This factor leads to an increase in production of the Modern Regional Jet (MRJ) airplane. In this regard, the importance of a safe flight becomes more vital considering more crowded airspace and new aircraft configurations having the ability to fly faster. Truss-braced wing aircraft is one of the re-raised high-performance configurations, which has attracted significant attention from both academia (Li et al., 2022) and industry (Sarode, 2022) due to its fuel burn efficiency. As a result, there would be a growing need for reliable modeling and simulations, analyzing the flight handling quality, and stability analysis for such configurations (Nguyen and Xiong, 2022; Zavaree et al., 2021), while very few studies have addressed the flight control design for this aircraft.

In the last decades, various classic methods for aircraft attitude control have been developed to enhance control performance. However, the most significant deficiency of these approaches is the insufficient capability to deal with unexpected flight conditions, while typically requiring a detailed dynamic model of the system.

Recently, the application of Reinforcement Learning (RL) has been extended to real problems, particularly, flight control design (Emami et al., 2022). Generally, there are two main frameworks to incorporate RL in the control design process, i.e., the high-level and low-level control systems. In Xi et al. (2022), a Soft Actor-Critic (SAC) algorithm was implemented in a path planning problem for a long-endurance solar-powered UAV with energy-consuming considerations. Another work (Bøhn et al., 2021) concentrated on the inner loop control of a Skywalker X8 using SAC and comparing it with a PID controller. In Yang et al. (2020) a ANN based Q-learning horizontal trajectory tracking controller was developed based on the MDP model of an airship with fine stability characteristics. Apart from the previous method, Proximal Policy Optimization (PPO) was utilized in Hu et al. (2022) for orientation control of a common strongly dynamic coupled fixed-wing aircraft in the stall condition. The PPO was successful to be converged after 100000 episodes. However, useful to say that the PPO performance is adequate to optimize PID controllers (Dammen, 2022).

There are some papers on maneuver flight such as landing phase control both in inner and outer loops. For instance, in Wang et al. (2018), a Deep Q-learning (DQL) is used to guide an aircraft to land in the desired field. In Yuan et al. (2019), a Deep Deterministic Policy Gradient (DDPG) was implemented for a UAV to control either path-tracking for landing glide slope and attitude control for landing flare section. Similarly, a DDPG method in Tang and Lai (2020) is used to control outer loop of a landing procedure in presence of wind disturbance. The works which have been referred to so far accompanied ANNs to be able to converge. But to our best knowledge, there is research in attitude control using discrete RL without aiming ANNs. In Richter et al. (2022), a Q-learning algorithm was implemented to control longitudinal and lateral angles in a general aviation aircraft(Cessna 172). This airplane profits suitable stability characteristics and also desired angles are zero. There are some fuzzy adaptations on Watkins and Dayan (1992) work like Glorennec and Jouffe (1997) where the Q-functions and action selection strategy are inferred from fuzzy rules. Also, Er and Deng (2004) proposed a dynamic fuzzy Q-learning for online and continuous tasks in mobile robots.

Motivated by the above discussions, the main contributions of the current study can be summarized as follows: a) A truss-braced wing aircraft (Chaka 50) (1) with poor stability characteristics has been selected carefully for attitude control alongside responding to global aviation society demands. b) It will be proven that the Q-learning performance in control problems strictly depends on reward function and problem definition. So, it is able to have prosperous results even in a low stable high degree of freedom plants. c) In this work, the performance of Q-learning will be examined in both MDP and POMDP problem modelings. Also, the learned Q-table is used to generate continuous elevator deflections using Fuzzy Action Assignment (FAA) illustrating Q-table capability tracking the desired angle and also variable angles.

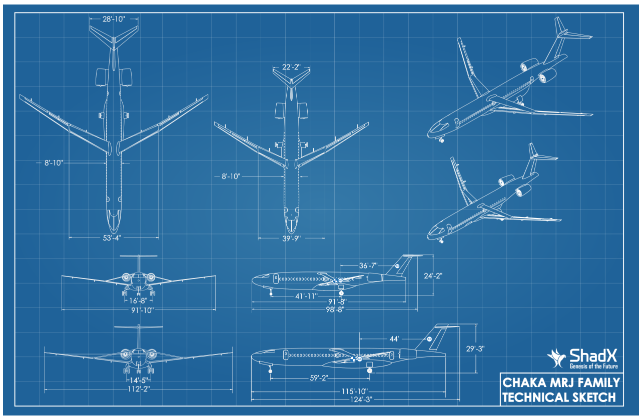

Fig. 1. Chaka MRJ Family (Zavaree et al., 2021)

2. MODELING AND SIMULATION

Nonlinear conservation of linear and angular momentum equations are used for modeling and simulation according to Zipfel (2014).

$$m\begin{bmatrix}\dot{u}\\ \dot{v}\\ \dot{w}\end{bmatrix}+m\begin{bmatrix}0 & -r & q\\ r & 0 & -p\\ -q & p & 0\end{bmatrix}\begin{bmatrix}u\\ v\\ w\end{bmatrix}=\begin{bmatrix}F_{A_x+T_x}\\ F_{A_y+T_y}\\ F_{A_z+T_z}\end{bmatrix}+m\begin{bmatrix}g_x\\ g_y\\ g_z\end{bmatrix} \tag{1}$$

$$\begin{bmatrix}I_x & 0 & 0\\ 0 & I_y & 0\\ 0 & 0 & I_z\end{bmatrix}\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix}+\begin{bmatrix}0 & -r & q\\ r & 0 & -p\\ -q & p & 0\end{bmatrix}\begin{bmatrix}I_x & 0 & 0\\ 0 & I_y & 0\\ 0 & 0 & I_z\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix}=\begin{bmatrix}L_{A+T}\\ M_{A+T}\\ N_{A+T}\end{bmatrix} \tag{2}$$

where assuming the moments of thrust are equal to zero and also the thrust is only implying in x direction. Therefore, $L_T = M_T = N_T = F_{T_y} = F_{T_z} = 0$. and the aerodynamic forces and moments in body axis are as follows:

$$\begin{bmatrix}F_{A_x}\\ F_{A_z}\\ M_A\end{bmatrix}^B=\bar{q}S\bar{c}\begin{bmatrix}c_{L_0} & c_{L_\alpha} & c_{L_{\dot{\alpha}}} & c_{L_u} & c_{L_q} & c_{L_{\delta_E}}\\ c_{D_0} & c_{D_\alpha} & c_{D_{\dot{\alpha}}} & c_{D_u} & c_{D_q} & c_{D_{\delta_E}}\\ c_{m_0} & c_{m_\alpha} & c_{m_{\dot{\alpha}}} & c_{m_u} & c_{m_q} & c_{m_{\delta_E}}\end{bmatrix}\begin{bmatrix}1\\ \alpha\\ \frac{\dot{\alpha}\bar{c}}{2V_{P_1}}\\ \frac{u}{V_{P_1}}\\ \frac{q\bar{c}}{2V_{P_1}}\\ \delta_E\end{bmatrix} \tag{3}$$

The vector of gravity acceleration in the body axis defined by (1) is as follows:

$$\begin{bmatrix}g_x\\ g_y\\ g_z\end{bmatrix}^B=\begin{Bmatrix}-g\sin(\theta)\\ g\cos(\theta)\sin(\phi)\\ g\cos(\theta)\cos(\phi)\end{Bmatrix} \tag{4}$$

And also, the rotational kinematic equations are necessary for transfer from body to inertia coordinates.

$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix}=\begin{bmatrix}1 & \sin\varphi\tan\theta & \cos\varphi\tan\theta\\ 0 & \cos\varphi & -\sin\varphi\\ 0 & \sin\varphi/\cos\theta & \cos\varphi/\cos\theta\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{5}$$

using (1), and (5), velocity vector in inertia coordinate is achievable. For contraction reasons; $\mathrm{sin}=\mathrm{s}, \mathrm{cos}=\mathrm{c}$.

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix}^I=\begin{bmatrix}\mathrm{c}\psi\mathrm{c}\theta & \mathrm{c}\psi\mathrm{s}\theta\mathrm{s}\varphi-\mathrm{s}\psi\mathrm{c}\varphi & \mathrm{c}\psi\mathrm{s}\theta\mathrm{c}\varphi+\mathrm{s}\psi\mathrm{s}\varphi\\ \mathrm{s}\psi\mathrm{c}\theta & \mathrm{s}\psi\mathrm{s}\theta\mathrm{s}\varphi+\mathrm{c}\psi\mathrm{c}\varphi & \mathrm{s}\psi\mathrm{s}\theta\mathrm{c}\varphi-\mathrm{c}\psi\mathrm{s}\varphi\\ -\mathrm{s}\theta & \mathrm{c}\theta\mathrm{s}\varphi & \mathrm{c}\theta\mathrm{c}\varphi\end{bmatrix}\begin{bmatrix}u\\ v\\ w\end{bmatrix}^B \tag{6}$$

Stability and control derivatives for the Chaka-50 are reported in Zavaree et al. (2021) based on Computational Fluid Dynamics (CFD). The summary of these derivatives for two flight phases is given in table (1). before six-degree-of-freedom (6DoF) simulation using equations (1-2), the trim conditions in a wings-level flight are calculated for simulation verification based on trim equations in Roskam (1998). In drag equation, absolute value of δ_E, i_{H_1}, α_1 is considered. Also, flight path angle γ_1, motor installation angle ϕ_T, and horizontal tail incidence angle i_H, are zero. The elevator deflection δ_E, and required thrust T_1 for a trim flight is obtained and shown in table (2). The values in the table (2) are important for 6DoF simulation validation.

Atmospheric Disturbance and Sensor Measurement Noise

This research has utilized the Dryden atmosphere turbulence for its simple mathematical modeling.

$$G_w(s)=\sigma_w\sqrt{\frac{L_w}{\pi u_1}}\left[\frac{1+\sqrt{3}\frac{L_w}{u_1}s}{1+(\frac{L_w}{u_1}s)^2}\right] \tag{7}$$

This model is applied in w direction where $L_w = h = 100m$, $\sigma_w = 10$, and $u_1 = 160\frac{m}{s}$. In addition, the sensor noise is defined as 10% of sensor measurement. Also, the geometric, mass, and moment of inertia data are given in the table (3).

Table 1. Stability, Control derivatives (1/rad)

Longitudinal Derivatives	Take-off	Cruise
c_{D_0}	0.0378	0.0338
c_{L_0}	0.3203	0.3180
c_{m_0}	-0.07	-0.06
c_{D_α}	0.95	0.8930
c_{L_α}	11.06	14.88
c_{m_α}	-12.18	-11.84
c_{D_u}	0.040	0.041
c_{L_u}	0	0.081
c_{m_u}	0	-0.039
c_{D_q}	0	0
c_{L_q}	11.31	12.53
c_{m_q}	-40.25	-40.69
$c_{D_{\delta_E}}$	0.1550	0.1570
$c_{L_{\delta_E}}$	0.96	0.78
$c_{m_{\delta_E}}$	-6.15	-5.98

Table 2. Trim Conditions of Chaka MRJ

Required Thrust (T_1)(lbs)	AoA (α°)	Required Elevator (δ°_E)
21433.02	0.39	-2.28

Table 3. Simulation Parameters

Parameter	Value	Parameter	Value
Wing Area(m^2)	43.42	$I_{xx}(kg.m^2)$	378056.535
Mean Aerodynamic Chord(m)	1.216	$I_{yy}(kg.m^2)$	4914073.496
Span(m)	28	$I_{zz}(kg.m^2)$	5670084.803
Mass(kg)	18418.27	$I_{xz}(kg.m^2)$	0

3. ATTITUDE CONTROL PROCESS USING Q-LEARNING

Q-learning is an off-policy, model-free control strategy algorithm that by interacting with an environment, seeks to find the best action to take given the current state. It is a branch of Temporal Difference (TD) which is a combination of Dynamic Programming, and Monte Carlo theories. Truss-braced wing aircraft usually suffer inadequate stability owing to their narrow mean aerodynamic chord (MAC). For example, the Phugoid and Short Period poles of Boeing N+3 TTBW (Nguyen and Xiong, 2022), and Chaka 50 (Zavaree et al., 2021) prove that intuitively in comparison with corresponding poles in Çetin (2018) for the Cessna 172. A summary of numerical results has been gathered in table (4).

Table 4. Longitudinal Dynamics Characteristic

Aircraft Roots	Short Period Roots	Phugoid Roots
Chaka 50	$-0.8 \pm 0.61i$	$-0.0064 \pm 0.05i$
Cessna 172	$-3.23 \pm 5.71i$	$-0.025 \pm 0.19i$
Boeing N+3	$-0.35 \pm 0.35i$	$-0.0082 \pm 0.07i$

3.1 MDP and POMDP Definition in Attitude Control

It is necessary to formalize sequential decision making like aircraft attitude control as MDPs where one action influence not just next state and its immediate reward, but also upcoming states and their future rewards (Sutton and Barto, 2018). For clarification, At each time-step t, the controller normally receives the state's information including $\theta_t \in \mathcal{S}_1$, and $\dot{\theta}_t \in \mathcal{S}_2$ from the environment. Based on that, the controller selects an action which in this model is the elevator deflection, $\delta_{E_t} \in \mathcal{A}(s)$. The simulation executes and in next time-step $t+1$, the controller receives a numerical reward, $R_{t+1} \in \mathcal{R}$ to evaluate its performance and find itself in next state, θ_{t+1}, $\dot{\theta}_{t+1}$.

$$\theta_0,\ \dot{\theta}_0,\ \delta_{E_0},\ R_1,\ \theta_1,\ \dot{\theta}_1,\ \ldots \tag{8}$$

In this problem, a random selection of R_t, θ_t, and $\dot{\theta}_t$ have a clear discrete probability distribution dependent on previous state and action only. Also, the problem has Markov property by considering θ_t, and $\dot{\theta}_t$ as states. The purpose of reinforcement learning finite MDP is to find a policy that gathers maximum reward over time. Consequently, to find an optimal policy for taking δ_E in state θ, $\dot{\theta}$, the state-action value function $Q_\pi(\theta, \dot{\theta}, a)$ defines to be maximize by expected return that is sum of discounted instant rewards by starting from one specific state following policy π to terminal state θ_T, $\dot{\theta}_T$.

$$Q_\pi(\theta,\ \dot{\theta},\ \delta_E) = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty}\gamma^k R_{t+k+1}\ \middle|\ \theta_t = \theta,\ \dot{\theta}_t = \dot{\theta},\ \delta_{E_t} = \delta_E\right], \tag{9}$$

Where γ is the discount factor for rewards to be weighted based on its time-step, and usually is $0 < \gamma < 1$.

3.2 Structure of Q-learning Controller

In this work, the optimal policy of elevator action selection in each state is approximated directly using an early breakthrough in reinforcement learning namely Q-learning (Watkins and Dayan, 1992). Because of the non-linearity and poor stability characteristics of Truss-braced wing aircraft, Q-learning implementation without utilizing NNs can be challenging.

Reward Function and Action Space Definition Defining an efficient reward function plays a main role in algorithm convergence. Therefore, this research has concentrated carefully in reward function design and hyper parameter tuning. In this way, the reward function contains different components such as θ, q, δ_E. The reward function would be computed in three consecutive steps. First, to restrict the operating frequency of the elevator, it is essential to give a large punishment in the case of aggressive elevator selection:

$$Reward_t = -10000, \quad \text{If } \left(|\delta_{E_t}| - |\delta_{E_{t-1}}|\right) > 0.1 \text{ rad}. \tag{10}$$

Subsequently, if the change rate of the elevator is satisfactory, the reward function will be computed as follows if the aircraft is in the vicinity of the desired state.

$$\begin{aligned} Reward_t = &(300, && \text{If } |err_{p_t}| < 0.05^\circ) \\ &+(300, && \text{If } |err_{p_t}| < 0.02^\circ) \\ &+(400, && \text{If } |q_{sim_t}| < 0.04^\circ) \\ &+(600, && \text{If } |q_{sim_t}| < 0.02^\circ) \\ &+(800, && \text{If } |q_{sim_t}| < 0.005^\circ), \end{aligned} \tag{11}$$

where $err_{p_t} = \theta_{sim_t} - \theta_{des_t}$. Finally, if none of the above two conditions are met, we should encourage the air vehicle

doi:10.20944/preprints202210.0360.v1

to move towards the desired state. This can be done using the following reward function:

$$Reward_t = -(100 \times |err_{p_t}|)^2 - (40 \times |q_{sim_t}|)^2. \qquad (12)$$

Accordingly, the farther the system is from the desired state, the less reward it receives. Also, a derivative term (the second term) has been incorporated into the reward function to avoid high pitch rates.

Considering the tabular Q-learning, the elevator commands are obtained discretely. So, the elevator commands are divided into -0.25 to $+0.25$ radians with 0.025 intervals, corresponding 21 elevator deflections. Also, ϵ-greedy action selection strategy with epsilon decay is used in this research.

$$\delta_{E_t} = \begin{cases} \arg\max Q(\theta_t,\, \dot{\theta}_t,\, \delta_E) & \text{with probability } 1-\epsilon \\ \text{random action} & \text{with probability } \epsilon \end{cases} \qquad (13)$$

The mentioned reward function has developed carefully during lots of dynamic examinations. Against some works, this research believes that the quality of learning convergence not only is not related to trial and error but also, is related to deep comprehension of dynamic feedback. For clarification, we first were omitting to observe q_{sim} feedback in equation (11) that caused worse results. Also, the proportion of the first term to the second in equation (12) plays a substantial role in the convergence rate.

3.3 Structure of Fuzzy Action Assignment

In order to make continuous elevator actions, a Fuzzy Action Assignment (FAA) is presented to evaluate the capability of the learned Q-table. In this method, instead of taking a discrete greedy action in a given state θ, $\dot{\theta}$, a weighted δ_E is selected based on the proposed membership function (MF). More precisely, the membership function corresponding to a cell of the table with θ_i and $\dot{\theta}_j$ is defined as follows:

$$MF_{i,j} = \exp\left(-\frac{1}{2}\left(\frac{\theta_{sim_t} - \theta_i}{\sigma_\theta}\right)^2\right) \exp\left(-\frac{1}{2}\left(\frac{q_{sim_t} - \dot{\theta}_j}{\sigma_{\dot{\theta}}}\right)^2\right), \qquad (14)$$

Where $\sigma_{\theta_i} = \frac{\theta_i - \theta_{i-1}}{2}$, and $\dot{\sigma}_{\theta j} = \frac{\dot{\theta}_j - \dot{\theta}_{j-1}}{2}$ are the half-length of the state span, provided that $\theta_i > \theta_{i-1}$, and $\dot{\theta}_j > \dot{\theta}_{j-1}$. Also, $\theta_n = \frac{\theta_i + \theta_{i-1}}{2}$, and $\dot{\theta}_n = \frac{\dot{\theta}_j + \dot{\theta}_{j-1}}{2}$ are the center of each state. So the δ_E at each time-step is calculated as:

$$\delta_{E_t} = \frac{\sum_i \sum_j MF_{i,j} \arg\max Q(\theta_i,\, \dot{\theta}_j,\, \delta_E)}{\sum_i \sum_j MF_{i,j}}. \qquad (15)$$

4. SIMULATION RESULTS AND DISCUSSION

The attitude control problem is divided into two parts. First, the environment includes the aircraft simulation, and second, the Q-learning agent which is a controller. In general, at each time step, the θ_{sim} is obtained, and then the calculated tracking error is used to rectify the action selection policy. It is continued for several episodes so as to reach an optimal strategy. The pseudocode of the presented method is as follows:

```
Algorithm 1 Q-learning Attitude Controller
Input: Learning Rate $\alpha$, Discount Factor $\gamma$, Desired Angle $\theta_{des}$ = 1deg.
Output: $Q_{\pi^*}(\theta, \dot{\theta}, \delta_E)$
//Initialize $Q(\theta_0, \dot{\theta}_0, \delta_{E_0}) \leftarrow \emptyset$, for all $\theta \in \mathcal{S}_1, \dot{\theta} \in \mathcal{S}_2, \delta_E \in \mathcal{A}(s)$
for each episode (5 sec) do
    // Initialize 6 DoF simulation with a random $\theta_0 \in [0, 2]$ deg.
    for each time-step (0.01 sec) do
        (1) Select an action $\delta_E$ based on FAA $\epsilon$-greedy strategy.
        (2) Execute 6 DoF simulation using computed $\delta_E$, observe $R_{t+1}$, $\theta_{t+1}$, $\dot{\theta}_{t+1}$.
        (3) Update the state-action value function:
        $Q(\theta_t, \dot{\theta}_t, \delta_{E_t}) = Q(\theta_t, \dot{\theta}_t, \delta_{E_t}) +$
        $\alpha\left[R_{t+1} + \gamma \max_{\delta_E} Q(\theta_{t+1}, \dot{\theta}_{t+1}, \delta_E) - Q(\theta_t, \dot{\theta}_t, \delta_{E_t})\right]$
        (4) Substitute simulation parameters in time-step $t$ with $t+1$.
    end
end
return $Q_{\pi^*}(\theta, \dot{\theta}, \delta_E)$
```

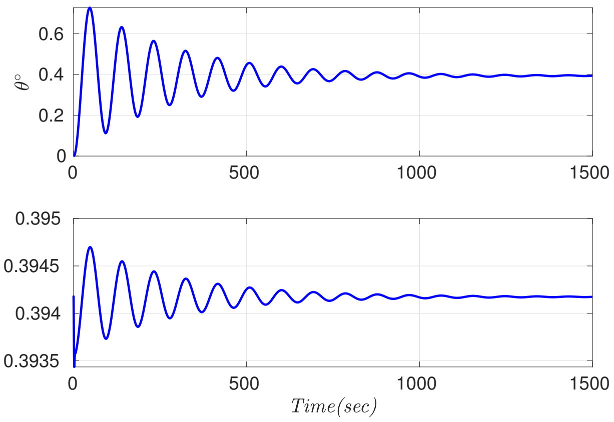

Fig. 2. The α and θ variation in a trim flight

The above method is applied to Chaka 50 MRJ in 5 second episodes. The difference between the proposed control theory and trim flight proves the performance of the Q-learning controller. Figure (2) illustrates the simulation results of trim conditions over 1500 seconds. Needless to say, the α angle is converged exactly to its theoretical value by the table (2). Also, the θ angle is converged in accordance with α. Consequently, the simulation is accurate. But low-stability existence resulting in long-time fluctuations is affected by the damping ratio of the Phugoid mode.

In this way, the simulations using the controller are performed in two problem modelings as MDP and POMDP. The difference between them is observing $\dot{\theta}$ in the MDP model where the POMDP is neglected. Obviously, the state-action table in the MDP model is 3D as $\theta \times \dot{\theta} \times \delta_E$ dimension whereas in the POMDP model, this table is shaped as dimension as $\theta \times \delta_E$. In addition to the number

Table 5. Q-learning Simulation Parameters

Parameter	Value
Epsilon(ϵ)	$[0.1 : 3e{-}6 : 0.04]$
Alpha(α)	$[0.02 : 9e{-}7 : 0.002]$
Gamma(γ)	0.99
Episode number	20000
θ(rad)	$[-10, -0.024 : 0.002 : -0.002, -0.001, 0]$
$\dot{\theta}$(rad)	$[-10, -0.04, -0.02, -0.005]$

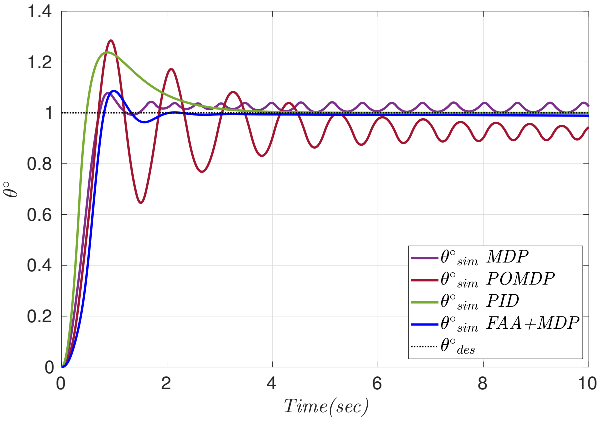

Fig. 3. Performance of improved MDP Q-learning using FAA, in comparison to MDP, POMDP, and PID controller.

of observed states, the bounds, and intervals of states are important in converging time. In this way, to learn the RL controller efficiently, it is momentous to divide θ, $\dot{\theta}$ intervals knowledgeably so as not to omit any main state. As a synopsis, the simulation parameters and the intervals are in table (5) where the ϵ, and α are quantified with their upper bounds in the first episode, and decay at a linear rate during episodes. Also, the state-action table intervals including θ, and $\dot{\theta}$ are mentioned in this table where blue numbers are intervals. These values are considered symmetrically with positive signs for the positive zone. Assuming this, all substantial states that the aircraft confronts during the learning phase are covered. The result of the Q-learning controller is illustrated in figure (3) in comparison to the PID controller. Obviously, the POMDP accuracy is worse than others because the environment modeling is lacking from a complete Markov Property. However, the POMDP performance is significantly better than the trim condition in figure (2). It is apparent that PID rise-time is lower than MDP and POMDP but the overshoot of PID continues to exist until approximately 2 seconds. There is a tiny oscillation in the MDP method because of discrete elevator deflections, but this flaw is eliminated by FAA using the same learned Q-table. Figure (4) shows the rewards of each episode for MDP and POMDP modeling. In early episodes, POMDP results are better and making fewer fluctuations. But after about 4000 episodes, MDP starts achieving positive rewards. The MDP converges fairly in episode number 10000 and surpasses the POMDP method. It is noticeable that the POMDP never achieves positive rewards. Consequently, encompassing the $\dot{\theta}$ plays a significant role in efficiency. Elevator deflections is illustrated in figure (5). Although the control effort of MDP is more than POMDP numerically, it sequels better θ_{does} tracking results. However, the FAA solves the consequent oscillations caused by discrete actions simultaneously reducing control effort even more superior to PID. Finally, figure (6), shows the tracking result for a variable θ. The tracking of a variable pitch angle became possible by defining a virtual desired pitch angle according to the current tracking error of the system. At first glance, it is obvious

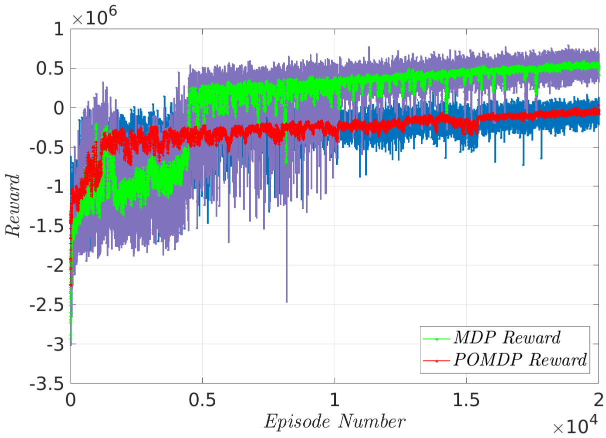

Fig. 4. MDP and POMDP rewards over 20000 episodes

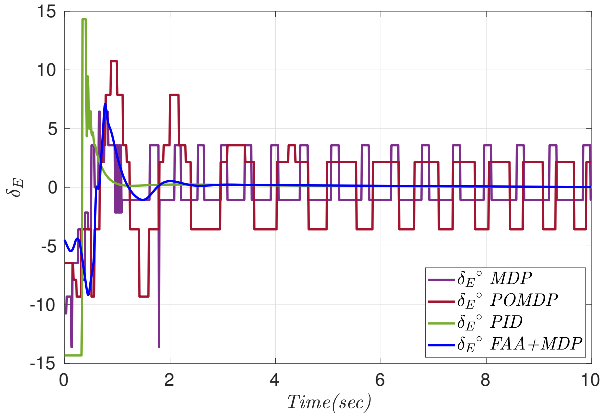

Fig. 5. Elevator deflections in $\theta_{des} = 1°$ tracking

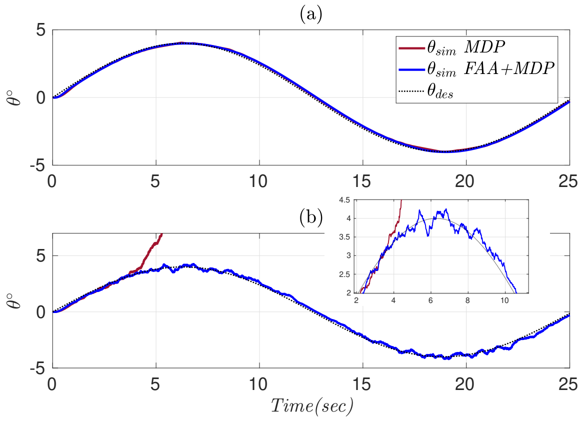

Fig. 6. Variable θ tracking using MDP and FAA-improved MDP a) at normal condition, and b) in presence of sensor measurement noise and atmospheric disturbance.

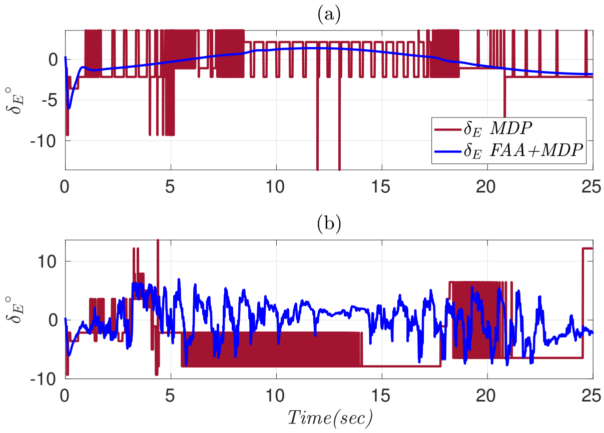

Fig. 7. Elevator deflections in variable θ tracking a) at normal condition, and b) in presence of sensor measurement noise and atmospheric disturbance.

that both MDP and FAA-improved MDP were prosperous to track variable θ in normal conditions. But the elevator's high working frequency of MDP which is shown in figure (6), proves the FAA significant superiority. Apart from that, in presence of atmospheric disturbances and sensor measurement noise that means an actual flight condition, FAA-improved MDP is able to track θ_{des} and demonstrate its robustness. But in figure (6) and its zoom-in, the MDP starts to diverge when the atmospheric disturbance is applied in addition to noise.

5. CONCLUSION

This work proposes a longitudinal attitude control method based on improved Q-learning under MDP and POMDP problem modeling. The aircraft is a novel regional Truss-braced wing airplane that is treated as an RL environment. The action selection strategy is ϵ-greedy, and the reward function defines to consider θ, q, and δ_E. This method is verified in constant and variable θ_{des} tracking simulations where the simulation results show a comparable outcome between the FAA-improved MDP model and the PID controller. Fruitful to conclude that POMDP modeling results are poor due to imperfect problem definition. Finally, the FAA-augmented method was used to construct continuous actions to eliminate fluctuations and make robust variable angle tracking in presence of atmospheric disturbance and sensor measurement noise.

REFERENCES

Bøhn, E., Coates, E.M., Reinhardt, D., and Johansen, T.A. (2021). Data-efficient deep reinforcement learning for attitude control of fixed-wing uavs: Field experiments. *arXiv preprint arXiv:2111.04153*.

Dammen, E.B. (2022). *Reinforcement Learning and Evolutionary Algorithms for Attitude Control, A comparison for aerial vehicles*. Master's thesis.

Emami, S.A., Castaldi, P., and Banazadeh, A. (2022). Neural network-based flight control systems: Present and future. *Annual Reviews in Control*, 53(4), 97–137. doi:10.1016/j.arcontrol.2022.04.006.

Er, M.J. and Deng, C. (2004). Online tuning of fuzzy inference systems using dynamic fuzzy q-learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(3), 1478–1489. doi: 10.1109/TSMCB.2004.825938.

Glorennec, P. and Jouffe, L. (1997). Fuzzy q-learning. In *Proceedings of 6th International Fuzzy Systems Conference*, volume 2, 659–662 vol.2. doi: 10.1109/FUZZY.1997.622790.

Hu, W., Gao, Z., Quan, J., Ma, X., Xiong, J., and Zhang, W. (2022). Fixed-wing stalled maneuver control technology based on deep reinforcement learning. In *2022 IEEE 5th International Conference on Big Data and Artificial Intelligence (BDAI)*, 19–25. IEEE.

Li, L., Bai, J., and Qu, F. (2022). Multipoint aerodynamic shape optimization of a truss-braced-wing aircraft. *Journal of Aircraft*, 1–16.

Nguyen, N.T. and Xiong, J. (2022). Dynamic aeroelastic flight dynamic modeling of mach 0.745 transonic truss-braced wing. In *AIAA SCITECH 2022 Forum*, 1325.

Richter, D.J., Natonski, L., Shen, X., and Calix, R.A. (2022). Attitude control for fixed-wing aircraft using q-learning. In J.H. Kim, M. Singh, J. Khan, U.S. Tiwary, M. Sur, and D. Singh (eds.), *Intelligent Human Computer Interaction*, 647–658. Springer International Publishing, Cham.

Roskam, J. (1998). *Airplane flight dynamics and automatic flight controls*. DARcorporation.

Sarode, V.S. (2022). *Investigating Aerodynamic Coefficients and Stability Derivatives for Truss-Braced Wing Aircraft Using OpenVSP*. Ph.D. thesis, Virginia Tech.

Sutton, R.S. and Barto, A.G. (2018). *Reinforcement learning: An introduction*. MIT press.

Tang, C. and Lai, Y.C. (2020). Deep reinforcement learning automatic landing control of fixed-wing aircraft using deep deterministic policy gradient. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, 1–9. doi:10.1109/ICUAS48674.2020.9213987.

Wang, Z., Li, H., Wu, H., Shen, F., and Lu, R. (2018). Design of agent training environment for aircraft landing guidance based on deep reinforcement learning. In *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, volume 02, 76–79. doi: 10.1109/ISCID.2018.10118.

Watkins, C.J. and Dayan, P. (1992). Q-learning. *Machine learning*, 8(3), 279–292.

Xi, Z., Wu, D., Ni, W., and Ma, X. (2022). Energy-optimized trajectory planning for solar-powered aircraft in a wind field using reinforcement learning. *IEEE Access*, 10, 87715–87732.

Yang, X., Yang, X., and Deng, X. (2020). Horizontal trajectory control of stratospheric airships in wind field using q-learning algorithm. *Aerospace Science and Technology*, 106, 106100. doi: https://doi.org/10.1016/j.ast.2020.106100.

Yuan, X., Sun, Y., Wang, Y., and Sun, C. (2019). Deterministic policy gradient with advantage function for fixed wing uav automatic landing. In *2019 Chinese Control Conference (CCC)*, 8305–8310. doi: 10.23919/ChiCC.2019.8866189.

Zavaree, S., Zahmatkesh, M., Eghbali, K., Zahiremami, K., Vaezi, E., Madani, s.A., Kariman, A., Heidari, Z., Mahmoudi, A., Rassouli, F., Varshavi, M., and Sadraey, M.H. (2021). Modern regional jet family (chaka: A high-performance, cost-efficient, semi-conventional regional jet family). doi:10.13140/RG.2.2.20347.64802.

Zipfel, P. (2014). Modeling and simulation of aerospace vehicle dynamics–third edition.

Çetin, E. (2018). *System identification and control of a fixed wing aircraft by using flight data obtained from x-plane flight simulator*. Master's thesis, Middle East Technical University.