International Forum on Aeroelasticity and Structural Dynamics
IFASD 2024
17-21 June 2024, The Hague, The Netherlands

STALL FLUTTER SUPPRESSION WITH ACTIVE CAMBER MORPHING BASED ON REINFORCEMENT LEARNING

Li Jinying[1], Dai Yuting[1,2], Yang Chao[1]

[1] Beihang University
Xueyuan Road 37, Beijing, China
[2] Tianmushan Laborotary
Shuanghongqiao Road 116, Hangzhou, China
lijinying@buaa.edu.cn
yutingdai@buaa.edu.cn
yangchao@buaa.edu.cn

Keywords: Stall flutter, reinforcement learning, flexible wing

Abstract: This study investigates the adaptation of reinforcement learning into stall flutter suppression. The geometric model is a NACA0012 airfoil with active trailing edge morphing. Firstly, an offline, rapid responsive stall flutter environment is constructed with differential equations, where the aerodynamic force is predicted with reduced-order models. A double-Q-network (DQN) algorithm is adapted to train the controlling agent with the proposed offline environment. The agent has 5 optional actions with different amplitudes and directions of morphing. The reward function is designed with a linear combined punishment of pitching angle and angular velocity, a large bonus reward on complete suppression, and a large punishment on over-limit morphing. The trained agent shows a rapid and complete stall flutter suppression performance in offline environment simulation, where different sets of observations and scores are discussed.

1 INTRODUCTION

Stall flutter is an aeroelastic phenomenon involving self-excited vibration caused by fluid-structure interaction, which leads to fatigue or even damage to wing structure[1]. In traditional aircraft design, the prevention of stall flutter leads to large limitations on weight and speed[2]. Therefore, effective, and efficient stall flutter control is a promising area to improve the aircraft's performance. However, the high dimension, nonlinearity, and unsteadiness characteristic of stall flutter make it hard to predict and control. Recent studies have brought jet actuators[3][4], flaps[5][6], and active camber morphing[7] into stall flutter and related dynamic stall control, where the active camber morphing on trailing edge demonstrates a good performance on high-fidelity computation and provides a clean profile of aerodynamics[8].

In recent years, the rise of data-driven methods has brought an inspiring perspective on aerodynamics topics. Nicola et al[9] use surrogated dynamic model decomposition and reduced order model for aeroelastic prediction and control. Jung et al[10] combine state-consistence enforcement with a generic algorithm for rapid aeroelastic analysis. Dai et al[11] accurately predicted the stall flutter boundaries and aerodynamics with gated recurrent unit neural networks.

Among the thriving data-driven methods, reinforcement learning (RL), especially deep reinforcement learning (DRL) shows its outstanding ability in complex model prediction, directness, and generalization ability[12]. Fruitful RL studies have been done on flow control and shape optimization. Viquerat et al[13] use DRL for shape optimization with various numbers of control points. Wang et al[14] developed an RL tool implanted into the computation fluid dynamics (CFD) platform OpenFOAM. Another research group Wang et al[15] uses DRL in flap foil control and further analyzed the flow characteristics based on the smart controller. However, RL application in the aeroelastic field, especially in stall flutter, is still rare, which is an inspiring direction to explore.

A DRL-based smart controller on active trailing edge morphing is designed in this paper with reinforcement learning methods to suppress stall flutter. The aeroelastic system of the morphing airfoil is first proposed in Section 2, followed by an introduction of the reduced-order model and reinforcement learning algorithm used in this paper. In Section 3, the single-input single-output nonlinear model of stall flutter is constructed with the reduced-order model. Then, a smart controller is learned with reinforcement learning under the environment of the proposed nonlinear stall flutter model. The selection of important learning inputs and parameters are detailly discussed to provide preferably control performances.

2 METHODOLOGY

2.1 Geometric model of morphing airfoil

The geometric model of the studied NACA0012 airfoil is shown in Fig. 1. The airfoil has one domain of freedom on pitch and an active morphing trailing edge. The freestream velocity U_∞ is 8 m/s. A body coordinate O_b, x_b, y_b and global coordinate O, x, y are used for the convenience of expression. The origin of the global coordinate is positioned at the leading edge of static NACA0012, whose angle of attack is 0 rad. The rotation center of pitch motion is $x_b = 0.35c$. The trailing edge is defined as $x_b > 0.7c$, with a maximum trailing edge morphing amplitude of $|y_b| < 0.1c$, which is equivalent with $\max\{\tan\theta_{tem}\} = 3$.

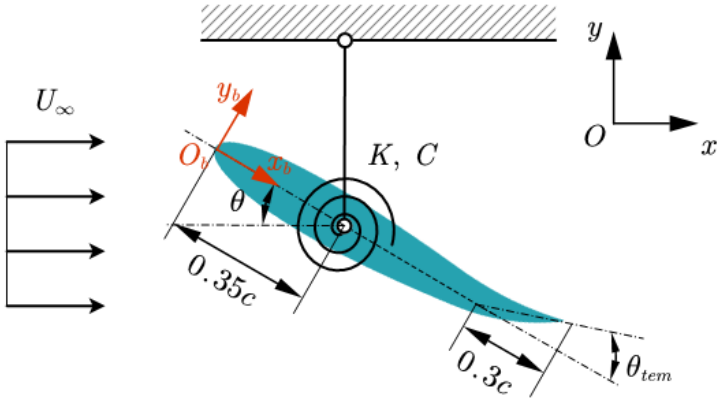

Fig. 1 Aeroelastic model of morphing NACA0012

The motion equation considering 1DoF pitch and active morph is:

$$I\ddot{\theta} + C\dot{\theta} + K\theta = M \tag{1}$$

where $I = 0.001,\ C = 6\%,\ K = 0.3$ are the mass inertia, damping ratio, and stiffness of pitch, and M is the moment force due to both pitch and morph motion:

$$M = f(\theta, \theta_{tem}) \tag{2}$$

For the convenience of calculation, the angle of the trailing edge morph θ_{tem} is replaced into a tangent formation:

$$\beta = 3\tan\theta_{tem} \tag{3}$$

where the morph factor β is valued between $[-1,1]$, in accordance with the maximum trailing edge morphing amplitude. It is worth mentioning that Eq.(2) is processed with a linear simplification, where the moment force of compounded pitch and morph motion is approximated into the sum of moment forces due to separated motions:

$$M = M_\theta + M_\beta \tag{4}$$

2.2 reduced order model: NARX neural network

A key difficulty in stall flutter control is the computation of moment force M in Eq.(2). Even with the simplification of Eq.(4), the computation is still very time-consuming with CFD. Therefore, a reduced order model is built to predict M during pitch and morph motions in this paper. The NARX neural network is a time-delay feedforward neural network, which is a commonly used model for nonlinear dynamic systems. The NARX neural network is selected to model the moment force M in this paper, because it provides both adequate accuracies in the prediction of long-term nonlinear systems and rather lower training cost at the same time, compared with other recurrent neural networks such as long short term memory networks. The NARX network can be mathematically expressed as:

$$O(t) = g\{O(t - N_o dt), ..., O(t - dt), u(t - N_u dt), ..., u(t - dt), u(t)\} \tag{5}$$

where t and dt are the time and time delay of the neural network, $O(t)$ and $u(t)$ are the output and input of the neural network, N_O and N_u are the number of time delays inputted into the network. The time delay is 0.005s and the number of delays is 10 in this paper. The structure of a NARX neural network is shown in Fig. 2 below.

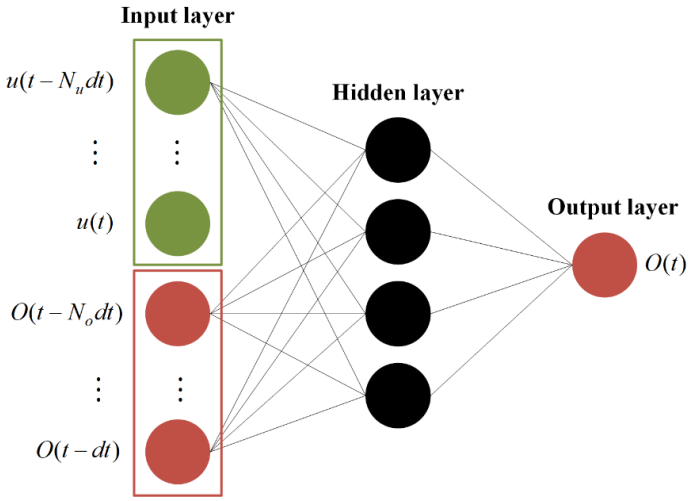

Fig. 2 The structure of a NARX neural network

2.3 reinforcement learning: DQN

With the proposed fast responsive stall flutter moment prediction, the control with trailing edge morph can be expressed as a single input multiple output (SISO) system:

$$\{\dot{\theta},\theta\} = g(\beta) \tag{6}$$

The system inputs the trailing edge morph factor β, and directly outputs the aeroelastic responses $\dot{\theta},\theta$, the SISO system with a smart controller is presented in Fig. 3.

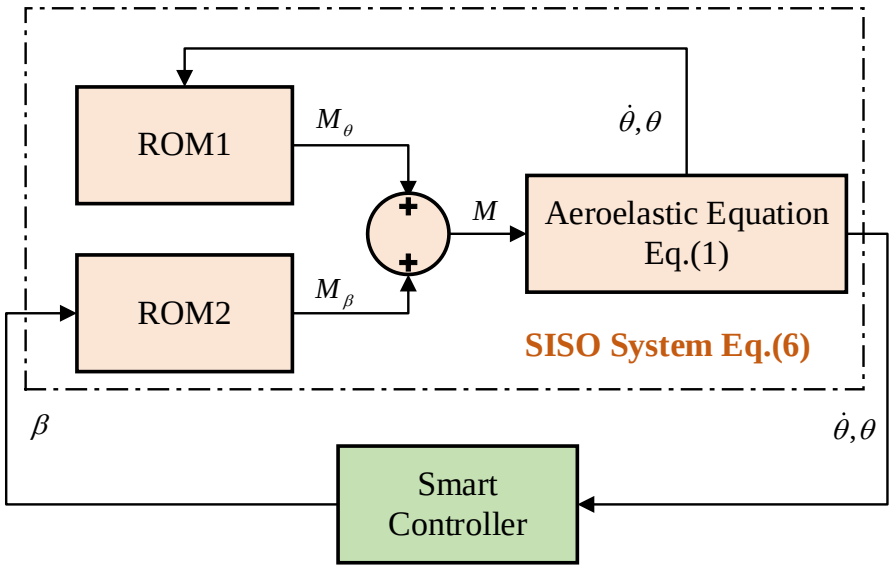

Fig. 3 The SISO system with control

Regardless of the internal operations of the SISO system, the smart controller perceives the aeroelastic response $\dot{\theta},\theta$ and decides on a subsequent trailing edge morph factor β. The ultimate purpose of the decision on trailing edge morph is to suppress the aeroelastic response to static, which can be expressed as $\dot{\theta}=0$ rad/s, $\theta=0$ rad. This concise observation and decision process

makes reinforcement learning a compatible algorithm for designing the controller. The process of reinforcement learning is an imitation of human learning, as shown in Fig. 4 below.

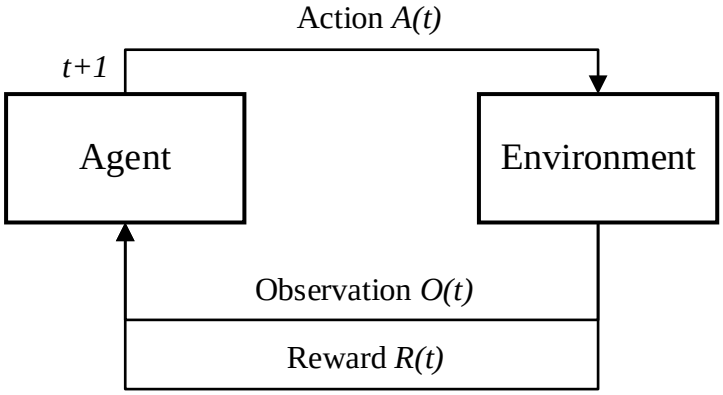

Fig. 4 The process of the reinforcement learning loop

The agent in the RL algorithm takes an action $A(t)$ in a single step based on observation of the last timestep $O(t-dt)$, existing knowledge of the agent, and a percentage of exploration. Then, the environment responds to the action, which can be observed by the agent, and gives a certain reward $R(t)$ based on the observation $O(t)$ from the last action. The knowledge of the agent is updated by aiming at maximizing an overall profit:

$$R_{\infty} = \sum_{n=0}^{\infty} \gamma^n R(t_0 + nt) \tag{7}$$

In this paper, the smart controller acts as the agent and is modeled through a dense neural network, and the SISO system is regarded as the environment. Another significant factor in RL is the modeling and learning strategy of the agent, where we use the Deep-Q-Network (DQN) algorithm. DQN is a classic strategy for RL in finite discrete space, which is widely used for its preferable stability and efficiency. For the brevity of the article, readers can refer to [16] for the mathematical details of DQN.

3 RESULTS AND DISCUSSION

In this section, the SISO system is first constructed with reduced order modeling and validated with predictions on stall flutter. Then, the SISO system is used for RL, where the parameter influence is discussed, including training hyperparameters, reward function, and observation.

3.1 Stall flutter prediction by SISO system

As introduced in Eq.(4), the compounded moment force is separately modeled by 2 ROMs: ROM1 for pitch motion and ROM2 for morph motion. The training data of ROM1 and ROM2 are obtained through CFD computation with given compounded sine motion:

$$\begin{aligned}
&\theta(t) = \sum_{i=1}^{6} A_i \cos \omega_i t \\
&\beta(t) = \sum_{i=1}^{6} B_i \sin \omega_i t \\
&\omega = [5\pi \quad 5.4\pi \quad 5.8\pi \quad 6.2\pi \quad 6.6\pi \quad 7\pi] \\
&s.t. \sum_{i=1}^{6} A_i < 1, \ \sum_{i=1}^{6} B_i < 1
\end{aligned} \tag{8}$$

where A_i and B_i are randomly distributed amplitudes, ω_i is the corresponding angular velocity. The identification results of ROM1~2 are shown in Fig. 5. The identification errors of ROM1 and ROM2 are 9.19% and 7.33%, respectively. Based on trained ROM1~2, the time response of a simplified nonlinear system under the freestream velocity of 8m/s is predicted, as shown in Fig. 6. The prediction results present a typical stall flutter phenomenon in a time of 5 seconds, with a pitching amplitude of 0.63 rad, which is consistent with experiment results[17].

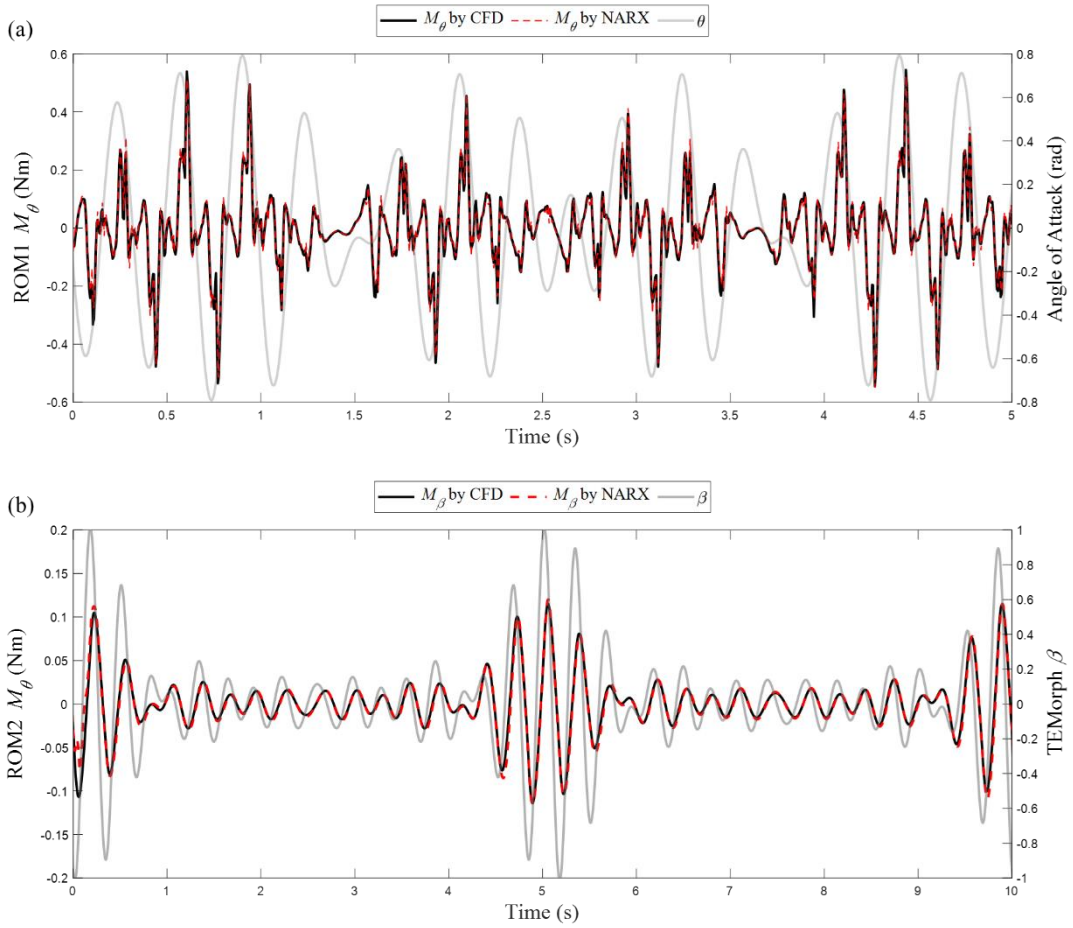

Fig. 5 Identification results of ROM1 and ROM2

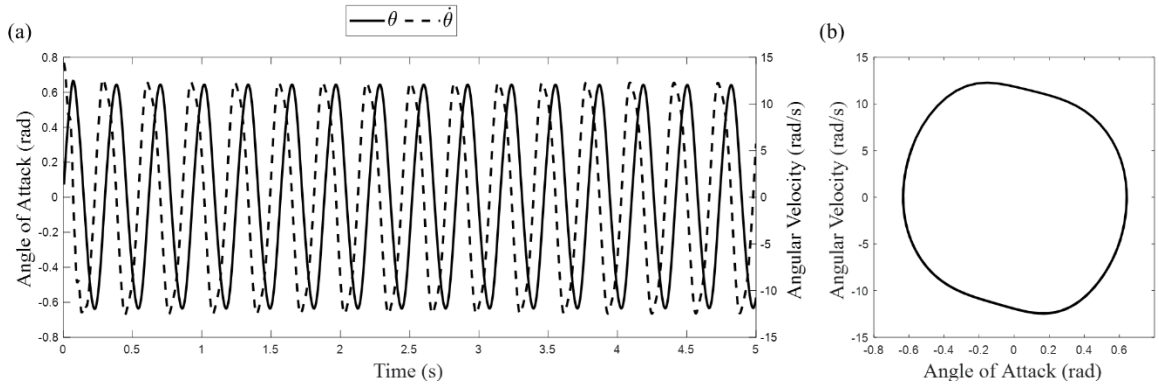

Fig. 6 Time-domain response predicted by a simplified nonlinear system

3.2 stall flutter suppression with DQN

3.2.1 Reward and Action

As shown in Fig. 4, the environment and agent are the SISO system and smart controller, respectively. To train the smart controller, the function for action $A(t)$, observation $O(t)$ and reward $R(t)$ need to be cautiously designed for better performance and efficiency.

An intuitive choice for action $A(t)$ is the trailing edge morph factor β. However, out of the exploration nature of the RL algorithm, the deviation of morph $\Delta\beta_t = \beta_{t+1} - \beta_t$ can be very large. Although rapidly changing morph factor can be responded to by the SISO system, it is not realistic considering further application into experiments or high-fidelity simulation due to the limitation on actuator or mesh deformation. Therefore, the action $A(t)$ of the smart controller is converted into a discrete formation here considering both the requirement of DQN and avoidance of unpractical dramatic morphing. The action is described as:

$$A(t) = \begin{cases} -\Delta\beta \\ 0 \\ \Delta\beta \end{cases} \tag{9}$$

$$\Delta\beta = 0.05 \text{ or } 0.03$$

Eq.(9) converts the difficult value selecting problem considering continuity into a limited selection of three simple actions: upward 0.05 rad, downward 0.05 rad or just stay static.

Another concern for reinforcement learning is the reward function, which evaluates the observation from action and is very influential on learning efficiency and effectiveness. Therefore, the reward function $R(t)$ is non-unique and often cautiously chosen. To suppress a pitching airfoil under stall flutter to static, it is natural to infer that a static state $\dot{\theta} = 0, \theta = 0$ should be rewarded and large $\dot{\theta}, \theta$ should be punished. The main situations are:

Table 1 Main situations of controlled stall flutter

Case No.	Responses from SISO system	State of stall flutter	Reward function

(i)	$\dot{\theta}=0,\theta=0$	Suppressed to static	$R(t)>0$
	$\lvert\dot{\theta}\rvert<\delta,\lvert\theta\rvert<\varepsilon$	About to static	
(ii)	$\lvert\dot{\theta}\rvert\geq\delta,\lvert\theta\rvert<\varepsilon$	About to equilibrium	$R(t)=h(\lvert\theta\rvert,\lvert\dot{\theta}\rvert)<0$
	$\lvert\dot{\theta}\rvert<\delta,\lvert\theta\rvert\geq\varepsilon$	About to peak	
	$\lvert\dot{\theta}\rvert\geq\delta,\lvert\theta\rvert\geq\varepsilon$	Quick pitching stall flutter	

Apart from ineffective suppression case (ii), the morphing beyond limitation is also not acceptable, which is considered as Case No. (iii): $\lvert\beta\rvert>0.1c$, which should bring a large punishment $R(t)<0$ regardless of the control effect. Therefore, the reward function is formatted as:

$$R(t)=\begin{cases}20 & \text{(i)}\\ -\lVert\theta_t\rVert-0.1\lVert\dot{\theta}\rVert & \text{(ii)}\\ -100 & \text{(iii)}\end{cases} \tag{10}$$

3.2.2 Observation

In the DQN algorithm, the observation vector is used as the input of the trained dense neural network, which is the smart controller. There are many alternative observable parameters from the SISO system, such as the angle of attack θ_t, angular velocity $\dot{\theta}_t$, airspeed $U_{\infty,t}$, trailing edge morph factor β, action $A(t)$, moment forces $M_{\theta,t}, M_{\beta,t}$, time t etc. The first observation vector is chosen as:

$$O_{b,1}=[\theta_t,\dot{\theta}_t,\beta_{t-dt},t] \tag{11}$$

The control performance with $O_{b,1}$ is shown in Fig. 7. Control starts at $t=1$s, and the stall flutter is stimulated by trailing edge morphing and then suppressed. The result shows that the designed smart controller successfully achieved the goal of suppressing static. However, the trailing edge morph factor β signal is hard to explain in physics with $O_{b,1}$ since periodic characteristics can only be observed in the first two periods after control. When $t>2$s, the trailing edge morph factor continuously decreases, which is inconsistent with the periodicity exhibited by aeroelastic responses $\dot{\theta},\theta$. The inconsistency is caused by the time-domain accumulative error of ROMs, which is also learned by the smart controller and covers the period information.

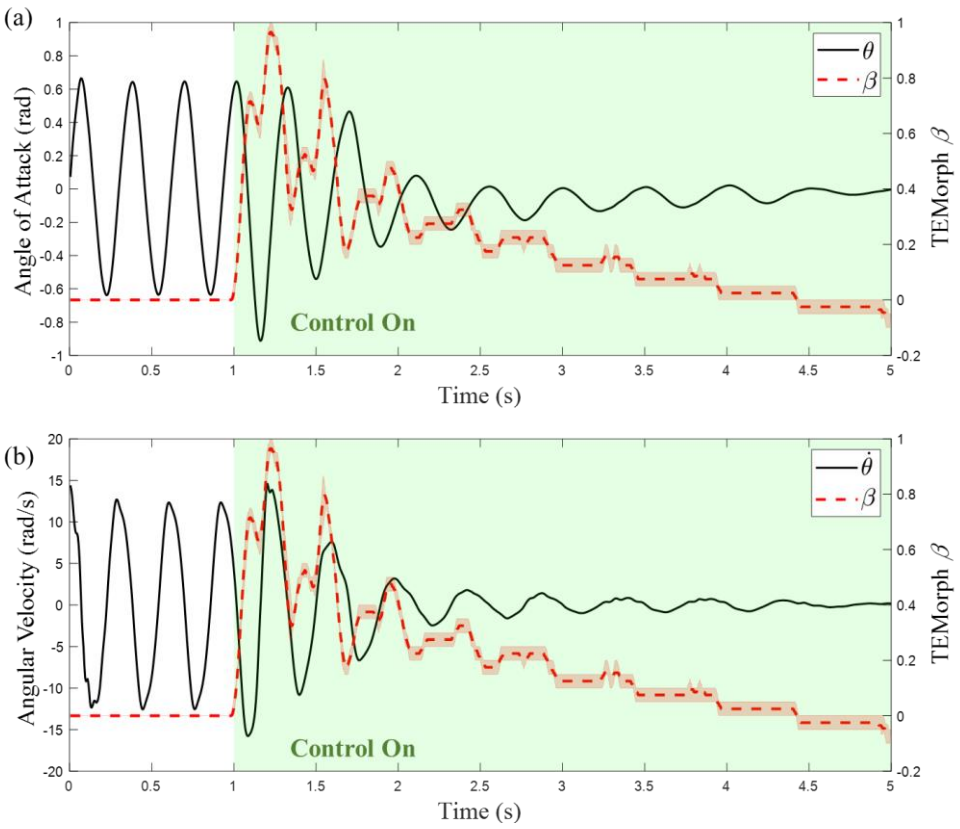

Fig. 7 Smart control with time observation: (a) angle of attack; (b) angular velocity

To avoid learning the cumulative error of ROMs, the second observation vector is adapted without time information:

$$O_{b,2} = [\theta_t, \dot{\theta}_t, \beta_{t-dt}] \tag{12}$$

During a learning episode, the agent cannot access the control length and concentrate on aeroelastic parameters. The control performance with $O_{b,2}$ is shown in Fig. 8. The stall flutter is also completely suppressed by the smart controller, while better periodic performance is observed in the trailing edge morph factor. The result indicates that removing time observation effectively reduced the impact of cumulative error from ROMs, and the $O_{b,2}$ is further adapted in the following discussions.

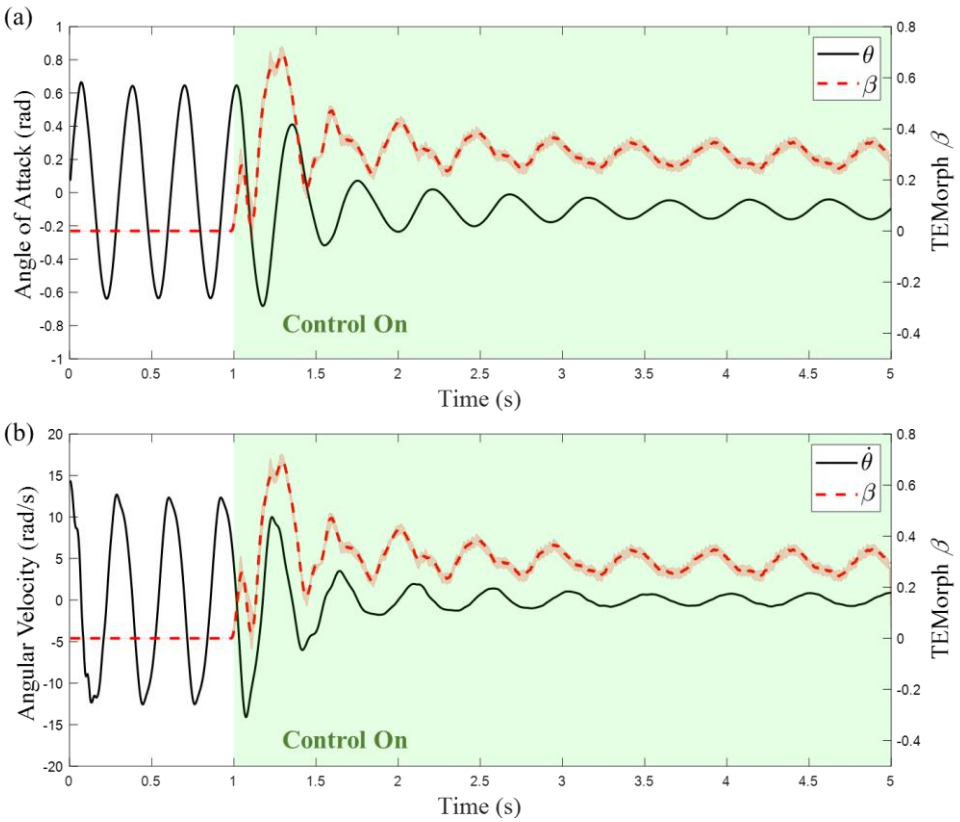

Fig. 8 Smart control without time observation: (a) angle of attack; (b) angular velocity

3.2.3 Score and episode

The hyperparameters are also influential on training efficiency and effectiveness. However, most of the parameters are chosen on experience. In the DQN training of this paper, a key factor to decide is the balance between training episodes and scores. In normal cases, more learning episodes bring higher learning scores, but the SISO system in this paper, as discussed in section 3.2.2, is a low-fidelity environment, which may lead to even higher errors with overlearning.

We consider a total static airfoil throughout the control period as a perfect situation, which means the controller will get a full reward score:

$$R_{max} = 20 \cdot \frac{T}{dt} = 20 \cdot \frac{4}{0.005} = 16000 \tag{13}$$

Three stages of the score are chosen to check the learning performance: 6000 of 37.5% full score, 10000 of 62.5% full score, and 12000 of 75% full score. The control effect of these cases is shown in Fig. 9~Fig. 11.

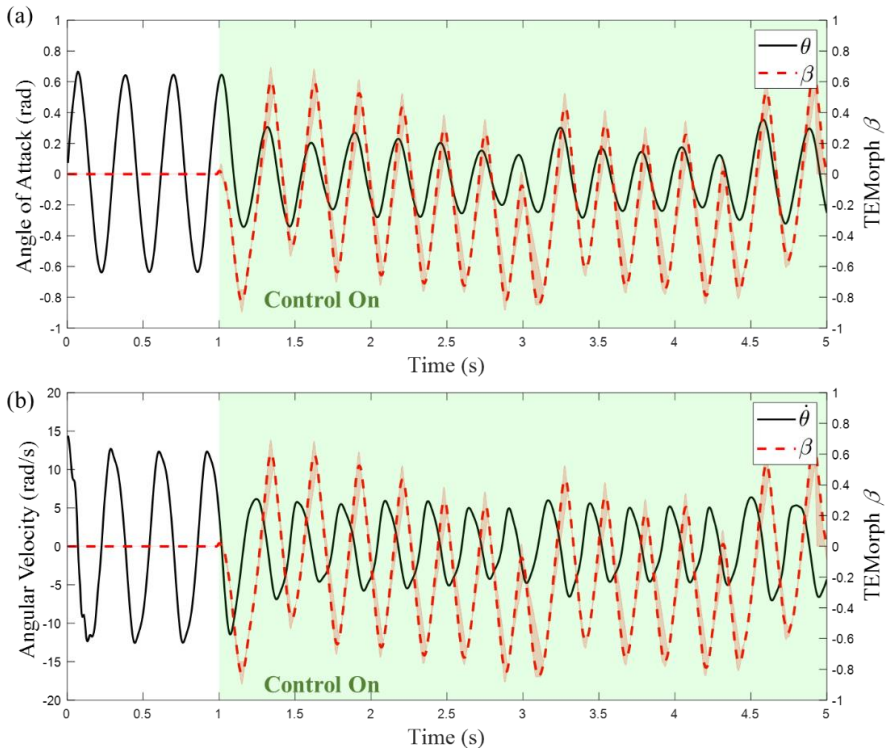

Fig. 9 smart control with score 6000: (a) angle of attack; (b) angular velocity

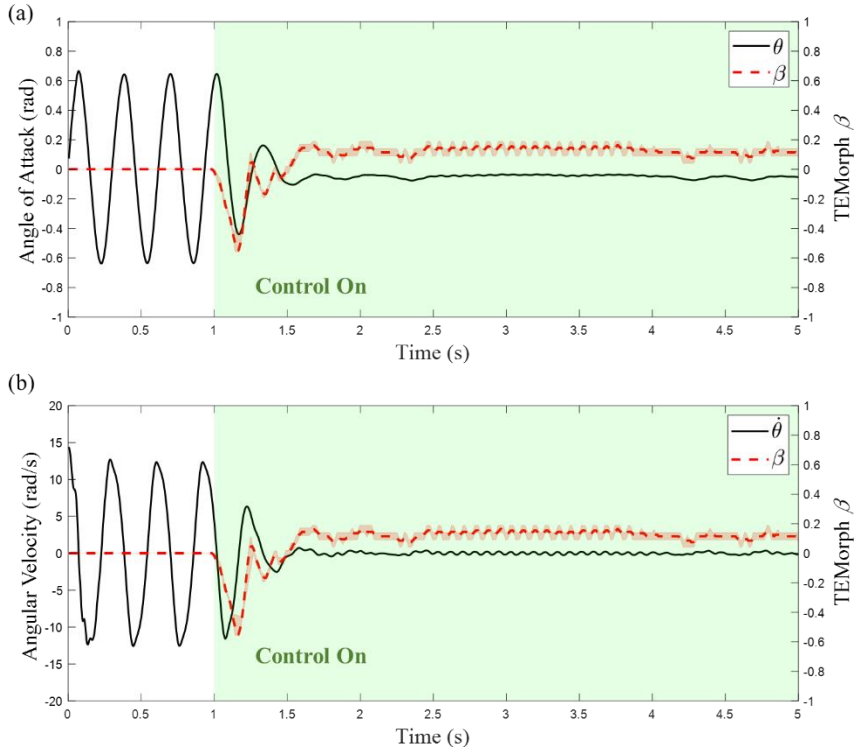

Fig. 10 smart control with score 10000: (a) angle of attack; (b) angular velocity

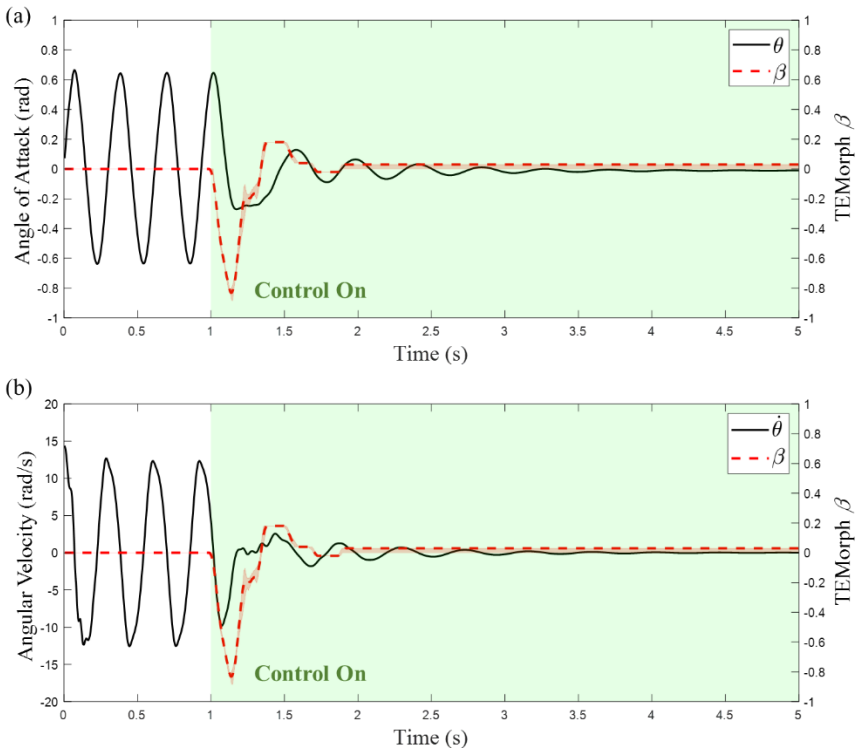

Fig. 11 smart control with score 12000: (a) angle of attack; (b) angular velocity

The periodic controlling effect is also clearly demonstrated in the case of the 6000 score, but the stall flutter is not completely suppressed, which presents an insufficient learning of the smart controller. In cases of 10000 and 12000 scores, quick suppression is both achieved within 3 pitching periods after control. However, in the case of 10000, there are still small deviations of trailing edge morph even after effective control, as shown in Fig. 10 ($t > 2.5\,\mathrm{s}$). the fluctuation of the trailing edge also leads to a high-frequency pitching of airfoil. When the score achieves 75% of the full score, the smart controller learns to stabilize itself after exiting the stall flutter, where both the trailing edge and airfoil present a complete static stage.

Cases with even higher score goals are also tested, but the training process shows that the increase of score becomes very slow after it achieves 1200 and it will be very time-consuming since it is impossible to achieve a perfect situation of instantaneous suppression. Therefore, the results prove that a 75% percentage of perfect situation can lead to a better learned smart controller, where the knowledge from the longer learning process overweighs the fidelity error of the SISO system.

4 CONCLUSIONS

In this paper, a smart agent is trained with the reinforcement learning algorithm Deep-Q-Network, based on a reduced order single-input single-output stall flutter system. The conclusions are as follows:

(1) Both reduced order model built for aerodynamic force computation identifies moment force with errors less than 10%. The SISO system based on ROMs is validated with an experiment, which shows adequate accuracy for control law design.

(2) The smart controller learned with DQN is tested with 2 sets of observations. Both observation cases completely suppressed stall flutter, and set without time observation shows better periodic characteristics.
(3) The smart controller learned with DQN is tested with 3 sets of scores. The results show that a higher score goal leads to better performance, and the goal of 75% perfect situation gains the best performance, which quickly suppresses stall flutter and maintains the static of both the trailing edge and airfoil.

REFERENCES

[1] Dimitriadis, G., and Jing, L., "Bifurcation Behavior of Airfoil Undergoing Stall Flutter Oscillations in Low-Speed Wind Tunnel", *AIAA Journal*, Vol.47, No.11, 2009, pp. 2577-2596
[2] Deman, T., and Dowell, E.H., "Experimental and Theoretical Study on Aeroelastic Response of High-Aspect-Ratio Wings", *AIAA Journal*, Vol.39, No.8, 2001, pp. 1430-1441.
[3] Hanns, F.M., Christoph, S., Christian, N.N., Christian O.P., and David, G., "Control of Thick Airfoil, Deep Dynamic Stall Using Steady Blowing", *AIAA Journal*, Vol.53, No.2, 2015, pp.277-295.
[4] Zhen, C., Zhiwei, S., Sinuo, C., and Zhangyi, Y., "Stall Flutter Suppression of NACA0012 Airfoil Based on Steady Blowing", *Journal of Fluids and Structures*, Vol.109, Feb. 2022, 103472.
[5] Gerontakos, P., and Lee, T., "Dynamic Stall Flow Control via a Trailing-Edge Flap", *AIAA Journal*, Vol.44, No.3, 2006, pp. 469-480.
[6] Xing, S.L., Xu, H.Y., and Zhang, W.G., "Trailing Edge Flap Effects on Dynamic Stall Vortex and Unsteady Aerodynamic Forces on a Pitching Airfoil", *International Journal of Aerospace Engineering*, published online, Mar. 2022.
[7] Wu, Y., Dai, Y., and Yang, C., "Time-Delayed Active Control of Stall Flutter for an Airfoil via Camber Morphing", *AIAA Journal*, Vol.60, No.10, 2022, pp. 5723-5734.
[8] Jiaying, Z., Alexander, D.S., Chen, W., Huaiyuan, G., Mohammadreza, A., Michael, I.F., and Benjamin, K.S.W., "Aeroelastic Model and Analysis of Active Camber Morphing Wing", *Aerospace Science and Technology*, Vol.111, Apr. 2021, p. 106534.
[9] Nicola, F., Steven, L.B., and Urban, F., "Data-Driven Modeling for Transonic Aeroelastic Analysis", *Journal of Aircraft*, Vol.61, No.2, 2024, pp. 625-637
[10] Jung, I.S., Yi, W., Alessandro, B., and Andrew, K., "Genetic-Algorithm-Guided Development of Parametric Aeroelastic Reduced-Order Models with State-Consistence Enforcement", *AIAA Journal*, Vol.61, No.9, 2023, pp. 3976-3994
[11] Dai, Y.T., Rong, H.R., Wu, Y., Chao, Y., and Xu, Y.T., "Stall flutter prediction based on multi-layer GRU neural network", *Chinese Journal of Aircraft*, Vol.36, No.1, 2023, pp. 75-90
[12] Vignon, C., Rabault, J., Vinuesa, R., "Recent advances in applying deep reinforcement learning for flow control: Perspectives and future directions", *Physics of fluids*, Vol.35, No.3, 2023
[13] Viquerat, J., Rabault, J., Kuhnle, A., Ghraieb, H., Larcher, A., and Hachem, E., "Direct shape optimization through deep reinforcement learning", *Journal of Computational Physics*, Vol428, 2021: 110080.
[14] Wang, Q., Yan, L., Hu, G., Li, C., Xiao, Y.Q., Xiong, H., Rabault, J., and Noack, B.R., "DRLinFluids: An open-source Python platform of coupling deep reinforcement learning and OpenFOAM", *Physics of Fluids*, Vol.34, No.8, 2022

[15] Wang, Z.P., Lin, R.J., Zhao, Z.Y., Chen, X., Guo, P.M., Yang, N., Wang, Z.C., and Fan, D.X., "Learn to Flap: Foil Non-parametric Path Planning via Deep Reinforcement Learning", *Journal of Fluid Mechanics*, Vol.984, No.A9, 2024

[16] Mnih, V., Kavukcuoglu, K., Silver, D., et al. "Human-level control through deep reinforcement learning", *Nature*, Vol.518, No. 7540, 2015, pp. 529-533

[17] Goyaniuk, L., Poirel, D., and Benaissa, A., "Pitch-Heave Symmetric Stall Flutter of a NACA0012 at Transitional Reynolds Numbers", *AIAA Journal*, Vol.58, No.8, 2020, pp.3286-3298.

COPYRIGHT STATEMENT