Closed-loop Flow Control Method Based on Deep Reinforcement Learning using a Co-flow Jet

Y. R. Zhao, H. Y. Xu† and Z. Y. Xie

National Key Laboratory of Science and Technology on Aerodynamic Design and Research, Northwestern Polytechnical University, Xi’an, 710072, China

†Corresponding Author Email: xuheyong@nwpu.edu.cn

ABSTRACT

A closed-loop control framework is developed for the co-flow jet (CFJ) airfoil by combining the numerical flow field environment of a CFJ0012 airfoil with a deep reinforcement learning (DRL) module called tensorforce integrated in Python. The DRL agent, which is trained through interacting with the numerical flow field environment, is capable of acquiring a policy that instructs the mass flow rate of the CFJ to make the stalled airfoil at an angle of attack (AoA) of 18 degrees reach a specific high lift coefficient set to 2.0, thereby effectively suppressing flow separation on the upper surface of the airfoil. The subsequent test shows that the policy can be implemented to find a precise jet momentum coefficient of 0.049 to make the lift coefficient of the CFJ0012 airfoil reach 2.01 with a negligible error of 0.5%. Moreover, to evaluate the generalization ability of the policy trained at an AoA of 18 degrees, two additional tests are conducted at AoAs of 16 and 20 degrees. The results show that, although using the policy gained under another AoA cannot help the lift coefficient of the airfoil reach a set target of 2 accurately, the errors are acceptable with less than 5.5%, which means the policy trained under an AoA of 18 degrees can also be applied to other AoAs to some extent. This work is helpful for the practical application of CFJ technology, as the closed-loop control framework ensures good aerodynamic performance of the CFJ airfoil, even in complex and changeable flight conditions.

1. INTRODUCTION

The co-flow jet is an active flow control (AFC) method that was first developed by Zha and Paxton (2004). The method is to create an injection slot near the leading edge and a suction slot near the trailing edge to form a jet channel on the upper surface of the airfoil, as shown in Fig. 1.

A high-energy jet is injected through the injection slot to the flow field along the surface of the airfoil, and the same amount of mass flow is withdrawn into the suction slot. With the injection of high-energy jets into the external flow field, the energy of the boundary layer of the suction surface surges; that is, strong jet mixing and energy exchange occur with the main flow to ensure that the energy of the boundary layer is sufficient to overcome the inverse pressure gradient at high AoAs, thereby inhibiting flow separation. The streamline of the baseline airfoil and CFJ airfoil for NACA2415 is shown in Fig. 2.
Since the concept of the CFJ was proposed, researchers have developed the technique from the mechanism study of 2D CFJ airfoils (Zha et al., 2006b) to the parametric study of CFJ airfoils with different shapes and various freestream conditions (Wang et al., 2008; Lefebvre & Zha, 2014; Ma & Xu, 2022; Ma et al., 2023). Using different study methods, such as numerical simulation (Wang & Zha, 2011; Gan et al., 2013), wind tunnel tests (Zha et al., 2006a; Zha et al., 2007a; Zha et al., 2018) and application research (Yang & Zha, 2019; Lei & Zha, 2021; Xu & Zha, 2021), CFJ technology has been demonstrated to have potential in lift enhancement and drag reduction with low energy consumption. However, despite years of development, CFJ technology is still an open-loop control method, which implies that the control strategy is fixed and nonadjustable. This approach is not sufficient to effectively address the complex and volatile nature of practical flight situations. Therefore, developing closed-loop methods for CFJ technology is significant.

There are many closed-loop control methods that can be selected for research. In general, closed-loop control methods can be divided into two categories: traditional PID control and model predictive control (MPC) (Franklin et al., 2019). Although PID control is widely used in engineering for its mature and simple technology, adjusting the PID parameters is still challenging, as it requires prior experience and multiple trials to acquire satisfactory parameters. Additionally, the control effect of PID is not ideal for high-dimensional and nonlinear systems. MPC approaches are better suited to solve nonlinear and high-dimensional control problems (Siegel et al., 2003; Glauser et al., 2004; Ahuja & Rowley, 2010), but they are comparatively difficult to implement. This is because researchers need to observe the entire system to derive the reduced-order model (ROM), which requires large amounts of data from experiments or numerical simulations and can restrict the feasibility of the method.

In the field of artificial intelligence (AI), a method called reinforcement learning (RL) is capable of finding control strategies automatically by interacting with the environment that needs to be controlled. The method has been used for solving low-dimensional problems since the 1980s (Sutton, 1988). Since the performance of computers has been greatly enhanced, deep learning has become a popular machine learning method using artificial neural networks (ANNs), which have worked well in perceptual problems such as speech and image recognition since 2012 (Krizhevsky et al., 2017). Therefore, deep reinforcement learning (DRL) was devised by combining deep learning and RL to utilize the great ability of ANNs in data representation (Akhiyik & Yavuz, 2021; Moshtagzadeh & Aligoodarz, 2022). The DRL method has been widely applied in different domains, including games, such as Atari games (Mnih et al., 2013) and AlphaGo (Silver et al., 2016), industrial applications, such as unmanned vehicles (Kiran et al., 2022), and the field of robot control (Gu et al., 2017).

Recently, researchers in the field of flow control have shown interest in utilizing DRL technology in AFC systems due to its ability to handle problems with nonlinearity and high dimensionality. Two typical objects were mainly discussed: 2D flow past circular cylinders and flow past airfoils. By using various DRL algorithms, such as proximal policy optimization (PPO) and deep Q-network (DQN), control strategies for different purposes, i.e., drag reduction (Rabault et al., 2019; Rabault & Kuhnle, 2019; Tang et al., 2020; Ren et al., 2021a; Paris et al., 2021) and hydrodynamic stealth (Ren et al., 2021b) of a circular cylinder and flow separation control of an airfoil (Shimomura et al., 2020), were successfully developed. Rabault et al. (2019) used the PPO algorithm to search for the best mass flow rate of jets implemented symmetrically on the upper and lower surfaces of a circular cylinder to reduce drag. This was the first time that DRL was used to perform AFC in a simple computational...
fluid dynamics (CFD) simulation, where the Reynolds number was relatively low. To speed up obtaining control strategies, Rabault and Kuhnle (2019) were inspired by parallel numerical simulation and developed an advanced version of the method discussed before. They achieved this by adapting the DRL algorithm for parallelization, which involved running several independent simulations in parallel to gather experiences more rapidly. To further improve this work, several research studies have been conducted to enhance the robustness of DRL methods (Tang et al., 2020; Ren et al., 2021a) and optimize the sensor layout (Paris et al., 2021). For the purpose of hiding traces of bluff bodies, Ren et al. (2021b) used the DRL algorithm to find the control policy of windward-suction-leeward-blowing actuators implemented on a bluff body. As a result, compared to the uncontrolled bluff body, the velocity deficit in the wake of the controlled bluff body was significantly reduced. To control the flow past 2D airfoils, Shimomura et al. (2020) carried out the first trial. They used a dielectric barrier discharge plasma actuator (DBDPA) installed on the upper surface of an NACA0015 airfoil to prevent flow separation. By applying the DQN algorithm, the authors searched for the optimal frequency of the actuator at AoAs of 12° and 15°. The effectiveness of this strategy was experimentally investigated in a low-speed wind tunnel at a chord Reynolds number of 6.3×10^5. This study further confirmed the potential of DRL and paved the way for applying DRL-learned policies to other AFC-controlled airfoils. In addition to the field of flow control, Lou et al. (2023) applied DRL to aerodynamic optimization. Using a deep Q-network (double DQN), the authors realized the goal of finding the best airfoil geometry to attain a high lift-drag ratio, and this trial also confirmed the power of DRL in the field of applied fluid mechanics.

Therefore, the aim of this research is to combine the power of the DRL algorithm with CFJ to establish a closed-loop control framework for a stalled airfoil experiencing significant flow separation. The main objective of this control framework is to enhance the lift coefficient of the airfoil when it is in a stalled condition, thereby mitigating flow separation. To accomplish this goal, ten strategically placed pressure sensors are utilized to provide the control framework with crucial information about the flow field. The DRL agent interacts with the environment and learns its policy through these sensor data. A designed reward function is used as feedback to guide the agent’s learning process, which allows the agent to determine the appropriate mass flow rate for the CFJ, ultimately achieving the desired lift coefficient. This research is the first endeavor in utilizing a closed-loop control method for CFJ airfoil systems. This trial holds potential in using CFJ technology in complex, changeable flying situations, making it more feasible for practical engineering applications of CFJ technology.

2. METHODOLOGY

2.1 Flow Solver and 2-D Mesh of CFJ0012 Airfoil for CFD Calculation

In this paper, numerical simulations of the flow field are conducted using Ansys Fluent software, which is based on the finite volume method. The Reynolds-averaged Navier–Stokes (RANS) equations, along with the Spalart–Allmaras (SA) turbulence model, are employed to validate the numerical method for the CFJ0025 airfoil. The CFJ0025 airfoil is a modified version of the NACA0025 airfoil, specifically designed to incorporate a CFJ.

The geometry model and computational conditions are based on the referenced paper (Zha et al., 2007a). The freestream Mach number is set to 0.11, and the Reynolds number, calculated based on the chord length, is 3.8×10^5. A two-dimensional mesh, as displayed in Fig. 3, is employed for the CFD calculation.

Figure 4 illustrates a comparison of the present computed value, the experimental value (Zha et al., 2007a) and the computed value calculated by Zha et al. (2007b) at various AoAs. In Fig. 4 (a), the present calculated lift coefficient matches well with the experimental result.
Table 1 Detailed information of three grids at different densities

<table>
<thead>
<tr>
<th>Densities</th>
<th>Wrap-around points</th>
<th>Normal layers</th>
<th>Growth rate</th>
<th>Size of the slot mesh</th>
<th>Size of the external flow field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coarse</td>
<td>154</td>
<td>81</td>
<td>1.20</td>
<td>99×30</td>
<td>154×80</td>
</tr>
<tr>
<td>Medium</td>
<td>315</td>
<td>151</td>
<td>1.15</td>
<td>230×50</td>
<td>315×150</td>
</tr>
<tr>
<td>Fine</td>
<td>649</td>
<td>191</td>
<td>1.10</td>
<td>297×70</td>
<td>645×190</td>
</tr>
</tbody>
</table>

Fig. 5 2-D mesh for the CFJ0012 airfoil at three different densities: (a) coarse grid, (b) medium grid, and (c) fine grid

Fig. 6 Computed result for the CFJ0012 airfoil in three different grid densities: (a) lift coefficient and (b) drag coefficient

when compared to the referenced computed values. This indicates that the present calculation method accurately predicts the lift coefficient of the airfoil, providing satisfactory agreement with the experimental data. However, there is a significant discrepancy between the experimental and computed values of the drag coefficient, as shown in Fig. 4 (b). This discrepancy could be attributed to the probable experimental measurement error or inadequate turbulence simulation by the RANS model (Zha et al., 2007b). In the present research, the lift coefficient of the airfoil is specifically focused on, and hence, the calculation method is sufficient for the study.

In this study, the CFJ0012 airfoil is selected as the subject of investigation, and the chord length of the airfoil is 1 m. The injection and suction slots are located at 7% and 85% of the chord from the leading edge, and the widths of the injection and suction slots are 0.45%c and 0.9%c, respectively. To ensure grid independence and evaluate grid sensitivity, three different density grids are created, as depicted in Fig. 5. The grids are set in an O-type structure. The total number of grids is 15290, 57400 and 144100, respectively. The far-field is set to 50c for all three grids. To ensure $y^+<1$, the heights of the first layer are $1.37\times10^5$, $7\times10^6$, and $1\times10^6$. Other detailed information on the grids is listed in Table 1. The freestream Mach number is 0.2, and the Reynolds number based on the chord length is $4.68\times10^6$. The jet momentum coefficient is 0.04. The computed lift coefficient and drag coefficient for these three grid sizes are presented in Fig. 6. The results indicate that the computed values obtained from the medium and fine grids exhibit good agreement. However, it is observed that the computed values from the coarse grid differ significantly from those of the medium and fine grids. Considering both computational efficiency and accuracy, the medium grid is chosen for subsequent studies.

2.2 DRL-Based Closed-Loop Control Framework Using the PPO Algorithm Applied to the CFJ0012 Airfoil

The training process of DRL involves learning and optimizing the policy of the DRL agent by continually interacting with the environment. This training process can be conceptualized as an episode, denoted as $\tau$ in Eq. (1), which records how the agent interacts with and receives feedback from the environment.

$$\tau = (s_0, a_0, r_0, s_1, a_1, r_1, ..., s_T, a_T, r_T)$$  \hspace{1cm} (1)

where $s$ represents the state of the environment; $a$ represents the action given by the agent; and $r$ represents
the reward value calculated through a defined reward function.

At the start of an episode, the agent observes the initial state of the environment. Based on this information, the untrained agent will choose an action randomly from its action set. The chosen action is then implemented in the environment, causing it to transition to a new state. The environment provides the agent with a reward value, which serves as feedback on the quality of the action taken. The agent then updates its policy using this information, with the goal of finding a policy that maximizes the discounted return defined in Eq. (2).

\[ R(t) = \sum_{t=0}^{\infty} \gamma^t r_t \]  

where \( \gamma \) is a discount factor ranging from 0 to 1, which is used to decrease the importance of future rewards.

Various algorithms have been proposed in the development history of DRL, including REINFORCE, TRPO (Trust Region Policy Optimization) (Schulman et al., 2015), PPO, etc. Of these algorithms, the PPO algorithm, proposed by Schulman et al. (2017), has been widely recognized for its remarkable efficiency and robustness in practical applications. Therefore, a PPO agent is utilized in this paper as a part of the closed-loop control system.

The detailed framework based on the PPO algorithm is shown in Fig. 7, and the training parameters of PPO are listed in Table 2. The PPO agent used in this paper includes an actor network, whose structure is automatically defined by tensorflow according to the input types and shapes. The PPO agent will optimize its policy by interacting with the environment of a numerical flow field of the controlled object: the CFJ0012 airfoil with a very weak \( C_p \) of 0.0001 at an angle of attack of 18°. The freestream Mach number is 0.2, and the Reynolds number based on the chord is 4.68×10^6. At the beginning of the training process, the actor network takes the initial state \( s \) as input and then outputs the average value \( \mu \) and the variance \( \sigma \) to form a Gaussian distribution. From this distribution, an action value \( \pi(a|s) \) within a set range will be randomly chosen. After implementing a jet with a mass flow rate of \( \tau \), the flow field transitions to the next state \( s' \). Simultaneously, a reward value \( r \) is computed, which provides immediate feedback for the agent’s actions. To improve the agent’s performance, a dataset consisting of the state-action-reward-next state tuple \((s, a, r, s')\) is stored in an experience replay buffer. This buffer allows the agent to learn from past experiences by sampling mini batches of transitions. During training, the parameter of the actor network \( \theta \) is updated by minimizing the loss function shown in Eq. (3).

\[ L_{\text{actor}} = \mathbb{E} [\min(l(\theta)R_c, \text{clip}(l(\theta), 1-\epsilon, 1+\epsilon)R_c)] \]  

where \( l(\theta) = \pi_{\text{old}}(a|s) / \pi_{\text{new}}(a|s) \) is the probability ratio of the new policy and the old policy and \( \epsilon \) is a hyperparameter with a value of 0.25, which limits the value of \( l(\theta) \) to the range of \([1-\epsilon, 1+\epsilon] \).

The above introduction to the DRL framework shows that it is crucial to carefully design the state space, the action space and the reward function. Hence, the design of these three parts is given as follows.

### A. Determination of the State Space

The state space of this framework involves the pressure data collected from ten carefully selected points located on the surface of the airfoil. The choice of these specific points is determined by two crucial criteria (Samy et al., 2010):

- **a. Steep Pressure Gradient**

  This criterion focuses on identifying points with a significant and noticeable change in the pressure distribution along the airfoil’s surface. These points are selected based on the presence of a steep pressure gradient, indicating a rapid transition in the pressure distribution.

- **b. Substantial Variance in the Pressure Values**

  The second criterion involves the identification of points that have considerable variation in the pressure values as the flight state changes. These points reflect sensitivity to alterations in the parameters such as the height \( H \), Mach number \( Ma \), and angle of attack (AoA).

To identify regions that meet the first criterion, the \( C_p \) distribution on the CFJ0012 airfoil is calculated. Figs. 8 (a)-(d) show the \( C_p \) distribution on the CFJ0012 airfoil under various flight conditions. Each figure within the set illustrates the \( C_p \) curves with the variation of a single parameter. Upon observing Figs. 8 (a)-(d), all figures consistently reveal that regions near the leading edge and the trailing edge, as well as the locations of the injection and suction slots, exhibit noteworthy changes in the pressure distribution along the surface. These areas display steep pressure gradients, indicating rapid transitions in the pressure values along the surface of the...
Fig. 8 $C_p$ distribution of the CFJ0012 airfoil under different flight conditions with only one flight parameter change: (a) $C_p$ curves with only $H$ changes, (b) $C_p$ curves with only $AoA$ changes, (c) $C_p$ curves with only $Ma$ changes, and (d) $C_p$ curves with only $C_μ$ changes.

Fig. 9 Locations of the pressure probes where the pressure gradient is steep.

Fig. 10 Numbering of the airfoil surface grid nodes.

As a result, the range of pressure probes can be narrowed down to the red area on the airfoil, as shown in Fig. 9 (starting from the leading edge point, a total of 317 grid nodes of the airfoil surface are numbered clockwise, as shown in Fig. 10).

To further refine and determine the appropriate locations of the pressure probes, Criterion 2 should be considered. In the case of the CFJ airfoil, four key flight parameters are taken into account: the height $H$ (ranging from 0 km to 8 km with an interval of 2 km), Mach number $Ma$ (ranging from 0.2 to 0.5 with an interval of 0.1), jet momentum coefficient $C_μ$ (ranging from 0.01 to 0.2 with an interval of 0.01), and angle of attack $AoA$ (ranging from $0°$ to $20°$ with an interval of $1°$). Among these parameters, $C_μ$ is a particular flight parameter for the CFJ airfoil. It is a dimensionless parameter that quantifies the intensity of the CFJ, and the definition equation is shown in Eq. (4).

$$C_μ = \frac{mV_j}{\frac{1}{2}\rho_∞V_∞^2S} \tag{4}$$

where $m$ is the mass flow rate of the jet; $V_j$ is the velocity of the jet; $\rho_∞$ is the density of the freestream; $V_∞$ is the velocity of the freestream; and $S$ is the reference area. Therefore, 8400 (5x4x20x21) groups of flight states are obtained through the method shown in Fig. 11.

Then, the variance of the pressure per point is calculated through Eq. (5).

$$D_j^2 = \frac{\sum_{i=1}^{8400}(x_{ij} - \bar{x}_j)^2}{8400} \tag{5}$$
Fig. 11 8400 samples of flight states

Fig. 12 Variances of the pressure values per point

where $i \in \{1, 2, 3, ..., 8400\}$ is the serial number of the flight states; $j \in \{1, 2, 3, ..., 317\}$ is the serial number of the grid point on the surface of the airfoil; $x_i$ denotes the pressure value of the $j^{th}$ grid point under the $i^{th}$ flight states; and $\bar{x}_j$ is the average pressure value of the $j^{th}$ grid point under 8400 flight states.

Figure 12 displays the variances of the pressure values for each grid point. P32, P40, P116, P165 and P300 exhibit larger variances compared to the other points. These specific locations are indicated in Fig. 13. Interestingly, these points fall within the red area of the airfoil, as depicted in Fig. 9.

In the final selection of the pressure probes for the DRL framework, a set of pressure values of 10 pressure probes (Fig. 14) is chosen as members of the state space, and the coordinates of the pressure probes are listed in Table 3. Except for P57 and P104, all other points in the set satisfy at least one of the previously mentioned criteria. P57 and P104 are intentionally chosen because they are located within the jet slot. These points have the potential to provide information regarding the intensity of the jet.

B. Determination of the Action Space

In this framework, the range of the action space is exactly the range of the jet momentum coefficient. The experimental results, shown in Fig. 15, show that at an AoA of 18 degrees, the lift coefficient of the airfoil varies between 1.7 ($C_\mu = 0.04$) and 2.3 ($C_\mu = 0.1$). Therefore, in the present study, 2.0 is set as the control target, and the goal of the DRL training is to find a corresponding $C_\mu$ to help the CFJ airfoil reach the lift coefficient of 2.0. To simplify the numerical simulation process, the boundary conditions of the injection slot and the suction slot are modified to the mass flow inlet and mass flow outlet, respectively. Based on Eq. (4), the range of the mass flow rate is supposed to be $[0.8$
kg/s, 1.2 kg/s]. However, to further display the decision-making ability of the DRL algorithm, the range is slightly broadened to [0.6 kg/s, 1.2 kg/s].

C. Determination of the reward function

The reward value should increase as the lift coefficient of the airfoil approaches the target value of 2.0. Therefore, the reward function of this case is designed and shown in Eq. (6).

\[ r_t = -|C_{L} - 2| \]  \hspace{1cm} (6)

3. RESULTS AND DISCUSSIONS

The whole training process starts from an initial flow field where \( C_{\mu} \) is very small \( (C_{\mu} = 0.0001) \), i.e., the initial state. To obtain the instant lift coefficient according to the change in \( C_{\mu} \), a transient solver is used in this section. One important indicator for assessing the performance of the DRL training process is the variations in the reward values over the episodes. Here, one episode includes 40 actions. The mass flow rate of the jet changes gradually to the action value given by the agent in 25 numerical time steps according to Eq. (7) to ensure that the CFD solver will not crash.

\[ m_t = m_{t-1} + \beta(a - m_{t-1}) \]  \hspace{1cm} (7)

where \( m_t \) is the current mass flow rate of the CFJ; \( m_{t-1} \) is the mass flow rate at the last numerical step; \( a \) is the action value given by the agent; and \( \beta = 1/3 \) is a parameter for adjusting the variation smoothness of the mass flow rate.

The reward curve over the episodes is depicted in Fig. 16. There are five points where the reward is extremely low, which can be attributed to the “reset” process. The “reset” process is that the beginning state of an episode will be reset to the initial state mentioned before, and its possibility is less than 20%. Otherwise, a new episode will start from the final state of the last episode. There is a noticeable increase in the reward value after the 40th episode, indicating that the agent begins to discover a favorable policy. However, the growth rate becomes stable after the 90th episode, suggesting that the agent’s policy variation becomes less pronounced.

Four tests are carried out after Episodes 60, 80, 100 and 120, as shown in Fig. 17, and each test lasts for one second. In the “without control” part of Fig. 17, the CFJ0012 airfoil with a very weak \( C_{\mu} \) of only 0.0001 suffers a very low lift coefficient of approximately 0.74. However, when the CFJ0012 airfoil is controlled according to the agent’s policy, its lift coefficient experiences a significant increase and eventually approaches the target value of 2.0. Among the four tests, the policy acquired after Episode 120 exhibits the best performance. The lift coefficient becomes stable at approximately 2.01, with a negligible error of only 0.5%. In addition, the policy has physical meaning: when the lift coefficient is much less than the target value 2, the agent tends to output large action values to help the airfoil rapidly increase its lift coefficient, whereas when the lift coefficient approaches 2.0, the jet momentum coefficient gradually decreases and settles at approximately 0.049.

Figure 18 provides a visual representation of the pressure contours at an AoA of 18° for two scenarios: flow fields with a weak jet (\( C_{\mu} = 0.0001 \)) and a much stronger jet instructed by the agent. In Fig. 18 (a), it is clearly observed
that flow separation takes place on the upper surface of the airfoil. This separation creates an adverse pressure gradient and reduces the overall lift generated by the airfoil. However, in Fig. 18 (b), when the CFJ is activated according to the agent’s policy, the flow separation is effectively suppressed. This is evidenced by the pressure contours and streamlines showing a smooth and attached flow along the upper surface of the airfoil. The successful suppression of flow separation and transition to attached flow, as depicted in Fig. 18 (b), leads to a significant enhancement in lift, as shown in the “with control” portion in Fig. 17.

To test the policy’s generalization ability, two more tests are conducted: one test at $\text{AoA}=16^\circ$ and the other test at $\text{AoA}=20^\circ$. The test results are shown in Figs. 19 (a) and (b), respectively. At $\text{AoA}=16^\circ$, the lift coefficient with a very weak jet ($C_\mu=0.0001$) is approximately 0.75. After implementing the jet based on the policy, the lift coefficient increases significantly and reaches 1.89 within one second. However, this value deviates from the target value of 2.0, resulting in an error of 5.5%. At $\text{AoA}=20^\circ$, when $C_\mu$ is very weak, the lift coefficient ($C_l$) fluctuates between 0.94 and 1.2. When the agent instructs a value of $C_\mu=0.055$, the $C_l$ curve rises to 2.1 within one second, with an error of 5% compared to the target value of 2. Therefore, the policy trained at $18^\circ$ can instruct the airfoils at $16^\circ$ and $20^\circ$ to realize the control target to some extent, which means that the policy has a generalization ability. Moreover, the agent will provide different action values according to different states, for instance, $C_\mu=0.049$ for $\text{AoA}=16^\circ$ and $C_\mu=0.055$ for $\text{AoA}=20^\circ$.

The pressure contours corresponding to the tests are shown in Fig. 20 and Fig. 21. Fig. 20 (a) displays the pressure contour of the flow field at $\text{AoA}=16^\circ$ with a very weak jet ($C_\mu=0.0001$). In this case, obvious flow separation occurs on the upper surface of the airfoil.
Fig. 21 Comparison of the pressure contours with and without control at AoA=20°: (a) the flow field without control, and (b) the flow field with control

However, when a stronger jet is applied according to the agent’s policy, the flow separation completely disappears, as shown in Fig. 20 (b). At AoA=20°, vortex shedding occurs on the upper surface of the uncontrolled airfoil (Fig. 21 (a)). This vortex shedding leads to fluctuations in the lift coefficient, as observed in the “without control” portion in Fig. 19 (b). However, when the jet with $C_\mu=0.055$ is implemented, this phenomenon is effectively suppressed (Fig. 21 (b)), resulting in no fluctuation in the lift coefficient.

4. CONCLUSION

In this paper, a closed-loop control framework based on the PPO algorithm is developed for the CFJ airfoil. The framework exhibits accurate control performance, successfully achieving the settled control target of enhancing the lift coefficient to a specific value of 2.0, with an error of only 0.5%, for the stalled CFJ airfoil at an AoA of 18°. This achievement is made possible by the policy derived from the PPO agent, which interacts continuously with the environment through the state-action-reward-next state tuple. Additionally, generalization ability tests are conducted to evaluate the performance of the policy at different AoAs of 16° and 20°. The results indicate that the policy can be applied to other AoAs as well. By effectively suppressing flow separation and vortex shedding on the upper surface of the airfoil, the lift coefficient is significantly improved.

Future studies are planned to focus on further enhancing the robustness of this closed-loop control framework and increasing the control efficiency in terms of energy conservation. The study of the robustness of the framework includes investigating whether the framework can facilitate the attainment of different control targets and examining the applicability of the policy in various flight situations, such as different Mach numbers and Reynolds numbers. In terms of energy conservation, penalty terms such as the power consumption coefficient and drag coefficient could be added to the reward function to conduct further research.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (Grant No. 11972306) and the 111 Project of China (B17037).

CONFLICT OF INTEREST

The authors have no conflicts to disclose.

AUTHORS CONTRIBUTION

Yiran Zhao: Conceptualization, Methodology, Investigation, Writing. Heyong Xu: Conceptualization, Supervision, Writing, Review. Zhengyao Xie: Methodology, Formal Analysis, Software.

REFERENCES


