Active Flow Control Using Deep Reinforcement Learning with Time-Delays in Markov Decision Process and Autoregressive Policy

DOI:
10.1063/5.0086871

Active Flow Control Using Deep Reinforcement Learning with Time-Delays in Markov Decision Process and Autoregressive Policy

Yiqian Mao (毛逸谦)[1,a)], Shan Zhong[1], Hujun Yin[2]

[1]Department of Mechanical, Aerospace and Civil Engineering,

[2]Department of Electrical and Electronic Engineering,

The University of Manchester, Oxford Road, Manchester, M13 9PL, UK

[a)] Author to whom correspondence should be addressed: yiqian.mao@postgrad.manchester.ac.uk

Abstract

Classical active flow control (AFC) methods based on solving the Navier-Stocks equations are laborious and computationally intensive even with the use of reduced-order models. Data-driven methods offer a promising alternative for AFC and they have been applied successfully to reduce the drag of two-dimensional bluff bodies, such as a circular cylinder, using deep reinforcement learning (DRL) paradigms. However, due to onset of weak turbulence in the wake the standard DRL method tends to result in large fluctuations in the unsteady forces acting on the cylinder as the Reynolds number increases. In this study, a Markov decision process (MDP) with time delays is introduced to model and quantify the action delays on the environment in a DRL process due to the time difference between control actuation and flow response along with the use of a first-order autoregressive policy (ARP). This hybrid DRL method is applied to control the vortex shedding process from a two-dimensional circular cylinder using four synthetic jet actuators at a freestream Reynolds number of 400. This method has yielded a stable and coherent control which results in a steadier and more elongated vortex formation zone behind the cylinder hence a much weaker vortex shedding process and less fluctuating lift and drag forces. Compared to the standard DRL method, this method utilizes the historical samples without additional sampling in training and it is capable of reducing the magnitude of drag and lift fluctuations by approximately 90% while achieving a similar level of drag reduction in the deterministic control at the same actuation frequency. This study demonstrates the necessity of including a physics-informed delay and regressive nature in the MDP and the benefits of introducing ARPs to achieve a robust and temporal-coherent control of unsteady forces in active flow control.

***Key words*:** Flow control, reinforcement learning, artificial neural networks, autoregressive model, circular cylinder, drag reduction.

1. Introduction

Flow control has attracted a great deal of research attentions since 1990's because of its potential in improving the aerodynamic performance of transport vehicles beyond a level that could be achieved through optimising the vehicles' geometry alone[1–3]. Passive flow control (PFC) and open-loop active flow control (AFC) methods have been extensively investigated due to their simplicity in implementation and low maintenance need[4–6]. However, these methods cannot always yield desired control effects, especially at off-design conditions[7]. Therefore, more research attentions have begun to be directed towards developing closed-loop AFC methods in recent years in pursuit of higher flow control effectiveness across a larger operation envelope[8–10].

Due to the high dimensional and nonlinear nature of the governing equations of complex fluid flows, reduced-order models are often required to implement optimization or control with classical AFC methods. Although some physics-based and mathematics-based models[11,12] have shown good feasibility and reliability in practice, achieving a good balance between high fidelity and efficiency remains to be a laborious task. By contrast, machine learning (ML) methods are data-driven and allow engineers to perform AFC without complete prior knowledge of fluid physics. In complex AFC problems with multi-input and multi-output, ML can be applied to develop a computationally efficient surrogate model for predicting control parameters to minimize cost function[13].

A number of studies have been undertaken to develop effective and reliable ML methods for AFC in recent years, e.g. Shimomura et al.[14], Li et al.[15] and Fukami et al[16]. Among these ML methods, supervised learning (SL) and reinforcement learning (RL) have been widely adopted[17–19]. SL aims to establish an optimal model based on existing knowledge and hence requires a sufficient amount of representative labeled data. Recent SL applications in AFC have made use of either artificial neural networks (ANNs), genetic algorithms[20,21] (GAs) or Gaussian process[22,23] (GP) to develop algorithms for modelling complex patterns, flow prediction and control problems. The fully-connected neural networks (FCNNs) and convolutional neural networks (CNNs) are widely-used in ANNs-based AFC methods[24–26]. A comparison of FCNNs and CNNs have been made by Han and Huang[27,28] in opposition control of turbulent channel flows. Some hybrid models combining CNNs with multi-layer perceptron[16] or autoencoder (AE)[29–31] have been proposed for spatial reconstruction, dimensionality reduction and flow estimation in the flow field. To describe the spatial-temporal flow evolution, a combined model of long short-term memory (LSTM) networks and CNN-AE has been proposed[32] and tested[33] for its capability and robustness in controlling the flow around a circular cylinder at a Reynolds number (*Re*) from 20 to160.

In most AFC tasks with SL, controls are hard to be labeled as 'correct' or 'wrong' even by an expert. Besides, the amount of representative labelled data (such as controlled variables and actuator signals) required by SL may

be unrealistic to obtain. In contrast, RL does not need the 'correct' strategy as supervisory information but generates its own data by exploring and evaluating actions against a reward function[34]. Benefiting from its capacity of modelling policies and value functions in complex RL tasks with continuous state and action space, deep reinforcement learning (DRL) which combines RL and deep learning has been applied to automatically perform AFC strategies[35–38]. In DRL, an RL agent samples action-state pairs through interacting with an environment and adopts ANNs as function approximators to estimate a value function or a policy from the sampled histories. Two popular DRL methods, i.e. proximal policy optimization (PPO)[39] and twin delayed deep deterministic policy gradients (TD3)[40], have been widely adopted in AFC to process continuous action space with high dimensions[41–45].

The application of DRL on the AFC of a bluff body is a sequential decision-making process, and the control difficulty increases with the complexity of the flow state. In view of the PPO method's satisfactory training efficiency and little demand for hyperparameter tuning, Rabault et al.[46] applied a DRL paradigm combining a PPO method with a fully connected ANN to control the Karman vortex street behind a two-dimensional circular cylinder using two synthetic jets, one located on the top surface and the other on the bottom surface of cylinder, at a Reynolds number of 100. Directed by a reward function which minimizes the drag while keeping the lift low, the DRL agent found a control strategy which suppresses the vortex shedding process and achieves a drag reduction of approximately 8%. The control strategy discovered using the PPO method was capable of counteracting against the vortex shedding with the synthetic jet actuation hence stabilizing the fluctuations of aerodynamic forces while reducing drag

Adopting learning parallelization through a multi-environment approach proposed by Rabault et al.[47], Tang et al.[48] implemented the DRL paradigm to control the vortex shedding process behind a two-dimensional cylinder using four synthetic jets over a range of Reynolds numbers from 100 to 400. A drag reduction of 20.4% and 33.1% were achieved at Re = 200 and 400, respectively. However, random fluctuations in drag and lift appeared to develop at Re = 200 and increase in magnitude at Re = 400. Tang et al.[48] attributed the unstable control to the inherent instability of the flow at Re=400. This is because when Re exceeds about 260, turbulence starts appearing in the shear layer and begins to affect in the state space observed in the near wake[49]. They then introduced global training and smooth action updates to improve the robustness of a DRL paradigm. Although the control becomes more stable at Reynolds numbers of 200, 300 and 400, a considerable level of irregular fluctuations in drag and actuation is still present[48].

Ren et al.[50] also encountered a similar problem in applying DRL control at a higher Reynolds number. Using the lattice-Boltzmann method (LBM) to simulate the flow environment, they applied DRL to the same flow setup as in Rabault et al.[46] at a Reynolds number of 1000. The DRL agent was able to find effective AFC strategies with more training required than at Re = 100. Nevertheless, the temporal variations in drag exhibit much more random and significant fluctuations at Re = 1000 and this was also attributed by the authors to the presence of the weakly turbulence in the flow at this higher Reynolds number.

Overall, the results from the aforementioned studies have revealed that although a considerable amount of reduction in the time-averaged drag can be achieved through AFC with DRL, as the Reynolds number increases the level of temporal fluctuations in drag tends to become increasingly more random and severe. At Reynolds numbers higher than 200, the standard PPO method tends to find a 'cheating' policy under which although the time-averaged reward is maximized, random fluctuations and sudden large extremes in drag and lift caused by irregular control appear at some instants. This inferior control policy can be difficult to avoid by modifying the reward function since an instant return of the reward at each numerical time step is too expensive and not practical. Due to the appearance of turbulence in the state space, insufficient regression of the ANN of the time series in the decision process may result in a deteriorating robustness and temporal-coherence of control trained through the PPO method.

In this study, a Markov decision process (MDP) with time delays is introduced to quantify the action delays in the DRL process due to the time elapse between actuation and response of flow along with the use of a first-order autoregressive policy (ARP). This hybrid DRL method is applied to control the vortex shedding process from a two-dimensional circular cylinder using four synthetic jet actuators at a freestream Reynolds number of 400. This method has yielded a stable and coherent control which results in a steadier and more elongated recirculation zone behind the cylinder hence a much weaker vortex shedding process. Compared to the standard PPO method, this method utilizes the historical samples without additional sampling in training and it is capable of reducing the magnitude of drag and lift fluctuations by approximately 90% while achieving a similar level of drag reduction in the deterministic control at the same actuation frequency. This study demonstrates the necessity of including a physics-informed time delay in the MDP and the benefits of introducing ARP to achieve a robust and temporal-coherent control of unsteady forces in active flow control. Reduction of high-level temporal fluctuations in drag or lift will help to decrease dynamic loads and structural fatigue leading to an improved structural durability and operational safety[51–53].

2. Methodology and implementation

2.1. Numerical setup

The flow configuration in this study is a two-dimensional laminar flow passing around a circular cylinder at a Reynolds number (*Re*) of 400, where $Re = \bar{U}D/\nu$, $\bar{U}$ is the average velocity of inflow, ν is the kinematic viscosity of the fluid, and *D* is the cylinder diameter. Tang et al.[48] applied DRL control within the range of *Re* = 100 – 400 and they obtained the largest amount of drag reduction as well as the highest level of drag force fluctuations at *Re* = 400. Therefore, *Re* = 400 is selected in this study to enable a comparison of flow control results using different DRL strategies. In this study, the length, velocity, time and vorticity are nondimensionalized with D, $\bar{U}$, $t = D/\bar{U}$ and $1/t$, respectively.

To enable a direct comparison, the same numerical setup used by Tang et al.[48], which has already been validated for simulating this flow, is also applied in our CFD simulations. The size of the rectangular computational domain is chosen to be $22D(length) \times 4.1D(height)$. As shown in Fig. 1, the cylinder centre is placed with an offset of $0.05D$ from the centreline of the computational domain to initiate vortex shedding in the simulations at this Reynolds number. A no-slip condition is applied on the top and bottom boundary as well as on the surface of the cylinder. The inflow boundary is located at $2D$ upstream of the cylinder centre, and a velocity inlet condition with a parabolic velocity profile is applied at this boundary so as to match with the wall boundary conditions imposed on the top and bottom of the computational domain. An outflow boundary condition is set at the outlet boundary.

Four control jets (J_1, J_2, J_3, J_4) are symmetrically installed on the cylinder surface with an angular position of $(\varphi_1, \varphi_2, \varphi_3, \varphi_4) = (75°, 105°, 255°, 285°)$ respectively. The velocity profile at each jet exit follows the cosine distribution defined in Eq. (1) to ensure the velocity continuity between the velocity inlet of jets and no-slip surfaces of cylinder,

$$U_i(\varphi, Q_i) = \frac{\pi}{\alpha D} Q_i \cos\left(\frac{\pi}{\alpha}(\varphi - \varphi_i)\right) \tag{1}$$

where φ is the angle measured from the positive semi-axis of X-axis, α is the angular width of each jet which is set as $\pi/18$, Q_i is the maximum mass flow rate of J_i taken at its centre location, φ_i. Let $Q_i^* = Q_i/Q_{ref}$, where Q_{ref} is the reference mass flow rate, $Q_{ref} = \int_{-D/2}^{D/2} \rho U(y) dy$, in which $U(y)$ is the inlet velocity profile upstream of the cylinder. A constraint of $|Q_i^*| \leq 0.05$ is applied to each jet to avoid large power consumption and non-physical actuations. A constraint of zero total mass flow rate of the four jets is applied to avoid adding and

subtracting mass directly from the environmental flow. This condition ensures that control strategies learned by the agent are more realistic and the numerical scheme is more stable.

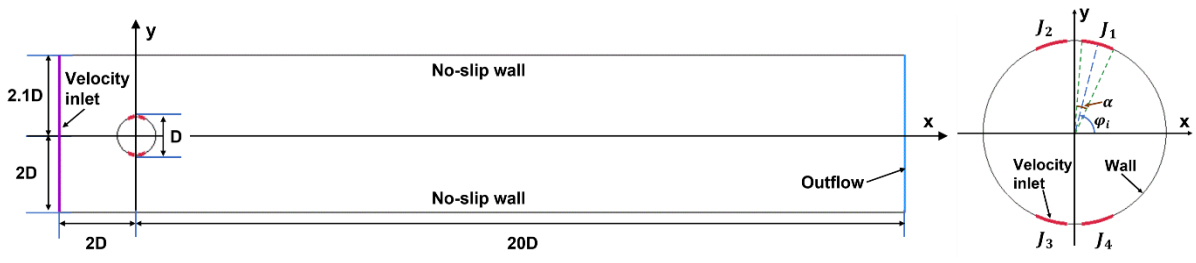

Fig. 1. Schematics showing computational domain, boundary conditions and control jet locations.

In the present work, the flow is assumed to be viscous and incompressible. The incremental pressure correction scheme[54] based on the finite element method is adopted to perform the numerical simulations using the open-source computing platform FEniCS[55]. The same computational mesh and solution methods used by Tang et al.[48] are applied in our study to eliminate discrepancies in results due to the use of different numerical methods. An unstructured mesh composed of 25865 triangular grids is established for the domain discretization where the grids near the cylinder surface are refined, see Fig. 2. To avoid duplications, the detailed computational setting will not be repeated here. For more details of the numerical setup, please refer to Tang et al.[48]

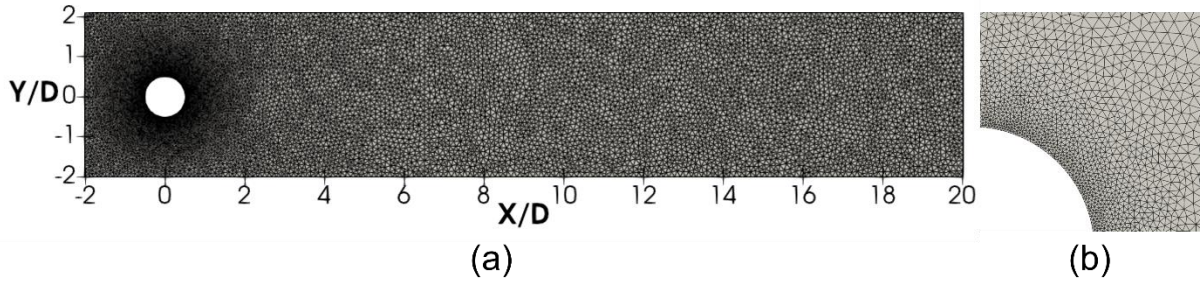

Fig. 2. Computation mesh used in the simulations. (a) The whole domain; (b) near the cylinder surface.

The drag F_D and lift F_L acting on the circular cylinder are obtained via surface integrations of the viscous stress and pressure acting around the cylinder. The drag coefficient, C_D, and the lift coefficient, C_L, are calculated using Eqs. (2) and (3) from F_D and F_L. The standard deviation of C_D and C_L are defined as σ_D and σ_L,

$$C_D = \frac{2F_D}{\rho \bar{U}^2 D} \tag{2}$$

$$C_L = \frac{2F_L}{\rho \bar{U}^2 D} \tag{3}$$

The Strouhal number, $Sth = fD/\overline{U}$, is used to represent the non-dimensional frequency in this study. The Strouhal number corresponding to the dominant frequency of C_D and C_L are denoted as Sth_D and Sth_L. The Strouhal number corresponding to the period of vortex shedding T_S is denoted as Sth_S.

The same code for flow simulations used by Tang et al[48] was adopted in this study, for which case validation has been presented by Tang et al[48]. Nevertheless, we have conducted our own simulations using different mesh configurations and time-step discretization to validate the spatio-temporal convergence of our simulations and provided the results in Table 1. The numerical solver adopts the direct solution of lower-upper (LU) decomposition and hence the iterative residuals do not exist in the computations. Configuration 2 has the identical mesh configuration and time step used by Tang et al[48]. As it can be seen in the table, the results produced with the medium mesh (Configuration 2) are within 0.1% of those with the fine mesh (configuration 3) except for a slightly larger discrepancy in Sth of less than 0.5%. At to the convergence of time step, when δt is reduced from 0.001 to 0.0005 (the smallest time step) the discrepancies of all results are within 0.1%. Based on our independent study, the mesh density and time step used in Configuration 2, which are identical to those used by Tang et al, have produced converged results. Therefore, they are used in the simulations in our work. This will ensure that any possible discrepancies due to differences in the numerical setup can be eliminated when different DRL methods are compared.

Table 1 Results of spatiotemporal convergence study. δt is the numerical time step.

Reynold number	Configuration	Mesh resolution		δt	$\overline{C_D}$	$C_{D,max}$	$\overline{\lvert C_L \rvert}$	$C_{L,max}$	Sth_S
400	1	Coarse	7166	0.001	3.166	3.501	1.852	2.962	0.3416
	2	Medium	25865	0.001	3.170	3.467	1.810	2.919	0.3417
	3	Fine	128442	0.001	3.171	3.468	1.809	2.919	0.3400
	4	Medium	25865	0.0005	3.168	3.467	1.808	2.918	0.3417
	5	Medium	25865	0.0015	3.172	3.471	1.813	2.925	0.3451

2.2. DRL procedure

The DRL framework used in this study is presented in Fig. 3. The DRL methods used in this study are based on the PPO algorithms implemented through TensorForce API[56]. The DRL framework considers the AFC problem as a goal-seeking agent interacting with an uncertain flow environment. The agent is a computer program representing a learner and decision-maker. The environment is the flow system simulated by the CFD solver, responding to the agent's actions and presenting new states. In each training case, the agent starts observing the initial state and act when the vortex shedding is fully developed in the uncontrolled wake flow. In each episode,

the PPO agent samples a sequence τ defined in Eq. (4) composed of N_a combinations of state (s_t), action (a_t) and reward (r_t) through interactions with the environment,

$$\tau = (s_1, a_1, r_1), (s_2, a_2, r_2), \cdots (s_{N_a}, a_{N_a}, r_{N_a}), \tag{4}$$

where N_a is the number of times the DRL agent applying the policy π each episode. A small N_a corresponds to a large update interval of action, leading to inefficient sampling and training, while a large N_a may produce the jet actuation too frequently and affect the numerical stability. In the present work, N_a is set following the experience of Tang et al[48]. The agent pauses learning by τ at the end of each episode. Once learning is complete, the agent resumes the paused flow environment and proceed to a new episode. The state space consists of the velocity vector of 236 probes with the same arrangement as in the study of Tang et al.[48]. The action space consists of four action factors corresponding to the mass flow rates of four jets with zero total mass flow rate. The reward is defined by the following equation aiming to minimize the drag while keeping the lift low,

$$r = -\langle C_D \rangle_a - w \cdot |\langle C_L \rangle_a| + C \tag{5}$$

where $\langle \cdot \rangle_a$ is the time average over an action time step T_a. w is a weighting coefficient used for adjusting the pursue of agent in large drag reduction and keeping lift low. In the present work, w is set as 0.2, the same as in the study of Tang et al[48]. C is a constant of 4 for plotting more intuitive learning curves in the results and does not affect training. Instead of focusing on the current reward, the agent aims to maximize the average cumulative reward $R(t) = \sum_{i>t} \gamma^{i-t} r_i$, where $\gamma \in (0,1)$ is a discount factor used for adjusting the interest of the agent to focus on long-term or short-term goals and should be set close to but not exceed 1. Following the same setup of Tang et al, γ is set as 0.97 in the present study.

The PPO algorithms are policy gradient method. The policy function, π_θ is represented by an ANN where all weights are collectively given by the variable, θ. Fig. 4 presents the network architectures of the standard and hybrid PPO methods. The PPO algorithms in this study have two networks: an actor network and a critic network. Both actor and critic networks comprise an input layer, an output layer and two fully connected hidden layers. The size of each hidden layer is set to 512, following an empirical test by Rabault et al.[46] with both modeling ability and training efficiency.

In the actor network, the input is the state set, and the output is the action distribution. The critic network is used to approximate the state-value function $V(s)$. The state-action value function $Q(s,a)$ is replaced by an advantage function $A(s,a) = Q(s,a) - V(s)$ to reduce its variability and accelerate training. When the decision-making horizon is infinite, the advantage function can be represented as Eq. (6).

$$\hat{A}_t = R(t) - V(s_t) \tag{6}$$

where $\hat{A}_t$ is an estimator of A at time step t. The PPO algorithms, proposed by Schulman et al.[39] as a optimization of the TRPO algorithms, aim to maximize a "surrogate" objective defined in Eq. (7).

$$L^{CPI}(\theta) = \hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}\hat{A}_t\right] = \hat{\mathbb{E}}_t\left[r_t(\theta)\hat{A}_t\right] \tag{7}$$

where $\hat{\mathbb{E}}_t[\cdot]$ indicates the empirical expectation over time, $r_t(\theta)$ denotes the probability ratio of current policy π_θ to previous policy $\pi_{\theta_{\text{old}}}$. To avoid an excessively large policy update, a clipped surrogate objective defined in Eq. (8) is applied to the actor network.

$$L^{CLIP}(\theta) = \hat{\mathbb{E}}_t\left[min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right] \tag{8}$$

where the hyperparameter ϵ is chosen as 0.2 as suggested by Schulman et al.[39]. The objective of the critic network is defined as $\hat{\mathbb{E}}_t\left[-\hat{A}_t^2\right]$.

In the present study, the duration of each episode is set as $T_e = 20$, and the number of action updates per episode N_a is set to 200. $T_a = T_e/N_a$ equals to 100 numerical time steps, corresponding to 3.3%T_S, which is the shedding period in the baseline case. Between two action updates, the value of jet control at each numerical time step is given by a linear interpolation defined in Eq. (9) to avoid jumps of computing pressure and velocity in the numerical procedure.

$$j_{l,k} = a_{l-1} + k\frac{a_l - a_{l-1}}{N_n} \tag{9}$$

where $N_n = 100$ is the total number of numerical time steps for an action time step. $k = 1,2\cdots N_n$ is the current numerical time step of the current action. $l = 1,2\cdots N_a$ is the current action time step. $j_{l,k}$ is the control value of the jet at *k*-th numerical time step of *l*-th action time step. a_l is the action space updated by DRL agent for the *l*-th action time step. A multi-environment approach proposed by Rabault and Kuhnle[47] is used for parallelizing the training into 32 processes.

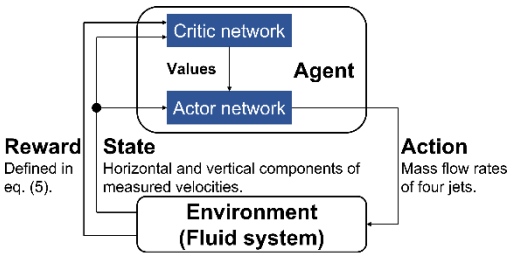

Fig. 3. Schematic of the DRL framework in the present work

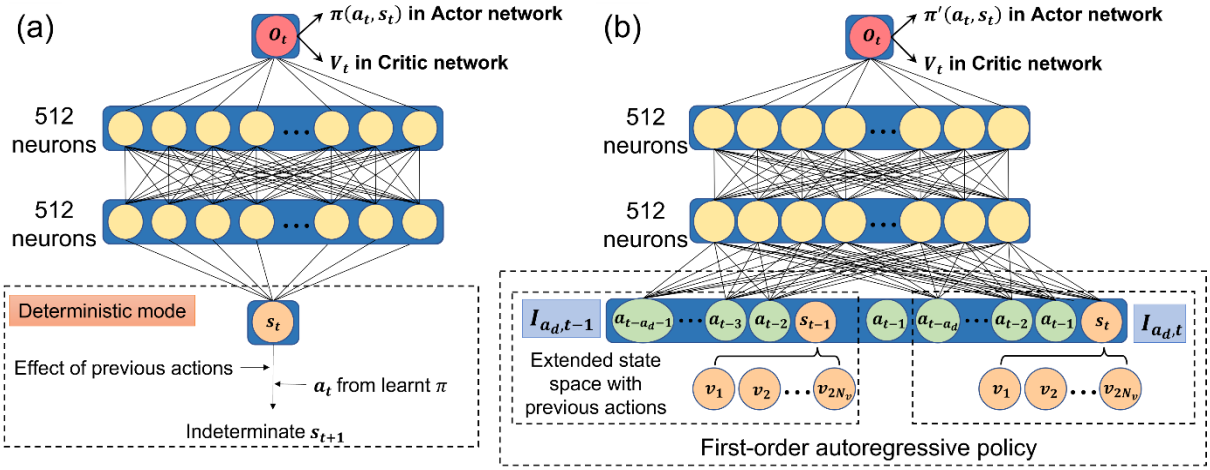

Fig. 4. Network architectures. (a) Standard PPO method. Since the effect of previous actions is not considered in the modelling of policy π, the environmental state transition may become random due to untreated action delays in the deterministic mode. (b) Proposed hybrid PPO method. A first-order autoregressive policy is applied to the extended state space with previous actions. N_v is the number of velocity probes.

2.3. MDP with delays

In real-world applications of reinforcement learning, three types of delays may exist in the MDPs[57]. They are: (1) observation delay, which exists when the state is not observed immediately; (2) action delay, which exists when an action does not affect the environment immediately but with a time delay; and (3) cost delay, which exists when the action-induced reward is not fully collected till after a certain time has elapsed.

An MDP with constant time delays in observation, action and cost can be denoted as a seven-tuple, i.e. $M_d = < S, A, P_A, r, o_d, a_d, c_d >$, where S and A are the state and action sets, respectively, P_A is the transition probability, and o_d, a_d and c_d represent the time delays in observation, action and cost, respectively. In this study, the states and the rewards are collected instantly leading to no observation and cost delays. However, an action delay is inherently present because a jet actuation taking place on the cylinder surface cannot affect the wake flow immediately since a finite period of time is required for its effect to propagate downstream. Therefore, the aforementioned MDP with constant action delays can be reduced to a five-tuple, i.e. $M_{ad1} =< S, A, P_A, r, a_d >$. Simple ANN is not able to model a_d in the decision process on its own. As shown in Fig. 4 (a), in the standard PPO method, since the effect of the previous actions is not considered in the modelling of policy, the environmental state transition becomes random for the same current state and action. For example, in the present AFC system, since the flow environment is influenced by the current and previous actions, if an agent under a deterministic policy acts only based on an isolate observation of the current state s_t at a discrete time step t, the environment may transit to different s_{t+1}. Such learnt policy is bound to result in fluctuations in lift and drag,

especially when turbulence occurs in the near wake. Therefore, a sufficient ANN's modelling of the control system should account for these action delays.

In the regression processes shown in Fig. 4 (a), the action distribution and the advantage at each time step are determined by the current states in an MDP without delays. However, when an action delay exists the information of previous jet actuation should be taken into account in the agent's current decision. Therefore, at each discrete time step the previous control actions should be added into the input layer of each network to correct the state-action transition and advantage estimation. According to Katsikopoulos and Engelbrecht[57], M_{ad1} can be converted into an MDP without time delays, i.e. $M_{ad2} =< I_{a_d}, A, P_A, r >$ through state space augmentation, where $I_{a_d} = \left(a_{t-a_d}, \cdots, a_{t-1}, s_t\right)$ and a_{t-i} with $i = 1, 2, 3, \ldots a_d$ are a series of action sets, i.e. the jet actuations which took place before the present time at which the state, s_t, is collected.

Given enough time a jet actuation will eventually reach the outlet of the computational domain. However, to ensure the efficiency of the training process the number of previous jet actuations to be included in the MDP has to be limited. In this study, these previous jet actuations are taken to be those which would have affected the near field flow within the vortex formation zone. This is a reasonable assumption since the lift and drag, which are used for defining the reward function, are mainly affected by the wake flow in the vortex formation zone. In the present work, the vortex formation length L_f is defined as the streamwise distance from the centre of cylinder to the point of maximum $\overline{u_x' u_x'}$, where u_x' is the fluctuation in the streamwise component of velocity with respect to the time averaged flow[58,59]. For the low Reynolds number flow studied in this paper, L_f of uncontrolled flow is calculated as 0.98. As reported in previous studies[48,50], the AFC strategies discovered that using DRL may increase the vortex formation length. Thus, a longer distance $d_a = 6L_f$ is selected to avoid missing delay information during the learning process. To estimate the amount of action delay in time, the convection velocity of jet actuation is also required and can be reasonably taken as the convection velocity of shedding vortices from the cylinder. The convection velocity of shedding vortices is found to decrease rapidly in the flow direction within a distance of 6*D* and it then becomes nearly constant[60]. To aid the determination of the averaged convection velocity of the shedding vortices in the vortex formation region, two velocity probes are placed at $(1, 0)$ and $(5, 0)$. Once the lift of the cylinder becomes periodically stable in the uncontrolled case, the velocity histories containing 20 vortex shedding cycles are sampled by the two probes. The averaged vortex convection velocity in the vortex formation region, $\bar{u}_s$, can be deduced given the axial spacing between the two probes and the time that it has taken the vortices to travel from the upstream probe to the downstream one found by a cross-correlation analysis of the velocity histories. With both the vortex formation zone length and the vortex convection velocity determined, the

time of action delay can now be calculated, i.e.$t_d = (d_a - D/2)/\bar{u}_s$. Finally, the normalised action delay a_d can be determined as $a_d = t_d/T_a$. In the present work, $a_d = 63$, and therefore the extended state space can be written as $I_{a_d} = (a_{t-63},\ \cdots, a_{t-1},\ s_t\ )$.

2.4. ARPs for continuous control DRL

In continuous control DRL, stochastic policy gradient algorithms generally depend on exploration with continuous probability distribution, such as Gaussian distribution, to discover new control strategies. However, Gaussian policies do not usually generate samples with temporal coherence thus cannot provide explorations with smooth trajectories corresponding to secure and rewarding behaviours in most practical continuous control tasks. In addition, due to the lack of historical information and experience memory in the state, the action update of the policy may not lead to effective exploration of the environment, and it becomes more and more inefficient as the action rate increases. Korenkevych et al.[61] introduced a series of stationary autoregressive processes (ARPs) to promote agent's exploration in continuous RL control tasks. These processes exhibit two main characteristics: subsequent process observations are temporally coherent with a continuously adjustable degree of coherence, and the process stationary distribution is normal.

When the complexity of the AFC problem increases, the sampling inefficiency of DRL and complicated state and action spaces may result in a more difficult exploration of the agent. A regression or time series model is expected to improve the utilization efficiency of samples and improve the temporal prediction of ANN. Therefore, the present work deploys a modified first-order ARP into the DRL paradigm. In the Gaussian policy, the actions are sampled through $\pi_\theta(a_t, s_t) \sim \mathcal{N}\left(\mu_\theta(s_t), \sigma_\theta^2(s_t)\right)$. θ is the parameter factor of π_θ. μ_θ and σ_θ^2 are both vectors parametrized by ANNs. When the episode is not updated, the policy π_θ does not change. Sampling by the same policy, the action mean will gradually approximate the expectation. We replaced the original observations s_t with extended observations $\widetilde{s_t} = \left(s_{t-p}, a_{t-p}, \cdots, s_{t-1}, a_{t-1}, s_t\right)$ of an p order ARP, while the substitute in (Korenkevych et al. 2019) is chosen as the white noise component. This modification allows ARP to improve the whole action update, rather than the noise term. As illustrated in Fig. 11 (b) later, the modified ARP is aimed to find the optimal control with overall temporal coherence. To implement the ARP, a history-dependent policy $\pi_\theta\left(a_t \middle| s_t,\ h_t^p\right)$ is defined, where $h_t^p = \left(s_{t-p}, a_{t-p},\ \cdots, s_{t-1}, a_{t-1}\right)$ includes past p states and actions. The new policy form replaces the original form of policy $\pi_\theta(a_t, s_t)$ during the descent process of policy gradient, which is a Markov stochastic process. According to the Markov property, the probability distribution of future states depends only on the current state but not on the entire historical path. Therefore, history-dependent policies with state space h_t^p do not necessarily induce Markov stochastic processes even if the environmental transition

probabilities are Markovian[62]. However, with an extended state space $\left(h_t^p, s_t\right)$ where h_t^p is fixed in size, such history-dependent policy induces a Markov stochastic process. Denote $\widetilde{M}^p = \left(\tilde{S}, \tilde{A}, \tilde{P}_A(\cdot\,|a,\tilde{s}), \tilde{r}(\tilde{s},a)\right)$ as a modified MDP with p order ARP. $\tilde{A} = A$. $\tilde{S}$, $\tilde{P}_A(\cdot\,|a,\tilde{s})$ and $\tilde{r}(\tilde{s},a)$ are defined as follows:

$$\forall\ \tilde{s},\ \tilde{s}' \in \tilde{S}:$$

$$\tilde{s} = \left(s_1, a_1, \cdots, s_p, a_p, s_{p+1}\right),\ a_k \in A, s_k \in S\ \forall k \tag{10}$$

$$\tilde{s}' = \left(s_1', a_1', \cdots, s_p', a_p', s_{p+1}'\right), a_k' \in A, s_k' \in S\ \forall k \tag{11}$$

$$\tilde{P}(\tilde{s}'|a,\tilde{s}) = \begin{cases} \tilde{P}\left(s_{p+1}'\middle|a, s_{p+1}\right),\ if\ s_k' = s_{k+1}, k \le p \\ \qquad\qquad\qquad\quad a_k' = a_{k+1},\ k < p \\ \qquad\qquad\qquad\quad a_p' = a \\ \qquad 0 \qquad\qquad otherwise \end{cases} \tag{12}$$

$$\tilde{r}(\tilde{s},a) = r\left(s_{p+1}, a\right) \tag{13}$$

The applicability of existing learning algorithms on $\widetilde{M}^p$ has already been validated by Korenkevych et al[61]. Therefore, it can be implemented into the present PPO algorithms through augmentation of the state space. In $\widetilde{M}_{ad2}^p$, the original state space $I_{a_d,t}$ in M_{ad2} is then replaced by $\tilde{I}_{a_d,t} = \left(I_{a_d,t-p}, a_{t-p}, \cdots, I_{a_d,t-1}, a_{t-1}, I_{a_d,t}\right)$. A corresponding initial state of $\widetilde{M}_{ad2}^p$ is defined as $\tilde{I}_{a_d,0} = \left(I_{a_d,0}, a_0, \cdots, I_{a_d,0}, a_0, I_{a_d,0}\right)$, where $I_{a_d,0}$ is the initial states of M_{ad2}, a_0 is any element of action set A because it does not affect future transitions and rewards. The sequence $\tilde{I}_{a_d,t}$ considers time-delays and possesses better temporal coherence in the regression process of NNs but does not change the noise proportion in actions. In the present work, we use a first-order ARP in which $p = 1$ and $\tilde{I}_{a_d,1} = \left(I_{a_d,t-1}, a_{t-1}, I_{a_d,t}\right)$ as shown in Fig. 4 (b).

3. Results and Discussion

3.1. The baseline flow

Before the evaluation of the performance of different DRL methods in training agents and implementing active control, a baseline case of uncontrolled flow is performed first. The flow passing around the cylinder without active control (i.e., $Q_i^* = 0,\ i = 1,2,3,4$) is simulated, and the baseline case is monitored for 20 nondimensional time when C_D and C_L begin to vary periodically. Figure 5 (Multimedia view) presents the instantaneous and time averaged vorticity contours and a video showing the flow evolution in the baseline case. Figure 5 (a) is captured at the time corresponding to the minimum C_L during a vortex shedding period, T_S, and the subsequent snapshots are captured at every quarter of a cycle. All results of velocity and vorticity are normalized

by $\overline{U}$ and $\overline{U}/D$ respectively, in this study. In a vortex shedding period, two vortices with an opposite rotation are shed in an alternate manner, leading to periodic pressure variations on the cylinder surface.

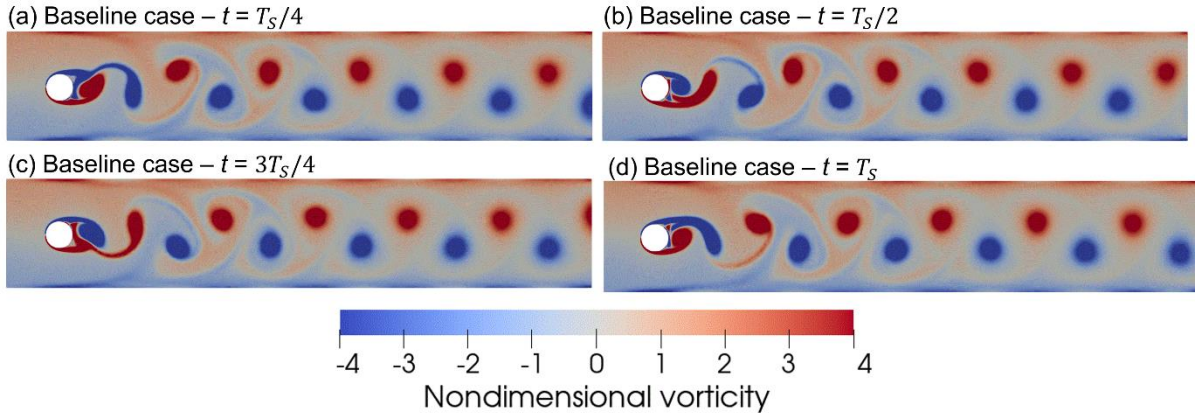

Fig. 5. Contours of instantaneous vorticity at t = (a) $T_S/4$, (b) $T_S/2$, (c) $3T_S/4$ and (d) T_S in the baseline case. (Multimedia view)

Figure 6 illustrates the time series of C_D and C_L in the baseline case and their frequency spectra obtained with Fast Fourier Transform (FFT). The time averaged C_D and C_L over a period are 3.17 and 0.0273, respectively. Note, to enable a direct comparison of the oscillatory properties of the variables of interest, the time averaged value of C_D and C_L have been subtracted before the FFT analysis. The vortex shedding frequency corresponds to $Sth_D = 0.333$. In a single vortex shedding period, C_L experiences one cycle, while C_D experiences two cycles, i.e. $Sth_D = 0.683 \approx 2Sth_S$. Due to a slight difference in the vortices shed from the top and bottom of the cylinder, the two circles of C_D becomes different, resulting in a local maximum obtained at $Sth = Sth_S$ in the spectrum of C_D. Furthermore, C_L is asymmetrical about the time axis.

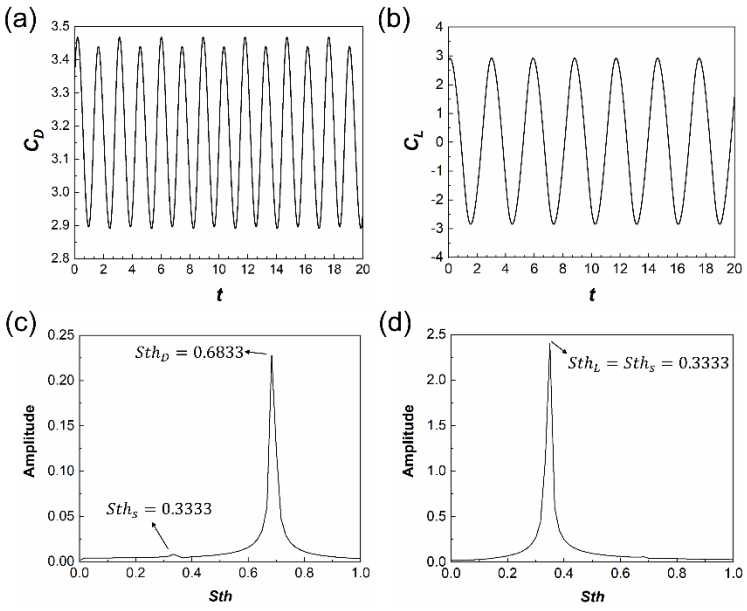

Fig. 6. Temporal variation of C_D (a) and C_L (b) and FFT analyses ofC_D (c) and C_L (d) in the baseline case. In FFT, the time series of C_D and C_L have their temporal mean subtracted.

3.2. Responses of C_D and C_L to different DRL methods

In the present work, four methods are tested and compared for their performance in AFC. The test consists of two phases: training and deterministic control. In the training phase, each method is used to train an agent until the policy learned by the neural networks of the agent reaches the convergence that is the reward has nearly stopped increasing as the episode number increases. After meeting the convergence, all agents have been trained to episode 3200. To accelerate training, each agent simultaneously samples the state-action-reward sets in 32 parallel environments to update a single policy. After training, the last environment is selected for comparison. For the four DRL methods tested, we denote the vanilla PPO algorithms as the PPO method, the PPO method considering delays in the MDP as the DMDP-PPO method, the PPO method employing first-order ARP as the ARP-PPO method, and the PPO algorithms combining MDP with delays and first-order ARP as the ARP-DMDP-PPO method. To avoid random effect, the PPO and ARP-DMDP-PPO methods have been used to perform training five times. While both ARP-PPO and DMDP-PPO methods have been adopted for training two times, due to their high training costs. Averaged performances are plotted for comparison.

In training, the average reward per episode is defined as $\langle r \rangle_e$. The averaged learning curve represented by variations of $\langle r \rangle_e$ with episode number and the confidence interval are illustrated in Fig. 7. All four methods can be applied to achieve robust training. In the initial stage of training, the average rewards of PPO and ARP-DMDP-PPO increase rapidly with the number of episodes. During the main exploration stage, the learning curve of PPO has a large confidence interval corresponding to the training instability exceeding the effect of exploration. The convergence differences by more than 20 episodes may indicate that the learnt policies finally reaching the maximum reward in different training repeats are inconsistent, which may also reflect the insufficient ability of ANN in time series regression for approximating the policy. The learning curves indicate that learning occurs consistently in about 50 episodes, then keeps tuned to 100 episodes to reach fine convergence. The maximum $\langle r \rangle_e$ of the averaged learning curves through the PPO and ARP-DMDP-PPO method is 1.75 and 1.79, respectively. Since the ARP-DMDP-PPO method improves the ANN's regression through embedding the historical data without additional sampling, it requires only a slightly larger training time than the PPO method under the same number of episodes.

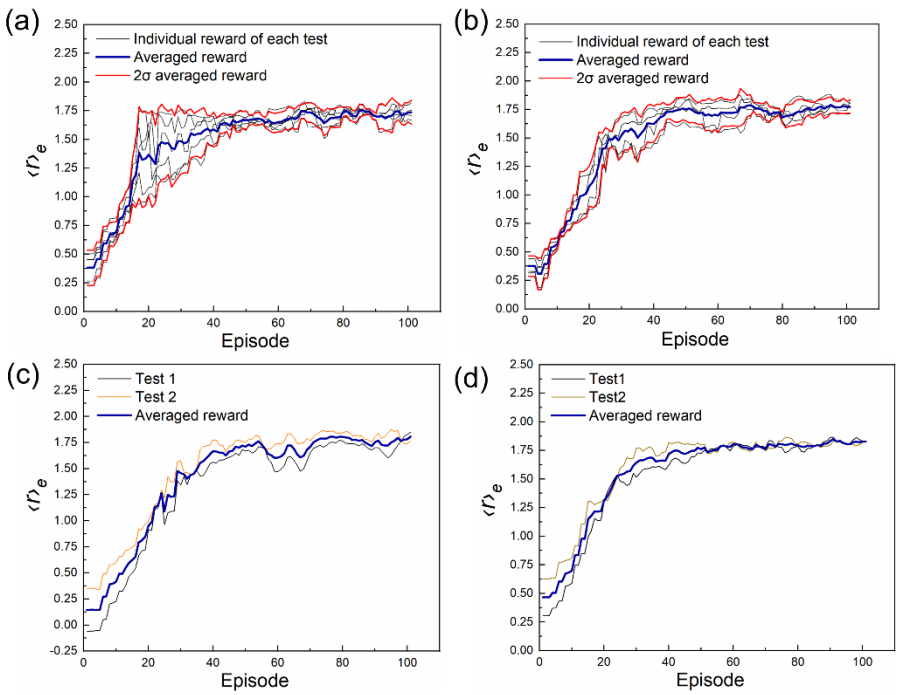

Fig. 7. Learning curves of different methods in the last environment to indicate the training robustness. For each DRL method, the training repeats have been performed using the same hyperparameters but different random seeds. For the PPO (a) and the ARP-DMDP-PPO method (b) with five training repeats, the averaged learning curve and the 95% confidence interval are presented. For the ARP-PPO (c) and DMDP-PPO method (d) with two training repeats, the averaged learning curve is presented.

In the deterministic control phase, each agent selects the best policy (the policy that achieved the greatest reward) in its training history to start control from the same initial state of the baseline case and updates the jet actuation with the same frequency as action updates in training. All results in the control phase are obtained by agents deterministically acting without exploration. The actions are not sampled for training since all agents do not update their policy during the control phase. The nondimensional time t of a complete control process is 120. Figure 8 and 9 presents the temporal variations of C_D and C_L controlled by four agents, respectively. It is noted that all agents produce a significant decrease in C_D within $t = 20$. The PPO and ARP-PPO agents cannot maintain a stationary C_D since their policies are learned by the networks without information of action delays. By contrast, when considering action delays, C_D controlled by the DMDP-PPO and ARP-DMDP-PPO agents has become nearly stationary after $t = 40$. Table 2 summarizes σ_D and σ_L controlled by four agents after $t = 40$.

The drag reduction in control is defined as $(\langle C_D \rangle_b - \langle C_D \rangle_d)/\langle C_D \rangle_b$, where $\langle C_D \rangle_d$ is the time averaged C_D in the DRL cases. Similar to the results of the same flow configuration by Tang et al[48], the PPO agent has obtained a rapid C_D reduction , but then C_D and C_L keep fluctuating significantly and randomly in the remaining time. The

drag reduction by the PPO agent is 36.8%, greater than the drag reduction of 33.1% achieved by Tang et al[48]. This difference of control may indicate the training of the PPO agent in Tang et al[48] can still be tuned by more episodes to reach a better convergence. As the number of episodes increases, the best policy during training will be more probable to close to the optimal policy. Therefore, we choose a higher number of training episodes than Tang et al. based on the available computational resources to avoid the potential influence of insufficient training. The result shows that sufficient training of PPO method still cannot find a stable control strategy to reduce the fluctuations of C_D and C_L in subsection 3.3.

After considering action delays in MDP, the DMDP-PPO agent controls C_D and $|C_L|$ rapidly decrease, and then fluctuate with a much lower σ_D and σ_L than the PPO agent. After the rapid decrease of C_D, the DMDP-PPO agent tends to find a stationary control with small fluctuations at the beginning. But the opposite is true, and C_D and C_L fluctuate in an increasing variance, which indicates the control effects cannot be maintained but gradually worsen as fluctuation accumulate.

Historical states and actions are valuable for the neural networks to estimate advantages A_t and output smooth and temporal coherent actions. Autoregressive models are popular and effective tools to perform a regression of the variable utilizing past values. Here, the ARP-PPO method and PPO method are compared to investigate the effects of first-order ARP. As shown in Fig. 8 (b) and Fig. 9 (b), after the rapid decrease in the initial time, C_D and C_L controlled by the ARP-PPO agent fluctuate in a lower variance than the PPO agent, but they are non-stationary with an unsteady trend. As shown in Table 2, the ARP-PPO agent produces smaller σ_D and σ_L than the PPO agent but larger σ_D and σ_L than the DMDP-PPO agent. The performance of the ARP-PPO agent illustrates the positive effect of ARP, but also indicates that ARP cannot replace DMDP to quantify delays. Therefore, ARPs should be deployed into the DMDP-PPO method instead of directly adopted in the PPO method.

Finally, a hybrid ARP-DMDP-PPO method is used to implement effective control in the AFC system of this study. After the ARP-DMDP-PPO agent performs control, C_D rapidly decreases to reach convergence and begins to change with a small amplitude after $t = 40$. The drag reduction by the ARP-DMDP-PPO agent is 38%, which is higher than the drag reduction obtained by the PPO agent. The ARP-DMDP-PPO agent spends the same time as the PPO agent to reach around the minimum C_D but maintains a steadier flow state with a much lower σ_D and σ_L of 87% and 90%, respectively.

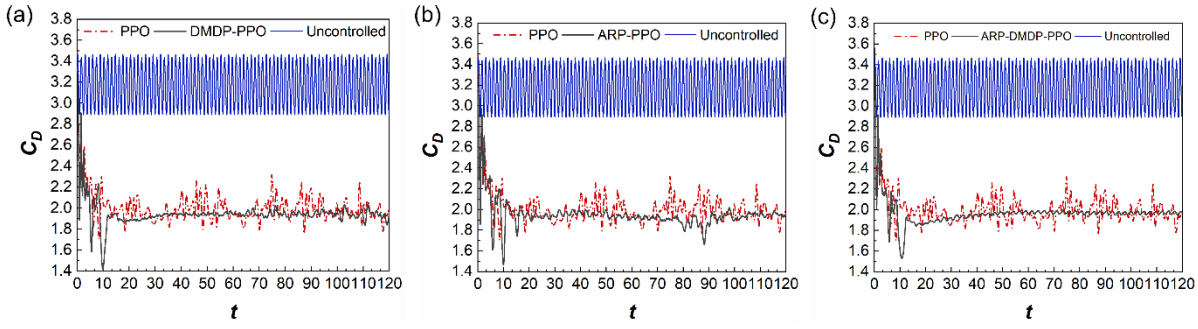

Fig. 8. Real-time control by different agents showing temporal variations in C_D. (a) the DMDP-PPO agent, (b) the ARP-PPO agent, (c) the ARP-DMDP-PPO agent.

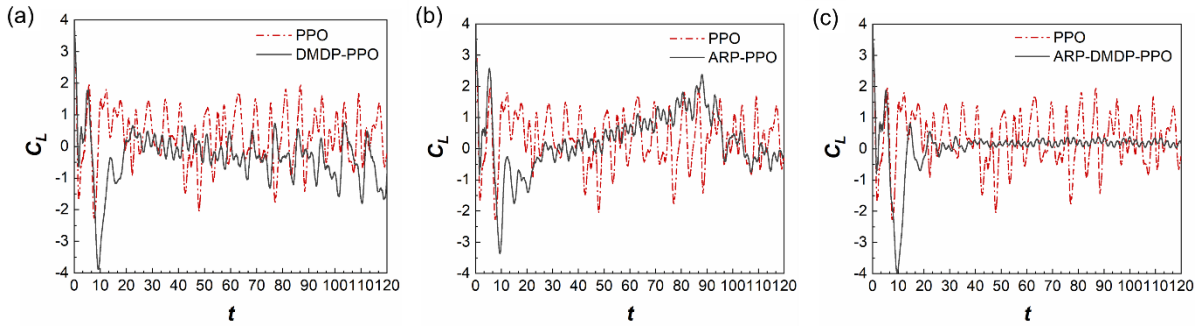

Fig. 9. Real-time control by different agents showing temporal variations in C_L. (a) the DMDP-PPO agent, (b) the ARP-PPO agent, (c) the ARP-DMDP-PPO agent.

Table 2 Comparison of σ_D and σ_L sampled from $t = 40$ to 120 in four controlled cases.

Agent	σ_D	σ_L
PPO	0.0928	0.811
DMDP-PPO	0.0273	0.546
ARP-PPO	0.0464	0.663
ARP-DMDP-PPO	0.0121	0.081

Figure 10 presents the FFT of the time series data of C_D and C_L sampled from $t = 60$ to 120 in the PPO case and ARP-DMDP-PPO case. C_D and C_L are subtracted by their mean value before FFT analysis. Similar to the results of Tang et al.[48], amplitudes of C_D and C_L at their dominant frequencies have been significantly reduced by the PPO agent. However, the PPO agent has introduced a considerable amount of fluctuation of C_D and C_L at the Strouhal numbers lower than Sth_D and Sth_L in the baseline case. Through the modification of jet actuation

controlled by the ARP-DMDP-PPO agent, Sth_D and Sth_L become consistent with Sth_1 corresponding to the actuation frequency presented in Fig. 11. Compared to the baseline case, the ARP-DMDP-PPO agent has led to not only the dominant frequency being shifted to a slightly higher frequency but also its amplitude being reduced substantially. Compared to the PPO agent, the ARP-DMDP-PPO agent suppresses the amplitude of C_D and C_L more effectively and meanwhile avoids introducing unnecessary random interference at other Strouhal numbers.

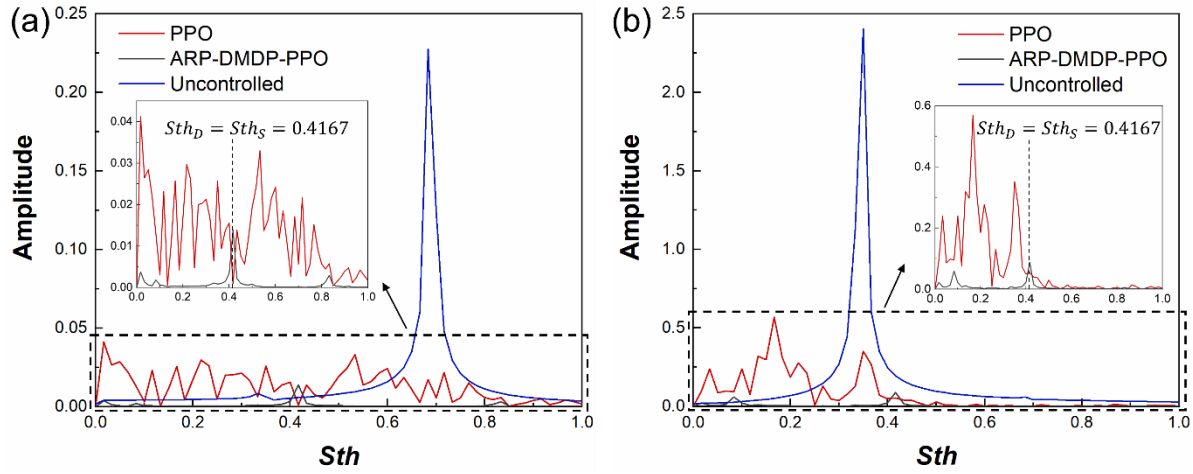

Fig. 10. FFT analysis of C_D and C_L sampled from $t = 60$ to 120 in the baseline case and two controlled cases. Time series of C_D and C_L have been subtracted by their temporal mean. (a) C_D. (b) C_L.

A realistic AFC system usually requires DRL agents to output smooth and temporal coherent jet control due to the physical constraints of jet actuators. Figure 11 compares the mass flow rates of all jets and their FFT between the PPO case and ARP-DMDP-PPO case during control. The FFT is sampled from the time series data of Q_1^* to Q_4^* (nondimensional mass flow rates of four jets) from $t = 60$ to 120 in both cases. To directly compare the pure oscillatory properties of the variables of interest, Q_1^* to Q_4^* are subtracted by their mean value before the FFT analysis. The PPO agent keeps outputting stochastic jet signals during the entire control process. Similar to the control strategy discovered by Rabault et al. at *Re* = 100, the ARP-DMDP-PPO agent changes the flow configuration into a low-drag state through large-amplitude jet actuation. From $t = 20$ to $t = 40$, since the main C_D reduction has already been obtained, each jet signal gradually converges and oscillates to slightly modify the ambient flow. After $t = 40$, all jet signals begin to settle in a stationary trend, resulting in a small pseudo-periodic oscillation of C_D and C_L shown in Fig. 8 (c) and Fig. 9 (c). All jet signals by the ARP-DMDP-PPO agent have almost the same spectral distribution where the amplitudes are mainly concentrated at $Sth_1 = 0.4167$ and $Sth_2 = 0.0833$. On the contrary, the PPO agent has introduced many fluctuations at frequencies lower than Sth_S to each

jet signal. Furthermore, all control signals are inconsistent with each other and have much larger amplitudes than the control signals by the ARP-DMDP-PPO agent.

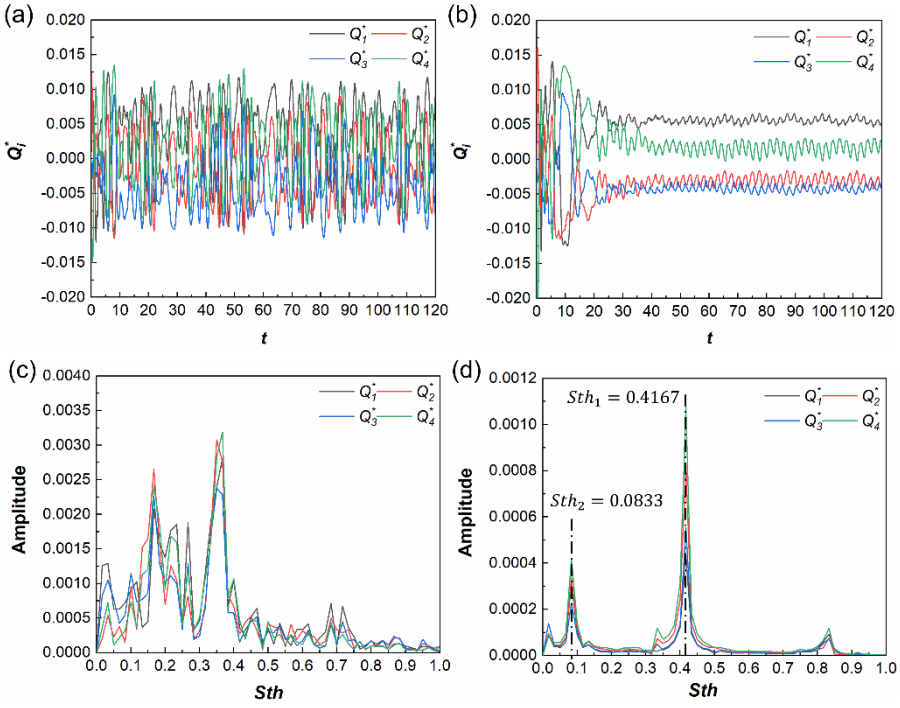

Fig. 11. Temporal variation of Q_1^* to Q_4^* (The nondimensional mass flow rates of four jets) in (a) PPO case and (b) ARP-DMDP-PPO case. FFT analysis of Q_1^* to Q_4^* sampled from $t = 60$ to 120 in (c) PPO case and (d) ARP-DMDP-PPO case, the time series have been subtracted by their temporal mean.

Since the pressure drag and lift are the x-integral and y-integral of cylinder surface pressure in the numerical computation, the optimization mechanism by the agents is specifically analysed from the temporal variation of the pressure distribution on the cylinder surface. Figure 12 presents the time-averaged distribution of pressure and the standard deviation of surface pressure on the cylinder surface through the whole control process. $\theta = 180°$ corresponds to the stagnation point of the cylinder face. $\bar{p}$ is the time-averaged pressure normalized by the dynamic pressure $q = \rho\bar{U}^2/2$, σ_p is the standard deviation of surface pressure. Compared with the baseline case, both agents significantly reduce the absolute value of time-average negative pressure and σ_p on the leeward side of the cylinder, resulting in a large reduction of C_D and σ_D. However, the PPO agent cannot stabilize the controlled flow because its policy lacks the information of delays and historical states-actions sequences. Compared to the baseline case, the large jet actuation in the PPO case increases the pressure fluctuation around $\theta = 105°$ and $255°$ even though it reduces the fluctuation in the flow direction. Furthermore, the PPO agent causes a more asymmetric

pressure distribution at the upper and lower surfaces of the cylinder, resulting in a higher time averaged C_L. On the contrary, the ARP-DMDP-PPO agent performs a more precise control to reduce the pressure difference between the upper and lower surfaces along the horizontal axis of the cylinder. The ARP-DMDP-PPO agent is capable of identifying the asymmetry in the vortex structure in the wake caused by the offset of the cylinder relative to the centerline of inflow shown and outputs an asymmetric jet actuation to maintain a symmetric time averaged pressure distribution. As a result of the asymmetric jet actuation, σ_p is asymmetrically distributed around the cylinder. Compared to the PPO agent, the ARP-DMDP-PPO agent reduces σ_p at most angles especially at $\theta = 105°$ and $255°$, resulting in a lower fluctuation of drag and lift.

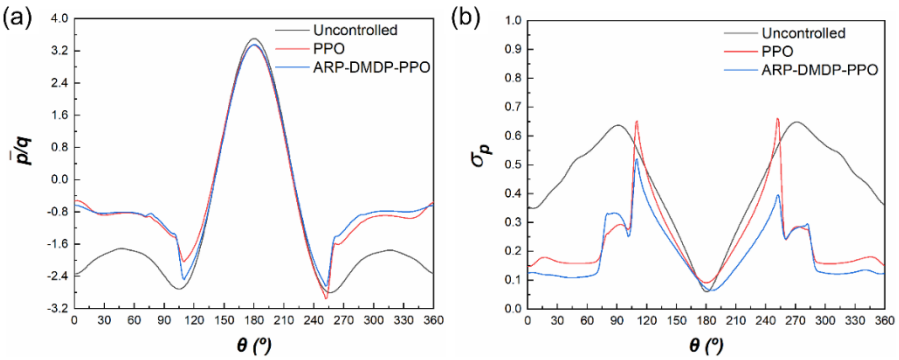

Fig. 12. Distribution of $\bar{p}/q$ and σ_p on the cylinder surface calculated over the whole control process. (a) $\bar{p}/q$. (b) σ_p.

3.3. Effects of flow control on vortex formation

The effects of controlling jets on the vortex shedding are further analysed in this section. Figure 13 (Multimedia view) presents the instantaneous vorticity contours and a video of the flow evolution during the entire control process in the baseline, PPO and ARP-DMDP-PPO case. In the ARP-DMDP-PPO case, the first image is captured at a time corresponding to the minimum C_L in a single period T_{S_ADP}, and the subsequent figures are captured every quarter of a cycle. The vorticity contours of the baseline case have been presented in Fig. 13 (a1) – (a4). It can be seen clearly that in the baseline case, formation of the shedding vortices takes place within a distance much closer to the cylinder than in the controlled cases and the strength of vortices in remains high towards the computational domain. In the PPO case, the formation zone has clearly been extended further downstream compared to the baseline case and the vortices further downstream are much weaker. The ARP-DMDP-PPO agent seems to produce the longest and most steady formation zone with the vortex shedding taking place at the end of it. The location of vortex shedding is closer to the centreline of the cylinder and the width of the vortex street behind it appears narrower than in the other two cases. A longer and steadier formation zone is

expected to lead to a less fluctuating pressure in the near wake resulting in a lower vortex-induced drag. Similar phenomena were reported by researchers using passive flow control techniques[63,64], such as fairings and splitter plates.

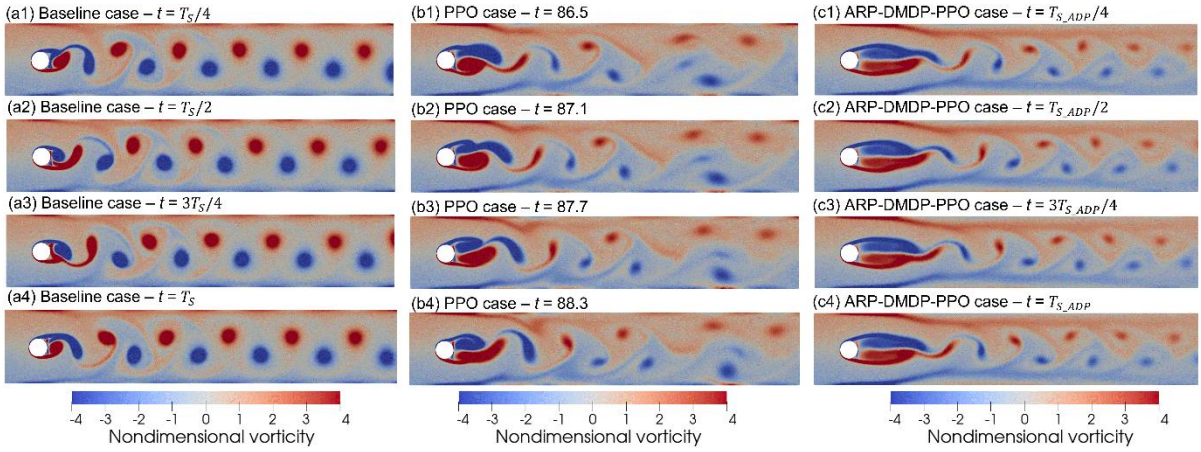

Fig. 13. Contours of instantaneous vorticity at (a1) - (a4): $t = T_S/4, T_S/2, 3T_S/4$ and T_S in the baseline case, (b1) - (b4): t = 86.5, 87.1, 87.7 and 88.3 in the PPO case and (c1) - (c4): $t = T_{S\_ADP}/4, T_{S\_ADP}/2, 3T_{S\_ADP}/4$ and T_{S_ADP} in the ARP-DMDP-PPO case. (Multimedia view)

Figure 14 presents the contours of time-averaged velocity magnitude and vorticity over the entire control process (t=120) in the baseline case ($40T_S$), the PPO case and the ARP-DMDP-PPO case ($50T_{S\_ADP}$). In comparison to the baseline case, with the jet control the recirculation bubble in the near wake is extended substantially in length, resulting in a reduced pressure drop shown in Fig. 12 (a). Compared to the PPO agent, the ARP-DMDP-PPO agent gives arise to a more streamlined recirculation zone within which the magnitude of vorticity is smaller and the regions of high vorticity are located further away from the cylinder, leading to a steady pressure distribution on the cylinder surface.

Figure 15 shows the streamwise $\overline{u_x'u_x'}$ over the entire control process in the three cases. The location of peak streamwise $\overline{u_x'u_x'}$ gives of good indication of where the vortex formation zone ends. In the baseline case, the length of vortex formation zone is $L_f = 0.98$. In the PPO controlled case, the vortex formation zone is extended to $L_f = 2.34$ and the distance between the centres of vortex formation region and the central line of cylinder is significantly reduced. Compared to the PPO case, the vortex formation length L_f in the ARP-DMDP-PPO case is even greater and the vortex formation region of two side eddies almost overlap, resulting in a longer and steadier recirculation zone.

In summary, in comparison to the PPO agent, the ARP-DMDP-PPO agent is more effective in extending the length of the recirculation zone and moving the location of vortex shedding substantially downstream. This results in a much steadier near wake flow and the significant reduction of σ_D and σ_L as presented in Table 2.

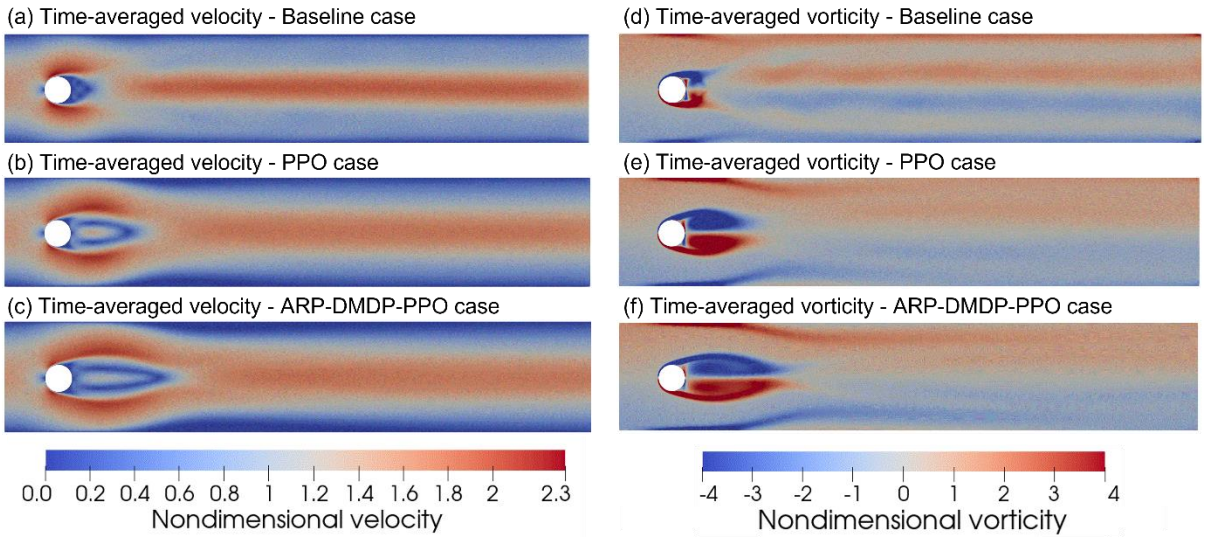

Fig. 14. Contours of time-averaged velocity over the entire control process in (a) Baseline case; (b) PPO case; (c) ARP-DMDP-PPO case. Contours of time-averaged vorticity over the entire control process in (d) baseline case; (e) PPO case; (f) ARP-DMDP-PPO case.

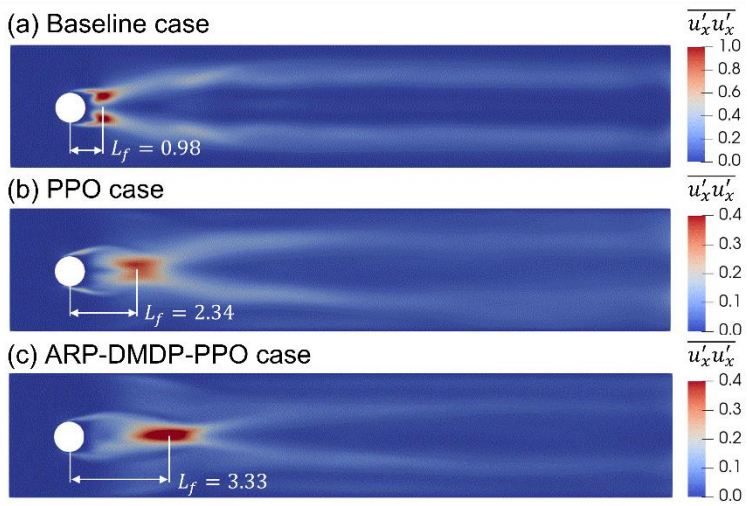

Fig. 15. Contours of $\overline{u_x'u_x'}$ over the entire control process in (a) baseline case, (b) PPO case and (c) ARP-DMDP-PPO case.

4. Conclusions

In this study, we have developed a new hybrid DRL method which combines a Markov decision process (MDP) with time delays with the use of a first-order autoregressive policy (ARP), i.e. the ARP-DMDP-PPO

method. This hybrid DRL method is applied to control the vortex shedding process from a 2D circular cylinder using four synthetic jet actuators at a freestream Reynolds number of 400. The aim of this work is to improve the time series regression of ANN and control stability by reducing the large fluctuations in lift and drag accompanying the drag reduction; a problem which remains unsolved in the existing work and tends to escalate as the Reynolds number increases.

The Markov decision process (MDP) with time delays is introduced to quantify the action delays in the DRL process due to the time elapse between actuation and response of flow. A cross-correlation analysis has been performed between signals of the actuation jets and the velocity probes in the wake to validate the action delay in the decision process of DRL. The ARP-DMDP-PPO method has trained an agent successfully to implement a smooth and temporal coherent control. Compared to the standard DRL method, this method utilizes the historical samples without additional sampling in training and it is shown to be capable of reducing the magnitude of drag and lift fluctuations by approximately 90% while achieving a similar level of drag reduction in the deterministic control at the same actuation frequency.

In comparison to the existing the standard PPO method, this new method has yielded a stable and coherent control which results in a steadier and more elongated vortex formation zone behind the cylinder hence a much weaker vortex shedding process and less fluctuating lift and drag forces. This study demonstrates the necessity of including a physics-informed time delay in the MDP and the benefits of introducing ARPs to achieve a robust and temporal-coherent control of unsteady forces in active flow control. The pseudo-periodic control signals that the control strategy produced by the DRL algorithms are also more regular making them more applicable to AFC in real settings. Reduction of high-level temporal fluctuations in drag and lift will help to decrease dynamic loads and structural fatigue leading to an improved structural durability and operational safety which will benefit many industrial applications.

Acknowledgement

Yiqian Mao would like to acknowledge the Scholarship awarded from both the University of Manchester and the China Scholarship Council to his PhD studies. The authors would like to thank Dr. Jean Rabault (University of Oslo, Oslo, Norway) and Mr. Hongwei Tang (Nanjing University of Aeronautics and Astronautics, Nanjing, China) for making their open-source codes for deep reinforcement learning and numerical simulation available online at https://github.com/jerabaul29/Cylinder2DFlowControlDRLParallel and https://github.com/thw1021/Cylinder2DFlowControlGeneral.

Author declarations

Conflict of interest

The authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

[1] S. Zhong, M. Jabbal, H. Tang, L. Garcillan, F. Guo, N. Wood, and C. Warsop, "Towards the design of synthetic-jet actuators for full-scale flight conditions : Part 1: The fluid mechanics of synthetic-jet actuators," Flow, Turbul. Combust. **78**, 283-307 (2007).

[2] X. Li, F. Wu, Y. Tao, M. Yang, R. Newman, and D. Vainchtein, "Numerical study of the air flow through an air-conditioning unit on high-speed trains," J. Wind Eng. Ind. Aerodyn. **187**, 26 (2019).

[3] M. Urquhart, M. Varney, S. Sebben, and M. Passmore, "Drag reduction mechanisms on a generic square-back vehicle using an optimised yaw-insensitive base cavity," Exp. Fluids **62**(241), (2021).

[4] J.C. Schulmeister, J.M. Dahl, G.D. Weymouth, and M.S. Triantafyllou, "Flow control with rotating cylinders," J. Fluid Mech. **825**, 743 (2017).

[5] H. Choi, W.P. Jeon, and J. Kim, "Control of flow over a bluff body," Annu. Rev. Fluid Mech. **40**, 113 (2008).

[6] C. Zhu, J. Chen, J. Wu, and T. Wang, "Dynamic stall control of the wind turbine airfoil via single-row and double-row passive vortex generators," Energy **189**, 116272 (2019).

[7] S.S. Collis, R.D. Joslin, A. Seifert, and V. Theofilis, "Issues in active flow control: theory, control, simulation, and experiment," Prog. Aerosp. Sci. **40**(4–5), 237 (2004).

[8] T. Shaqarin, P. Oswald, B.R. Noack, and R. Semaan, "Drag reduction of a D-shaped bluff-body using linear parameter varying control," Phys. Fluids **33**, (2021).

[9] B. Plumejeau, S. Delprat, L. Keirsbulck, M. Lippert, and W. Abassi, "Ultra-local model-based control of the square-back Ahmed body wake flow," Phys. Fluids **31**(8), 085103 (2019).

[10] D. Gao, H. Meng, Y. Huang, G. Chen, and W.L. Chen, "Active flow control of the dynamic wake behind a square cylinder using combined jets at the front and rear stagnation points," Phys. Fluids **33**(4), 047101 (2021).

[11] M. Gadalla, M. Cianferra, M. Tezzele, G. Stabile, A. Mola, and G. Rozza, "On the comparison of LES data-

driven reduced order approaches for hydroacoustic analysis," Comput. Fluids **216**, 104819 (2021).

[12] A. Towne, O.T. Schmidt, and T. Colonius, "Spectral proper orthogonal decomposition and its relationship to dynamic mode decomposition and resolvent analysis," J. Fluid Mech. **847**, 821 (2018).

[13] S.L. Brunton, B.R. Noack, and P. Koumoutsakos, "Machine Learning for Fluid Mechanics," Annu. Rev. Fluid Mech. **52**, 477 (2020).

[14] S. Shimomura, S. Sekimoto, A. Oyama, K. Fujii, and H. Nishida, "Closed-loop flow separation control using the deep q network over airfoil," AIAA J. **58**(10), 4260 (2020).

[15] Y. LI, J. CHANG, C. KONG, and W. BAO, "Recent progress of machine learning in flow modeling and active flow control," Chinese J. Aeronaut. **35**(4), 14 (2021).

[16] K. Fukami, K. Fukagata, and K. Taira, "Assessment of supervised machine learning methods for fluid flows," Theor. Comput. Fluid Dyn. **34**(4), 497 (2020).

[17] P. Zwintzscher, F. Goméz, and H.M. Blackburn, "Data-driven control of the turbulent flow past a cylinder," J. Fluids Struct. **89**, 232 (2019).

[18] H. Xu, W. Zhang, and Y. Wang, "Explore missing flow dynamics by physics-informed deep learning: The parameterized governing systems," Phys. Fluids **33**(9), 095116 (2021).

[19] E. Kharazmi, D. Fan, Z. Wang, and M.S. Triantafyllou, "Inferring vortex induced vibrations of flexible cylinders using physics-informed neural networks," J. Fluids Struct. **107**, 103367 (2021).

[20] G. Minelli, T. Dong, B.R. Noack, and S. Krajnović, "Upstream actuation for bluff-body wake control driven by a genetically inspired optimization," J. Fluid Mech. **893**, A1 (2020).

[21] F. Ren, C. Wang, and H. Tang, "Active control of vortex-induced vibration of a circular cylinder using machine learning," Phys. Fluids **31**(9), 093601 (2019).

[22] G. Ortali, N. Demo, and G. Rozza, "A Gaussian Process Regression approach within a data-driven POD framework for engineering problems in fluid dynamics," Math. Eng. **4**(3), 1 (2022).

[23] N. Alhazmi, Y. Ghazi, M. Aldosari, R. Tezaur, and C. Farhat, "Training a neural-network-based surrogate model for aerodynamic optimization using a gaussian process", in AIAA Scitech 2021 Forum (AIAA, 2021), pp. 1–9.

[24] I. Kang, K.H. Lee, J.H. Lee, and J.W. Moon, "Artificial neural network-based control of a variable refrigerant flow system in the cooling season," Energies **11**(7), 1643 (2018).

[25] W. Zhang, Y. Yu, Y. Qi, F. Shu, and Y. Wang, "Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning," Transp. A Transp. Sci. **15**(2), 1688 (2019).

[26] H. Eivazi, H. Veisi, M.H. Naderi, and V. Esfahanian, "Deep neural networks for nonlinear model order reduction of unsteady flows," Phys. Fluids **32**(10), 105104 (2020).

[27] B.Z. Han and W.X. Huang, "Active control for drag reduction of turbulent channel flow based on convolutional neural networks," Phys. Fluids **32**(9), 095108 (2020).

[28] C. Lee, J. Kim, D. Babcock, and R. Goodman, "Application of neural networks to turbulence control for drag reduction," Phys. Fluids **9**(6), 1740 (1997).

[29] K. Fukami, T. Nakamura, and K. Fukagata, "Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data," Phys. Fluids **32**(9), 095110 (2020).

[30] M. Morimoto, K. Fukami, K. Zhang, A.G. Nair, and K. Fukagata, "Convolutional neural networks for fluid flow analysis: toward effective metamodeling and low dimensionalization," Theor. Comput. Fluid Dyn. **35**, 633 (2021).

[31] J.Z. Peng, S. Chen, N. Aubry, Z.H. Chen, and W.T. Wu, "Time-variant prediction of flow over an airfoil using deep neural network," Phys. Fluids **32**(12), 123602 (2020).

[32] T. Nakamura, K. Fukami, K. Hasegawa, Y. Nabae, and K. Fukagata, "Convolutional neural network and long short-term memory based reduced order surrogate for minimal turbulent channel flow," Phys. Fluids **33**(2), 025116 (2021).

[33] K. Hasegawa, K. Fukami, T. Murata, and K. Fukagata, in *Fluid Dyn. Res.* (2020), p. 065501.

[34] R.S. Sutton, *Reinforcement Learning : An Introduction*, Second edition. (The MIT Press, Cambridge, MA, 2018).

[35] J. Rabault, F. Ren, W. Zhang, H. Tang, and H. Xu, "Deep reinforcement learning in fluid mechanics: A promising method for both active flow control and shape optimization," J. Hydrodyn. **32**(2), 234 (2020).

[36] F. Ren, H. bao Hu, and H. Tang, "Active flow control using machine learning: A brief review," J. Hydrodyn. **32**(2), 247 (2020).

[37] P. Garnier, J. Viquerat, J. Rabault, A. Larcher, A. Kuhnle, and E. Hachem, "A review on deep reinforcement learning for fluid mechanics," Comput. Fluids **225**, 104973 (2021).

[38] Y.-Z. Wang, Y.-F. Mei, N. Aubry, Z. Chen, P. Wu, and W.-T. Wu, "Deep reinforcement learning based synthetic jet control on disturbed flow over airfoil," Phys. Fluids **34**(3), 033606 (2022).

[39] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," arXiv:1707.06347, (2017).

[40] S. Fujimoto, H. Van Hoof, and D. Meger, "Addressing Function Approximation Error in Actor-Critic Methods,"

in *35th Int. Conf. Mach. Learn. ICML 2018* (PMLR, 2018), pp. 1587–1596.

[41] D. Fan, L. Yang, Z. Wang, M.S. Triantafyllou, and G.E. Karniadakis, "Reinforcement learning for bluff body active flow control in experiments and simulations," Proc. Natl. Acad. Sci. U. S. A. **117**(42), 26091 (2020).

[42] F. Ren, C. Wang, and H. Tang, "Bluff body uses deep-reinforcement-learning trained active flow control to achieve hydrodynamic stealth," Phys. Fluids **33**(9), 093602 (2021).

[43] R. Paris, S. Beneddine, and J. Dandois, "Robust flow control and optimal sensor placement using deep reinforcement learning," J. Fluid Mech. **913**, A25 (2021).

[44] J. Li and M. Zhang, "Reinforcement-learning-based control of confined cylinder wakes with stability analyses," J. Fluid Mech. **932**, A44 (2022).

[45] J. Dai, P. Liu, Q. Qu, L. Li, and T. Niu, "Aerodynamic optimization of high-lift devices using a 2D-to-3D optimization method based on deep reinforcement learning and transfer learning," Aerosp. Sci. Technol. **121**, 107348 (2022).

[46] J. Rabault, M. Kuchta, A. Jensen, U. Réglade, and N. Cerardi, "Artificial neural networks trained through deep reinforcement learning discover control strategies for active flow control," J. Fluid Mech. **865**, 281 (2019).

[47] J. Rabault and A. Kuhnle, "Accelerating deep reinforcement learning strategies of flow control through a multi-environment approach," Phys. Fluids **31**(9), 094105 (2019).

[48] H. Tang, J. Rabault, A. Kuhnle, Y. Wang, and T. Wang, "Robust active flow control over a range of Reynolds numbers using an artificial neural network trained through deep reinforcement learning," Phys. Fluids **32**(5), 053605 (2020).

[49] M.M. Zdravkovich, *Flow around Circular Cylinders : A Comprehensive Guide through Flow Phenomena, Experiments, Applications, Mathematical Models, and Computer Simulations* (Oxford University Press, Oxford, 1997).

[50] F. Ren, J. Rabault, and H. Tang, "Applying deep reinforcement learning to active flow control in weakly turbulent conditions," Phys. Fluids **33**(3), 037121 (2021).

[51] Q. Zhang, C. Su, M. Tsubokura, Z. Hu, and Y. Wang, "Coupling analysis of transient aerodynamic and dynamic response of articulated heavy vehicles under crosswinds," Phys. Fluids **34**(1), 017106 (2022).

[52] X. Chen, T. Liu, Y. Xia, W. Li, Z. Guo, Z. Jiang, and M. Li, "The evolution of airtight performance for a high-speed train during its long-term service," Measurement **177**, 109319 (2021).

[53] S. Meng, D. Zhou, X. Xiong, and G. Chen, "The Effect of the Nose Length on the Aerodynamics of a High-Speed Train Passing Through a Noise Barrier," Flow, Turbul. Combust. 2021 1082 **108**, 411 (2021).

[54] K. Goda, "A multistep technique with implicit difference schemes for calculating two- or three-dimensional cavity flows," J. Comput. Phys. **30**(1), 76 (1979).

[55] A. Logg, K.A. Mardal, and G. Wells, *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book (Lecture Notes in Computational Science and Engineering)* (Springer, 2012).

[56] A. Kuhnle, M. Schaarschmidt, and K. Fricke, Tensorforce: a TensorFlow library for applied reinforcement learning, https://github.com/tensorforce/tensorforce.

[57] K. V. Katsikopoulos and S.E. Engelbrecht, "Markov decision processes with delays and asynchronous cost collection," IEEE Trans. Automat. Contr. **48**(4), 568 (2003).

[58] M.S. Bloor, "The transition to turbulence in the wake of a circular cylinder," J. Fluid Mech. **19**(2), 290 (1964).

[59] G. Chopra and S. Mittal, "Drag coefficient and formation length at the onset of vortex shedding," Phys. Fluids **31**(1), 013601 (2019).

[60] M.S. Bloor and J.H. Gerrard, "Measurements on turbulent vortices in a cylinder wake," Proc. R. Soc. London, Ser. A **294**(1438), 319–342 (1966).

[61] D. Korenkevych, A. Rupam Mahmood, G. Vasan, and J. Bergstra, "Autoregressive policies for continuous control deep reinforcement learning," in *IJCAI Int. Jt. Conf. Artif. Intell.* (IJCAI, 2019), pp. 2754–2762.

[62] M.L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming* (John Wiley & Sons, 2014).

[63] K. Liu, J. Deng, and M. Mei, "Experimental study on the confined flow over a circular cylinder with a splitter plate," Flow Meas. Instrum. **51**, 95 (2016).

[64] Y.Z. Law and R.K. Jaiman, "Wake stabilization mechanism of low-drag suppression devices for vortex-induced vibration," J. Fluids Struct. **70**, 428 (2017).