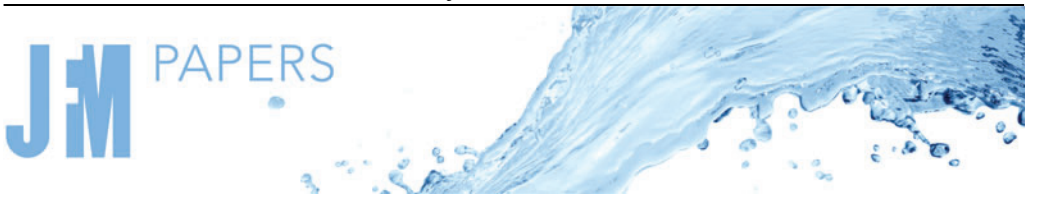
J. Fluid Mech. (2024), *vol*. 988, A4, doi:10.1017/jfm.2024.333



Dynamic feature-based deep reinforcement learning for flow control of circular cylinder with sparse surface pressure sensing

Qiulei Wang[1], Lei Yan[1,†], Gang Hu[1,2,3,†], Wenli Chen[3,4], Jean Rabault[5] and Bernd R. Noack[6]

[1]School of Civil and Environmental Engineering, Harbin Institute of Technology, Shenzhen 518055, PR China

[2]Guangdong Provincial Key Laboratory of Intelligent and Resilient Structures for Civil Engineering, Harbin Institute of Technology, Shenzhen 518055, PR China

[3]Guangdong-Hong Kong-Macao Joint Laboratory for Data-Driven Fluid Mechanics and Engineering Applications, Harbin Institute of Technology, Shenzhen 518055, PR China

[4]Key Laboratory of Smart Prevention and Mitigation of Civil Engineering Disasters, the Ministry of Industry and Information Technology, Harbin Institute of Technology, Harbin 150090, PR China

[5]Independent Researcher, Oslo 0376, Norway

[6]School of Mechanical Engineering and Automation, Harbin Institute of Technology, Shenzhen 518055, PR China

(Received 5 June 2023; revised 23 February 2024; accepted 1 April 2024)

This study proposes a self-learning algorithm for closed-loop cylinder wake control targeting lower drag and lower lift fluctuations with the additional challenge of sparse sensor information, taking deep reinforcement learning (DRL) as the starting point. The DRL performance is significantly improved by lifting the sensor signals to dynamic features (DFs), which predict future flow states. The resulting DF-based DRL (DF-DRL) automatically learns a feedback control in the plant without a dynamic model. Results show that the drag coefficient of the DF-DRL model is 25 % less than the vanilla model based on direct sensor feedback. More importantly, using only one surface pressure sensor, DF-DRL can reduce the drag coefficient to a state-of-the-art performance of approximately 8 % at Reynolds number (Re) = 100 and significantly mitigates lift coefficient fluctuations. Hence, DF-DRL allows the deployment of sparse sensing of the flow without degrading the control performance. This method also exhibits strong robustness in flow control under more complex flow scenarios, reducing the drag coefficient by 32.2 % and 46.55 % at $Re = 500$ and 1000, respectively. Additionally, the drag coefficient decreases by 28.6 % in a three-dimensional turbulent flow at $Re =$ 10 000. Since surface pressure information is more straightforward to measure in realistic scenarios than flow velocity information, this study provides a valuable reference for

† Email addresses for correspondence: 180410212@stu.hit.edu.cn, hugang@hit.edu.cn

experimentally designing the active flow control of a circular cylinder based on wall pressure signals, which is an essential step toward further developing intelligent control in a realistic multi-input multi-output system.

Key words: drag reduction, machine learning

1. Introduction

Flow control has been a popular research area of great academic and industrial interest, which can be divided into passive flow control and active flow control on the basis of whether external energy input is necessary. Passive control has the advantages of requiring no energy, being easy to set up and having low cost, but if the actual situation of the flow field differs from that expected, it is often difficult to achieve the best effect. Active control can be divided into open-loop control and closed-loop control according to whether it is necessary to obtain feedback information from the flow field and to adjust the flux of the actuator (Duriez, Brunton & Noack 2017). It has been found that, compared with open-loop control, closed-loop active control has a robust adaptive ability, which can give full play to the potential of the actuator with a small amount of energy input. For example, Korkischko & Meneghini (2012) has presented a sequence of experiments on the flow around a cylinder, which included moving surfaces boundary control. This method involved injecting momentum into the boundary layer of the cylinder by using two rotating cylinder modules, thereby delaying separation and preventing vortices. Nevertheless, the complexity of the nonlinear Navier–Stokes equations leads to a flow field with high dimensionality and multimodal characteristics, thus making it challenging to devise effective real-time closed-loop active flow control procedures.

In recent years, machine learning has made significant advances, and active flow control is becoming more effective and intelligent (Brunton & Noack 2015). One of the earliest machine learning techniques applied in this field is genetic programming (GP). Genetic programming uses a population of computer programs as potential solutions to a problem. The programs evolve using genetic operators such as mutation and cross-over, and the fittest ones produce the next generation of solutions. Gautier *et al.* (2015) applied GP to search explicit control laws to reduce the recirculation zone behind a backward-facing step. Zhou *et al.* (2020) applied the linear GP to control the dynamics of a turbulent jet and discovered novel wake patterns. Ren, Wang & Tang (2019) adopted GP-identified control laws to successfully suppress vortex-induced vibrations in a numerical simulation environment. Pino *et al.* (2023) demonstrated that many techniques from the machine learning (ML) family could be applied to active flow control (AFC) tasks, from GP to Bayesian optimization (Blanchard *et al.* 2021; Ji *et al.* 2022), Lipschitz global optimization (Pintér 1995) and reinforcement learning (Mnih *et al.* 2015; Rabault *et al.* 2020; Wang *et al.* 2022*b*), and that these methods have trade-offs relative to each other.

Artificial neural networks (ANNs) can also be trained to learn complex patterns and relationships in fluid dynamics data and to generate control strategies that optimize fluid manipulation, which can be used for various tasks, including predicting fluid flow patterns, controlling robotic arms that manipulate fluids and optimizing the design of microfluidic devices. Lee *et al.* (1997) applied an adaptive controller based on a neural network for turbulent channel flow, demonstrating a simple control scheme that reduced skin friction by up to 20 % and produced an optimum wall blowing and suction proportional to a local sum of wall-shear stress.

The rapid development of deep reinforcement learning (DRL), which is effective at interacting with complex nonlinear environments, has brought new ideas to the above flow control problems. Previous studies have shown that DRL can effectively acquire control strategies in high-dimensional, nonlinear and other complex environments. Suppose that DRL is employed to interact with a flow control environment. In such a scenario, it is essential for the closed-loop flow control method to establish the control law based on the learned strategy after continuous trial and error and adjustment of the optimization strategy.

Rabault *et al.* (2019) made a groundbreaking contribution by introducing DRL to AFC for the first time by applying DRL to blunt body drag reduction at Reynolds number (Re) = 100 and successfully demonstrated a closed-loop active control strategy that could achieve stable drag reduction of approximately 8 % by using proximal strategy optimization method. This study uses the velocity measured by 151 sensors around the cylinder and in the downstream flow field (each sensor collects both the flow lateral velocity) as the feedback signal. To investigate higher Reynolds numbers, Ren, Rabault & Tang (2021*a*) applied the lattice–Boltzmann method to establish a computational fluid dynamics (CFD) environment with weak turbulence conditions, and a Reynolds number of up to 1000 was effectively controlled. The results show that the DRL agent could find an effective feedback law and achieve a drag reduction of more than 30 %. Applications in even more chaotic conditions, corresponding to a two-dimensional (2-D) cylinder at a Reynolds number $Re = 2000$, have recently been presented by Varela *et al.* (2022), highlighting that DRL-based controllers can learn drastically different control laws as the underlying dominant physics is changed. In another study, Tang *et al.* (2020) placed four synthetic jets on the lower and upper sides symmetrically for AFC of the cylinder.

Apart from the jet actuator, two small rotating cylinders were placed obliquely behind the main cylinder at a Reynolds number of 240 in the Xu *et al.* (2020) study. The rotational speed of the small cylinders was controlled by a DRL agent. This experimental set-up aimed to investigate the potential of wake stabilization using DRL-controlled rotating control cylinders. The study findings were later confirmed by Fan *et al.* (2020), who experimentally verified the effectiveness and feasibility of this approach.

In addition to its application in the field of AFC tasks, researchers have also aimed to utilize the DRL approach to achieve other objectives. These objectives include reducing the energy expenditure of the follower (Novati *et al.* 2017; Verma, Novati & Koumoutsakos 2018), mitigating vortex-induced vibration (Ren *et al.* 2019; Mei *et al.* 2021; Ren, Wang & Tang 2021*b*; Zheng *et al.* 2021), shape optimization (Garnier *et al.* 2021; Li, Snaiki & Wu 2021; Viquerat *et al.* 2021) or the control of turbulent channel flows (Guastoni *et al.* 2023). As the field of DRL applications for fluid mechanics is evolving fast, we refer the reader curious for more details to any of the recent reviews on the topic, e.g. Garnier *et al.* (2021) and Vignon, Rabault & Vinuesa (2023).

Most of the aforementioned studies have collected state information using a large number of velocity sensors in the wake region, which poses significant challenges for practical structural flow fields. For instance, in the case of vehicles and high-rise buildings, it would be more convenient and easier to maintain and deploy surface pressure sensors. However, compared with the state in the wake region, the pressure on the surface of the structure may have insufficient characteristic information, making it difficult for the DRL agent to estimate the state of the entire flow field. This will result in typical reinforcement learning methods being unable to learn effective control strategies. Based on this fact, we introduce the dynamic feature (DF) lifting approach to DRL and propose the DF-DRL method. In the case of flow around the cylinder, this method can significantly enhance the

convergence performance of the DRL algorithm, enabling it to achieve a drag-reduction effect that is almost consistent with the benchmark (147 velocity sensors deployed in the wake region) with a reduction of 99.3 % of the sensor quantity.

In the present study, we utilized the DRLinFluids package (Wang *et al.* 2022*a*) to train a DRL agent and execute interactions. The package leverages the Tensorforce (Schaarschmidt *et al.* 2018) and Tianshou (Weng *et al.* 2022) packages to provide DRL algorithm libraries, and OpenFOAM (Jasak, Jemcov & Tukovic 2007) as the CFD interaction environment. Firstly, we compare the performance of vanilla DRL and DF-DRL-based plants to a benchmark case study of flow around a circular cylinder with a Reynolds number of 100. Subsequently, we varied the number of pressure surface sensors to validate the effectiveness of the proposed method. Finally, we trained a DF-DRL agent with a single surface pressure sensor and deployed it to more complex flows at higher Reynolds numbers to illustrate the robustness of the approach.

2. Active flow control system with DRL-based jet actuators

The present section is partitioned into two components: (i) an illustration of the DRL algorithm, especially for the soft actor-critic (SAC) method, which will be used as the DRL part in the whole study; (ii) a detailed introduction of the DF-based DRL framework, including the DF lifting and the coupling with the flow simulation.

2.1. *Deep reinforcement learning*

Deep reinforcement learning is a powerful method of optimal control based on a parameterized policy, commonly called an agent, that learns through trial and error. In the context of CFD, the environment can be modelled as the flow over a circular cylinder. During the optimization procedure, the DRL agent interacts with this environment to generate experiences according to the current policy. These experiences are then cached in a buffer and used by the training algorithm to improve the policy. This iterative process is repeated until the agent can yield a control strategy that satisfies the desired performance criteria. Thus, DRL has the potential to revolutionize the field of fluid mechanics by enabling the discovery of previously unknown control strategies that can enhance the performance of fluid systems.

There are several types of DRL algorithms (François-Lavet *et al.* 2018). One of the most popular types of DRL algorithms is Q-learning (Mnih *et al.* 2013; Bellemare, Dabney & Munos 2017; Andrychowicz *et al.* 2018), which uses a neural network to approximate the optimal action-value function, and updates the network's weights using the Bellman equation to minimize the difference between the predicted and actual reward. Another type of algorithm is employed in policy gradient methods (Schulman *et al.* 2015, 2017; Mnih *et al.* 2016), which directly optimize the agent's policy to maximize the expected reward and often use techniques like Monte Carlo sampling or trust region optimization. Actor-critic methods (Fujimoto, van Hoof & Meger 2018; Haarnoja *et al.* 2018; Lillicrap *et al.* 2019) combine the advantages of both Q-learning and policy gradient methods by simultaneously learning a value function and a policy. Another type of DRL algorithm is model-based reinforcement learning (Silver *et al.* 2017; Ha & Schmidhuber 2018; Weber *et al.* 2018), which involves learning a model of the environment dynamics and using it to plan actions. Model-based algorithms can be more sample efficient than model-free algorithms like Q-learning but require additional computational resources to learn and maintain the model. Referring to our previous work (Wang *et al.* 2022*a*), the SAC algorithm is a feasible choice selected in the following study.

The SAC method (Haarnoja *et al.* 2018) is an actor-critic off-policy DRL algorithm that learns by leveraging a maximum entropy reinforcement learning algorithm. The agent aims to maximize entropy and prospective reward and reach the desired value while acting as randomly as possible. Since it is an off-policy algorithm, training can be performed efficiently with limited samples. The optimal policy can be formulated as

$$\pi^* = \arg\max_{\pi(\theta)} \mathbb{E}_{(s_t,a_t)\sim\rho_\pi} \left[\sum_t \underbrace{R\left(s_t, a_t\right)}_{Reward} + \alpha \underbrace{H\left(\pi_\theta\left(\cdot \mid s_t\right)\right)}_{Entropy} \right], \tag{2.1}$$

$$H\left(\pi_\theta\left(\cdot \mid s_t\right)\right) = -\sum_{i=1}^{n} p\left(\pi_\theta\left(\cdot \mid s_t\right)\right) \log p\left(\pi_\theta\left(\cdot \mid s_t\right)\right), \tag{2.2}$$

where $\mathbb{E}_{(s_t,a_t)\sim\rho_\pi}$ represents the expected sum of the objective function, where (s_t, a_t) are sampled according to ρ_π. This distribution, ρ_π, captures the state and state-action marginal distributions of the trajectory induced by policy π. Additionally, p denotes the unknown state transition probability, defining the likelihood of transitioning from one state to another given a specific action within a Markov decision process, $R(s_t, a_t)$ represents the reward for taking action a_t in state s_t, $H(\pi_\theta(\cdot|s_t))$ is the entropy of the policy π_θ at state s_t and α is the temperature coefficient. The smaller α is, the more uniform the distribution of the output action becomes, and the maximum entropy reinforcement learning degrades to standard reinforcement learning (i.e. $\alpha \to 0$). The objective is to find the optimal policy π_θ that maximizes the expected reward while maximizing entropy.

The core idea behind the maximum entropy approach is to randomize the policy by distributing the probability of each action output widely rather than concentrating on a single action. This approach enables the neural network to explore all possible optimal paths and avoid losing the essence of maximum entropy to a single action or trajectory. The resulting benefits include: (i) learning policies that can serve as initializations for more complex tasks, as the policy learns multiple ways to solve a given task, making it more conducive to learning new tasks; (ii) strengthening the ability to explore makes identifying better patterns easier under multimodal reward conditions; and (iii) enhancing the robustness and generalization ability of the approach since the optimal possibilities are explored in different ways, making it easier to adjust in the presence of interference.

The entropy term in the SAC algorithm affects the policy's exploration in two important ways. First, the entropy term encourages the policy to take more exploratory actions by adding a penalty to the objective function for actions with low probability under the current policy. This penalty is proportional to the negative entropy of the policy, which measures the degree of randomness or uncertainty in the actions selected by the policy. Minimizing this penalty incentivizes the policy to explore more widely and try out new actions that may lead to higher rewards. Second, the entropy term also helps prevent the policy from becoming too deterministic, which can limit its ability to adapt to changes in the environment or learn new behaviours. By adding an entropy term to the objective function, SAC encourages the policy to maintain a balance between exploration and exploitation rather than becoming overly focused on a single optimal action. This can be particularly important in environments with multiple suboptimal solutions or where the optimal solution may change over time. In summary, the entropy term in SAC encourages exploration by penalizing the policy for taking low-probability actions. It helps prevent the policy from becoming too deterministic by promoting a balance between exploration and exploitation. This can lead to better performance and more robust learning in complex, dynamic environments.

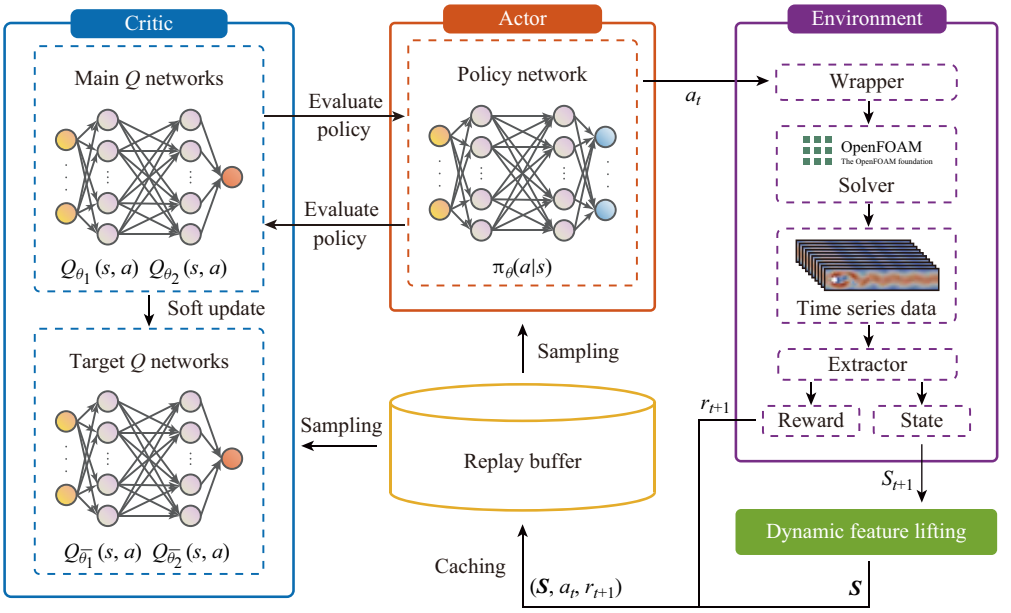


Figure 1. Schematic of the DF-DRL (SAC) framework used in the present study. The term wrapper refers to the process of encapsulating actions from the agent and sending them to the OpenFOAM solver. In contrast, extractor refers to the process of parsing CFD results and providing feedback to the agent. This framework is derived from the DRLinFluids package (Wang *et al.* 2022*a*).

The SAC algorithm used in this study employs a maximum entropy target as its optimization objective, which has been shown to enhance the algorithm's exploration properties and robustness. The ability to effectively explore is achieved by maximizing information entropy, which promotes a uniform distribution of the probability of each action output rather than focusing on a single action. For instance, the uniform strategy is a high-entropy strategy. On the other hand, the robustness of the algorithm is reflected in its ability to generate alternative action outputs when faced with environmental noise. In contrast, a previous greedy strategy may lead to the agent's inefficacy due to the certainty of its actions. The SAC algorithm ensures that every action has a varying probability rather than being either high or low. Therefore, the agent can still produce alternative action outputs without failure when the environment encounters noise. From these perspectives, the SAC algorithm is highly suitable for the present study, which involves the application of AFC using surface pressure temporal series. For a more detailed description of the SAC algorithm, please refer to Haarnoja *et al.* (2018).

The present work employs a closed-loop control framework for the AFC task described in § 3.2, as depicted in figure 1. The framework comprises two main components: the environment and the DRL agent (critic and actor in the case of the SAC algorithm used). The flow around a circular cylinder is simulated by OpenFOAM, as described in § 3.2. The flow velocity or pressure measured by specific sensors is collected as the state provided to the agent. Following the set-up of sensors in the Rabault *et al.* (2019) study, 147 sensors can capture sufficient flow information for control policy learning, which is adopted as a baseline. The DRLinFluids package (Wang *et al.* 2022*a*) is used to accomplish the interaction between the DRL agent and the CFD simulation, and Tianshou (Weng *et al.* 2022) is employed as the DRL algorithm backend. The DRL policy network consists of two dense layers, each with 512 fully connected neurons. The input layer receives data from

pressure sensors, and the output layer gives the jet velocity. The time interval between each step is set to 7.5 % of the vortex-shedding period of the cylinder without actuators. The SAC agent interacts with and updates the ANN parameters every 50 steps. The process is repeated three times with the same hyperparameters to ensure the stability and validity of the training. To save training time and provide a consistent start point, a vanilla case without control is simulated in advance until it reaches stable status. Then, the state of the flow field is stored and utilized as the initialization for the following DRL training stage.

To avoid non-physical abrupt changes in pressure and velocity resulting from the use of incompressible CFD algorithms, a continuous-time approach is adopted for the control mechanism. The control for each jet is determined at every time step during the simulation and smoothed to obtain a continuous control signal over time. An appropriate interpolation method is crucial to serialize this system's received time-discretized control signal. Hence, we smooth the jet actuation to ensure a continuous change in the control signal without excessive lift fluctuations due to sudden changes in jet velocity. Based on the interpolation functions demonstrated by Tang *et al.* (2020), the control action is set to change as follows:

$$V_{\Gamma_i(t)} = V_{\Gamma_i(t-1)} + \alpha[a - V_{\Gamma_i(t-1)}], \quad i = 1, 2, \tag{2.3}$$

where $\alpha = 0.1$ is a numerical parameter determined by trial and error, $V_{\Gamma_i(t)}$ and $V_{\Gamma_i(t-1)}$ are the jet flow velocities used at the non-dimensional times t and $t-1$, respectively, and a is one jet flow velocity in an agent step, i.e. the action generated by the DRL agent.

Flow control of circular cylinders is a highly popular topic in both the academic and industrial sectors. This research aims to reduce or eliminate drag and lift forces using advanced reinforcement learning techniques. This objective can be achieved by setting an appropriate reward function. A reward function that combines drag and lift coefficients is proposed to achieve the optimization goal, which is designated as follows:

$$r_t = (C_D)_{Baseline} - \langle C_D^t \rangle_T - 0.1 * |\langle C_L^t \rangle_T|, \tag{2.4}$$

where $(C_D)_{Baseline}$ is the mean drag coefficient of the circular cylinder without flow control, C_D^t and C_L^t are the temporal drag and lift coefficients at the time t, respectively, and $\langle \cdot \rangle_T$ indicates the sliding average back in time over a duration corresponding to one jet flow control period T with AFC. It consists of three parts: (i) the mean drag coefficient of the non-controlled flow around a cylinder, serving as a baseline for the reward function, which helps to shift the overall mathematical expectation of the function closer to zero. This prevents the probability of sampling a potentially optimal action from decreasing as the gradient ascent progresses in the action space (Weaver & Tao 2013); (ii) the moving average drag coefficient with two DRL-based actuators that is the main component of the overall reward function. It aims to indicate to the DRL agent that its primary objective is to reduce drag force (coefficient); and (iii) the absolute value of a scaled moving average lift coefficient as a penalty term. In the absence of this penalization, the policy network within the SAC algorithm has the potential to manipulate the flow pattern in order to achieve a greater reduction in drag, reaching up to approximately 18 % drag reduction (Rabault & Kuhnle 2019) in some cases. However, this comes at a significant trade-off as it also results in a substantial increase in induced lift, which is detrimental in the majority of practical applications.

2.2. *Active flow control with dynamic feature-based DRL enhancement*

Computational fluid dynamic numerical simulations allow the collection of space–time-resolved data within the considered computational domain. However, in the real world, obtaining a comprehensive view of the flow field is often difficult, which means only

a limited number of time-resolved sensor measurements $\boldsymbol{s}$ are accessible. This study aims to demonstrate how recent advancements in system identification and machine learning can be utilized to construct reduced-order models directly from these sparse sensor measurements. To achieve this, we simulate experimental conditions using direct numerical simulations and focus on a single-sensor measurement represented by

$$s\left(t\right) := p\left(t; \mathcal{L}\right), \tag{2.5}$$

where p represents the surface pressure, and $\mathcal{L}$ denotes the various sensor layout schemes, which are detailed in § 3.3. The measurement vector s can generally comprise various measurements such as the lift and drag coefficients, pressure measurements on a cylinder or velocity field measurements at specific locations, e.g. wake region. However, for the scope of this study, the pressure alone is deemed adequate to characterize the flow according to the results shown in § 4.1.

Given the sensor measurements $\boldsymbol{s}$, our objective is to develop a practical flow state estimation that enables a DRL agent to obtain efficient information based on it. However, raw signals may not be ideal for this purpose, and an augmentation or DF lifting is required to incorporate sensor measurement functions. In this regard, we define the augmented state $\boldsymbol{S}$ as a feature vector that encompasses such parts

$$\boldsymbol{S} = \boldsymbol{g}(\boldsymbol{s}). \tag{2.6}$$

Numerous options exist for the mapping function $\boldsymbol{g}$, which can enhance sensor measurements and improve model accuracy. If the sensors are adequate to determine the system state, the identity map can be utilized as $\boldsymbol{g}$, which means $\boldsymbol{S} = \boldsymbol{s}$. Alternatively, $\boldsymbol{g}$ can leverage proper orthogonal decomposition mode coefficients when the measurements provide high-dimensional snapshots. Takens (1981) and Brunton *et al.* (2017) use delay embedding technology to augment the measurements, resulting in a sufficiently high-dimensional feature vector that fully characterizes the system dynamics. Selecting an effective transformation function $\boldsymbol{g}$ is a critical unresolved issue relevant to both representation theory and the Koopman operator viewpoint on dynamical systems. Both Mezić (2005) and Brunton *et al.* (2016*a*) are actively investigating this problem. In this study, we choose g to augment the sensor measurement with its time derivative while appropriately scaling the augmented measurement. Furthermore, Loiseau, Noack & Brunton (2018) propose a comprehensive sparse reduced-order modelling for flow full-state estimation, which includes time-resolved sensor data and optional non-time-resolved particle image velocimetry snapshots. Inspired by the facts mentioned above, we present a novel approach, named DF-based DRL, to overcome the limitations of measurements in the real world and highlight the potential of DRL techniques for sparse surface pressure sensing. An effective augmentation function $\boldsymbol{g}$ at time t is used to lift the sensor signals so that a high-dimensional DF space is formed, which can be expressed as

$$\boldsymbol{S}_t = \begin{pmatrix} \alpha s_{t-M}^1 & \cdots & \alpha s_{t-M}^i & \beta a_{t-M}^1 & \cdots & \beta a_{t-M}^j \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \alpha s_t^1 & \cdots & \alpha s_t^i & \beta a_t^1 & \cdots & \beta a_t^j \end{pmatrix} \in \mathbb{R}^{(M+1)\times(i+j)}, \tag{2.7}$$

where $\boldsymbol{a}$ is the agent action at time t, M is the number of backtracking time steps t, which is set to 30 in this study, corresponding to twice the number of time steps of the baseline vortex shedding period, i and j are the identifiers of sensor and actuator, respectively, and α and β are the corresponding scaling factors. The final algorithm is listed in Algorithm 1. In the context of DF-DRL with SAC, the approach involves a cyclic iteration of gathering

Algorithm 1: The DF-DRL algorithm with SAC

Input: sensor state vector $s \in \mathbb{R}^{1\times i}$, initial critic network parameters θ_i, initial actor network parameters ϕ, state transition probability p (deterministic CFD simulation in this study)
Initialization: target network weights $\overline{\theta_i} \leftarrow \theta_i$, replay buffer $\mathcal{D} \leftarrow \emptyset$
for *each iteration* **do**
  for *each environment step* **do**
    $a_t \sim \pi_\phi(a_t \mid s_t)$;
    $s_{t+1} \sim p(s_{t+1} \mid s_t, a_t)$;
    $\boldsymbol{S}_t \leftarrow \boldsymbol{g}(\{s_1, s_2, \ldots, s_{t+1}\} \cup \{a_1, a_2, \ldots, a_t\})$;
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{(\boldsymbol{S}_t, a_t, r(s_t, a_t))\}$;
  end
  for *each gradient step* **do**
    Update θ_i, ϕ, $\overline{\theta_i}$, and temperature T according to (Haarnoja *et al.* 2018);
  end
end
Output: actuator control vector $a \in \mathbb{R}^{1\times j}$

experience from the environment based on the current policy and updating the function approximators by utilizing stochastic gradients from batches sampled from a replay pool. The original state vector within a state-action tuple will first be lifted according to the augmentation function g, then dumped to the replay buffer $\mathcal{D}$.

In past studies, a common practice has been to use a single snapshot of the flow field as state data, such as four sensors worth of pressure data in one time step, to provide input to the policy network. This is illustrated by the upper panel of figure 2. By contrast, the DF-DRL method uses pressure data assembled from sensor measurements extracted from the 30 previous action time steps t, resulting in an augmented agent state. The details of DF lifting within the DF-DRL method are illustrated in lower figure 2. More specific DF-DRL hyperparameters have been listed in table 4 of the Appendix. It is expected (and confirmed in § 4.1) that the policy will be improved using such DF lifting input data. However, a possible challenge is that this may increase the dimensionality of the state input to the ANN quite a bit since it is a two-dimensional array with one dimension corresponding to the sensor number and another corresponding to the time series index. In particular, sensors on the surface observe lower magnitude variations in flow velocity and pressure than sensors in the wake. They cannot observe changes in the trend of the wake and the shedding of cylinder vortices. Therefore, using the DF-DRL method is most appropriate for the surface sensors' AFC training process, which involves fewer sensors. Besides, it is also vital to use an input standardization method individually on each sensing time series. In particular, it is necessary to normalize the surface pressure sensor observations so that these fluctuations are well perceived even though these have a very different dynamic range compared with sensors in the wake region.

3. Numerical plant: flow around a circular cylinder

In this section, we choose benchmarks proposed by Schäfer *et al.* (1996), laminar flow around a cylinder and place jets symmetrically arranged on both lateral sides as actuators for active flow control. The objective is to reduce the cylinder's drag force and

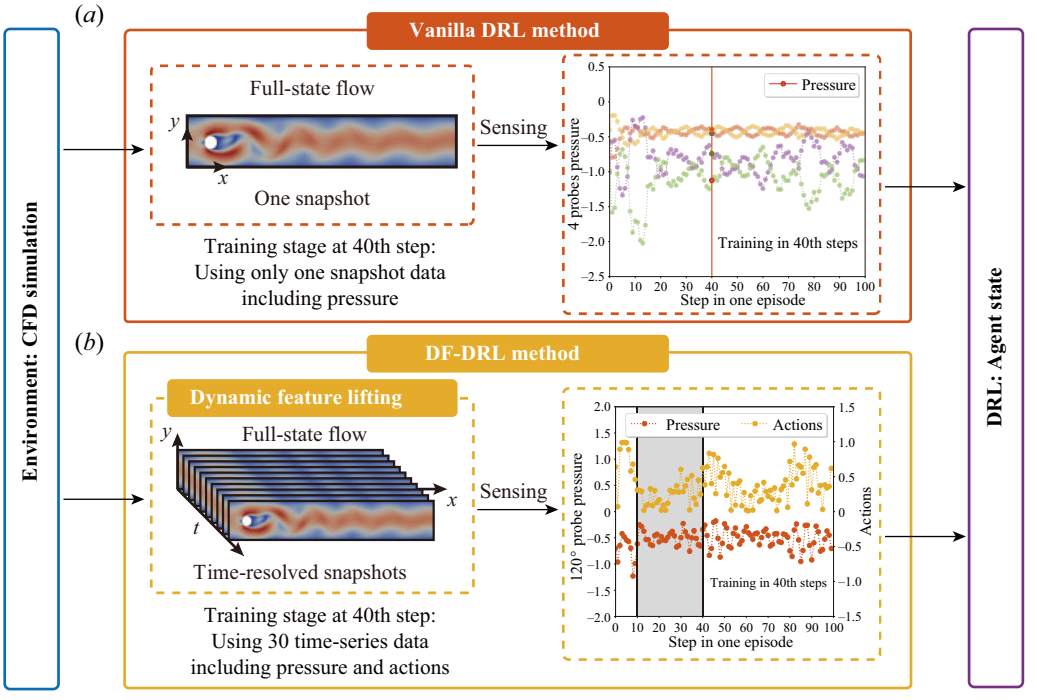


Figure 2. Flowchart of the two approaches for the state collected from the environment. (*a*) Vanilla method (sensor feedback): the agent only collects the flow field state at a single time step. For example, the signal obtained from four pressure sensors located in the flow field at the 40th time step returns a state vector $s \in \mathbb{R}^{4\times1}$; (*b*) DF-DRL method: the agent collects data from the most recent thirty time steps t, including historical sensor pressure $p \in \mathbb{R}^{30\times1}$ and action data $a \in \mathbb{R}^{30\times1}$ provided by the agent. This process indicates DF lifting and the dimension of the state vector $\boldsymbol{S} \in \mathbb{R}^{30\times2}$. Moreover, scaling the state vector will amplify signal fluctuations, which is helpful in capturing the flow characteristics.

lift fluctuation. Firstly, we formalize the research problem in § 3.1. Then, we provide a detailed description of the flow configuration and numerical solution methods in § 3.2, followed by a validation of the accuracy of the numerical algorithms. Finally, we define three types of sensor layout schemes in § 3.3.

3.1. *Problem formalization*

The AFC task formulated in this study aims to find a real-time control policy π of two jet actuators located on a circular cylinder with sensor feedback, which can effectively reduce the fluid force on it. Generally, the surface pressure information s_t can be regarded as the input state of the control policy π, and the jet intensity can be viewed as the DRL action a_t at the time t. The action is decided by the DRL controller based on the state observation. Therefore, the control processing can be modelled as a deterministic or stochastic relationship

$$a_t \sim \pi(s_t|\theta). \tag{3.1}$$

Hence, given a DRL agent with the control policy π, the objective is to minimize the lift and drag coefficients of the cylinder by optimizing the set of weights θ of the DRL agent

policy network

$$\pi^* = \pi\left(\theta^*\right), \tag{3.2}$$

$$\theta^* = \arg\max_{\pi(\theta)} \mathbb{E}_{(s_t,a_t)\sim\rho_{\pi(\theta)}} \mathcal{T}(s_t, a_t), \tag{3.3}$$

where the superscript * represents the optimal value, $\mathbb{E}$ is the expected value operator and $\mathcal{T}$ denotes a target function, which represents the current policy π.

3.2. *Flow configuration and numerical method*

In this work, we use the open-source CFD package OpenFOAM to perform simulations. Under the assumption of incompressible viscous flow, the governing Navier–Stokes equations can be expressed in a non-dimensional manner as

$$\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u}\cdot(\boldsymbol{\nabla}\boldsymbol{u}) = -\boldsymbol{\nabla}p + Re^{-1}\Delta\boldsymbol{u}, \tag{3.4}$$

$$\boldsymbol{\nabla}\cdot\boldsymbol{u} = 0, \tag{3.5}$$

$$Re = \frac{\bar{U}D}{\nu}, \tag{3.6}$$

where $\boldsymbol{u}$ is the non-dimensional velocity, t is the non-dimensional time, p is the non-dimensional pressure, ν is the kinematic viscosity of the fluid and $\bar{U}$ is the mean velocity at the inlet. The corresponding Reynolds number Re is 100 in the training stage.

This study focuses on 2-D simulations of flow around a circular cylinder with a diameter D, which is the characteristic length scale. The computational domain has dimensions of $L = 22D$ and $B = 4.1D$ in the streamwise and cross-stream directions, respectively, as shown in figure 3. Following the widely recognized benchmark conducted by Schäfer *et al.* (1996), the cylinder is slightly off centre to induce faster development of the vortex-shedding alley during the initial simulation convergence stage. The outlet boundary is placed $19.5D$ downstream of the cylinder to allow the wake to develop fully.

The inlet boundary, denoted as Γ_{in}, is subject to the parabolic velocity inlet boundary condition. The no-slip constraint, Γ_W, is applied to both the top and bottom of the channel and the surfaces of the cylinder. Additionally, the right boundary of the channel, Γ_{out}, is designated as a pressure outlet, wherein zero velocity gradient and constant pressure are maintained. The inlet boundary is assigned as a parabolic velocity form and expressed as the following in the streamwise direction:

$$U(0, y) = 4U_m y(H - y)/H^2, \tag{3.7}$$

where U_m is the maximum inflow velocity at the middle of the channel. Employing a parabolic inflow profile, U_m is 1.5 times the mean velocity $\bar{U}$, as defined by

$$\bar{U} = \frac{1}{H}\int_0^H U(y)\,\mathrm{d}y = \frac{2}{3}U_m. \tag{3.8}$$

To accomplish the AFC, a flow control technique using two jet actuators (Γ_1 and Γ_2) located on opposite sides of the cylinder is employed. A parabolic velocity profile with a jet width of $\omega = 10^\circ$ is imposed at both jets, as depicted in figure 3. Due to the velocity of the jet flow being orthogonal to the inflow direction, drag reduction is strictly achieved by effective actuation rather than by momentum injection. Moreover, the jet flow on both

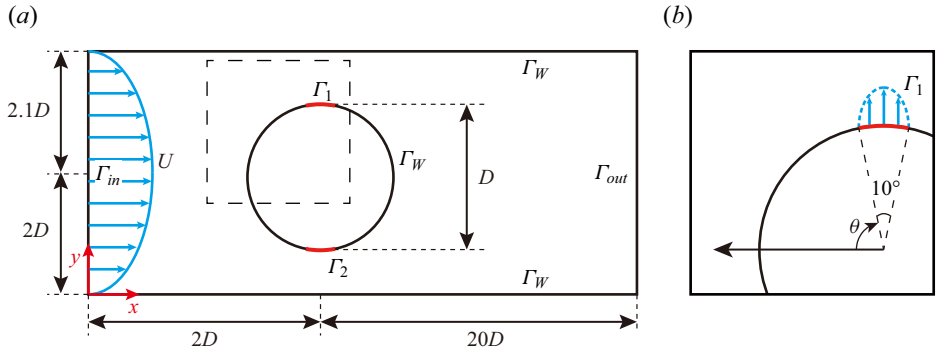


Figure 3. Description of (*a*) the numerical set-up, which is adapted from Schäfer *et al.* (1996). The origin of the coordinates is located at the lower left corner of the entire computational domain. Here, Γ_{in} stands for the inflow velocity with a parabolic flow profile, while Γ_{out} is for the outflow. A no-slip wall boundary constraint Γ_W is applied on the bottom and top of the channel and on the cylinder surfaces. Two jet holes (Γ_1 and Γ_2) are present at both sides of the cylinder. (*b*) An enlarged view of the dashed box in panel (*a*). The 0° azimuthal angle corresponds to the foremost point on the cylinder's windward surface and increases clockwise. The jet actuator opening angle is 10°, consistent with Rabault *et al.* (2019).

sides is constrained as synthetic jet flow, i.e. $V_{\Gamma_1} = V_{\Gamma_2}$, so the jets collectively do not add to or remove mass from the flow. The normalized jet flow rate Q_i^* of a jet is defined as

$$Q_i^* = \frac{Q_{jet,i}}{Q_{ref}} = \frac{U_{jet,i} \cdot D_{jet}}{\bar{U}D}, \tag{3.9}$$

where $U_{jet,i}$ is the i_{th} jet velocity, D_{jet} is the width of the jet and Q_{ref} is the reference mass flow rate intercepting the cylinder; Q_i^* is not greater than 0.2 during this study.

The current study adopts unstructured meshes for CFD simulations. Emphasis has been laid on refining the mesh around the surface boundary and the wake flow regions, as these are crucial for ensuring the appropriate resolution of these significant flow domains and the physics happening there. The numerical solution is obtained at each time step, and the drag (F_D) and lift (F_L) forces are computed by integrating over the cylinder surface, following:

$$\boldsymbol{F} = \int (\boldsymbol{\sigma} \cdot \boldsymbol{n}) \cdot \boldsymbol{e}_j \, \mathrm{d}, \tag{3.10}$$

where σ is the Cauchy stress tensor, and the unit vector n is defined as normal to the cylinder surface. At the same time, e_j is denoted as a unit vector in the direction of the inflow velocity for drag force calculations and as a vector perpendicular to the inflow velocity for lift force calculations. Specifically, the drag C_D and lift C_L coefficients can be expressed as follows:

$$C_D = \frac{F_D}{\frac{1}{2}\rho \bar{U}^2 D}, \tag{3.11}$$

$$C_L = \frac{F_L}{\frac{1}{2}\rho \bar{U}^2 D}, \tag{3.12}$$

where F_D and F_L are denoted as integral drag and lift force, respectively.

To further validate the accuracy of the CFD simulations, a series of mesh convergence studies is performed at a Reynolds number $Re = 100$. In particular, meshes of three

Case	Grid number	C_D^{max}	C_L^{max}	$\overline{C_D}$	St
Grid I	13 572	3.27	1.01	3.21	0.301
Grid II	16 200	3.24	1.024	3.205	0.299
Grid III	39 382	3.24	1.04	3.21	0.302
Schäfer *et al.* (1996)	—	3.22–3.24	0.99–1.01	—	0.295–0.305
Rabault *et al.* (2019)	9262	3.245	1.020	—	0.302

Table 1. Numerical results of mesh convergence study for the 2-D flow around a circular cylinder at $Re = 100$.

different resolutions are employed. The corresponding results for the maximum values of the drag coefficient (C_D) and lift coefficient (C_L), denoted as C_D^{max} and C_L^{max}, respectively, are reported in table 1. The numerical analysis reveals that the discrepancies among various mesh resolutions are insignificant. Considering the trade-off between computational cost and numerical accuracy, the meshing scheme of grid II is preferred for the DRL training stage. More specific flow configurations have been listed in table 3 of the Appendix.

3.3. *Layout of surface pressure sensors*

The present study proposes a series of pressure sensor layout schemes to study the influence of sensor location. First, a baseline configuration proposed by Rabault *et al.* (2019) with 147 pressure sensors is set up both around the cylinder and in the wake region, as shown in figure 4(*a*). Then, a varying number of pressure sensors, e.g. 4, 8 or 24 sensors, are symmetrically arranged on the surface of the cylinder (along the direction of inflow). To avoid inadequate information with more minor pressure fluctuations at the front of the cylinder, the sensors are uniformly distributed except for the point at the front, as shown in figure 4(*b*). Finally, a comprehensive study is carried out using a single sensor location. The placement of the single sensor is started by putting it at the front of the cylinder as $\theta = 0^\circ$ relative to the incoming flow. We change its position by gradually increasing its angular position on the cylinder in increments of 15° until it reaches the rear edge of the cylinder ($\theta = 180^\circ$), as shown in figures 3(*c*) and 4(*c*). As a consequence, a total of 17 single pressure sensor positions are investigated. One could expect that pressure sensors on the surface of the cylinder can provide valuable information about the flow to the DRL controller. However, using surface sensors like those shown in figures 4(*b*) and 4(*c*) presents a challenge due to the limited quantity of data provided and the placement being solely on the surface of the cylinder. This results in insufficient information regarding the cylinder wake and vortex-shedding pattern during the DRL training stage.

To facilitate the description in the next sections, the notation $\mathcal{L}$ is used to describe the different sensor layout schemes. The subscripts represent different sensor layout types. For example, $\mathcal{L}_I$ represents the baseline configuration with 147 sensors placed inside the flow field, $\mathcal{L}_{II}$ represents the sensor configuration placed on the cylinder surface and $\mathcal{L}_{III}$ represents a single-sensor configuration placed on the cylinder surface. For type $\mathcal{L}_{II}^N$, the superscript N indicates the number of sensors, and the polar coordinates of sensor i can be expressed as

$$r_i = \frac{1}{2}D, \quad \theta_i = \frac{2\pi i}{N+1}, \quad i = 1, 2, \ldots, N, \tag{3.13}$$

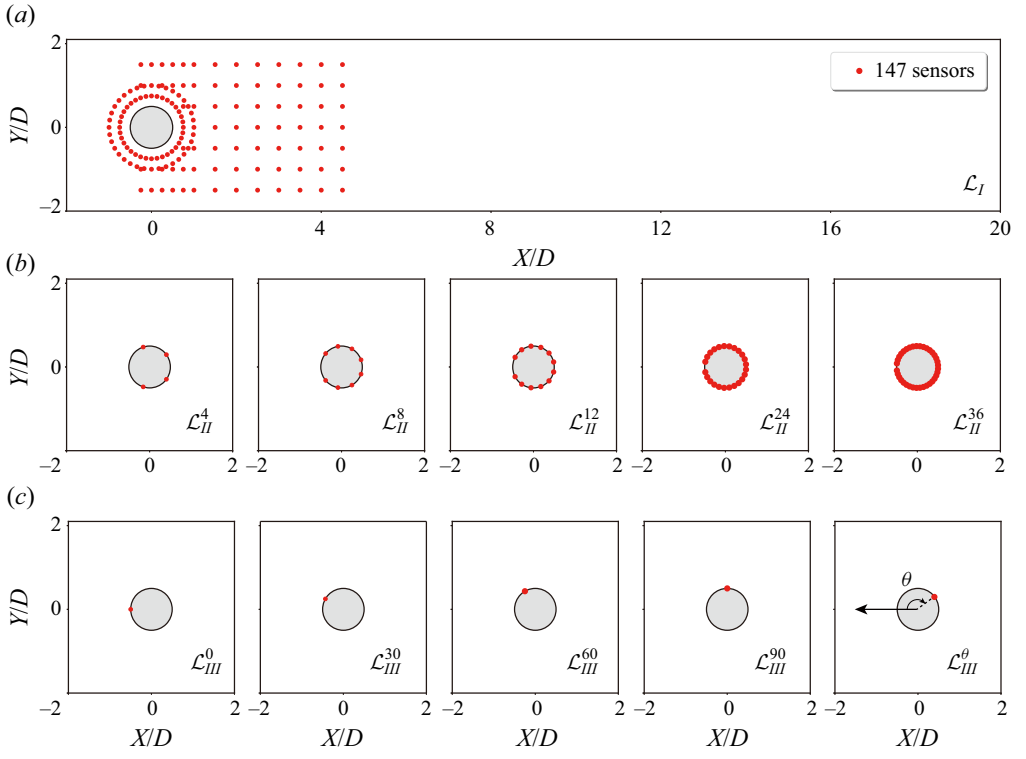


Figure 4. Number and configuration of the sensors used to generate the DRL controller state observation. (*a*) Using 147 sensors provides sufficient flow information for DRL training, stated as $\mathcal{L}_I$. (*b*) Layout using 4, 8, 12, 24 or 36 sensors in the surface of the cylinder respectively, denoted by $\mathcal{L}_{II}^4$, $\mathcal{L}_{II}^8$, $\mathcal{L}_{II}^{12}$, $\mathcal{L}_{II}^{24}$ and $\mathcal{L}_{II}^{36}$. (*c*) Layout using only one sensor located on the surface of the cylinder with an azimuthal angle of 0°, 30°, 60°, 90° or $\theta°$, signified by $\mathcal{L}_{III}^0$, $\mathcal{L}_{III}^{30}$, $\mathcal{L}_{III}^{60}$, $\mathcal{L}_{III}^{90}$ and $\mathcal{L}_{III}^{\theta}$.

while for type $\mathcal{L}_{III}^{\theta}$, the superscript θ indicates the degree of the sensor placement where the coordinate axis of the polar coordinate system has an origin opposite to the inflow direction, i.e. the leading edge of the cylinder is denoted as the 0° point, which is illustrated in figure 4.

4. Results and discussion

In this section, we first evaluate the performance and reliability of the proposed DF-DRL approach in § 4.1, comparing it with a vanilla DRL algorithm. Then, the impact of different numbers of surface sensor configurations and layouts of single surface sensors on the performance of flow control is investigated in § 4.2. Furthermore, we verify the robustness of the DF-DRL controllers under two different Reynolds numbers, $Re = 500$ and $Re = 1000$ in § 4.3.

4.1. *The DRL-based AFC with sparse surface pressure sensing*

To evaluate the effectiveness and reliability of the DF-DRL approach, sensor locations $\mathcal{L}_I$ and $\mathcal{L}_{II}^4$ are selected for illustration, as depicted in figures 4(*a*) and 4(*b*), respectively. Figure 5 shows the learning curves for active flow control using both vanilla DRL and DF-DRL techniques under two different sensor quantity configurations (4 and

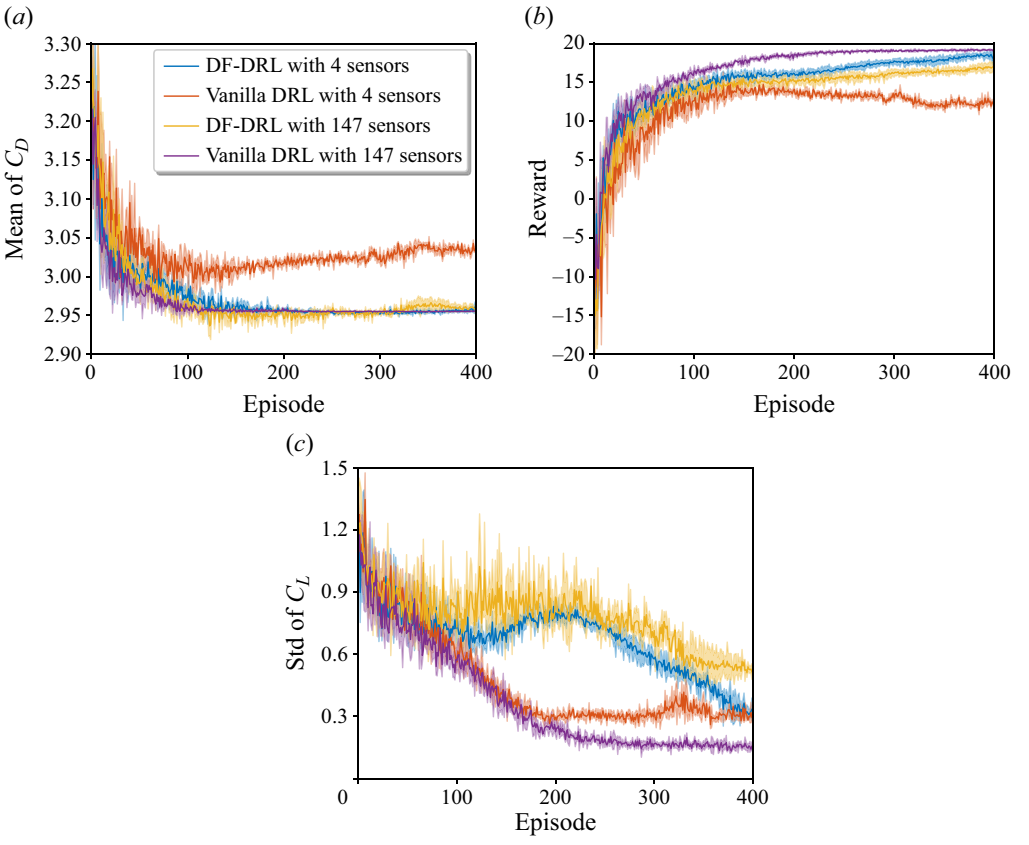


Figure 5. Comparison of (*a*) the mean C_D, (*b*) reward and (*c*) the std of C_L when using different DRL methods, i.e. using the DF-DRL method or not, and a different number of sensors (4 sensors in the surface of cylinder and 147 sensors around the cylinder). The learning case condition, which contains 147 sensors without time series, reaches the maximum reward. These cases are trained three times repeatedly, and we present the average between the three runs (thick line) and the std between these runs (shadowed area).

147 sensors). The three panels (*a*), (*b*) and (*c*) correspond to the mean C_D, reward, and the standard deviation (std) of C_L, respectively. It seems that the C_L learning curve, as shown in figure 5(*c*), has not reached a converged state, especially for the two cases that utilize the DF lifting technique. However, we maintain that DRL training stabilizes as evidenced by the clear convergence observed in the C_D learning curve as depicted in figure 5(*a*). This is because, as explained in § 2.1, C_L is only used as a penalty term to prevent the policy network from finding control strategies that are not practical or realistic.

The results indicate that there are significant differences between these four cases. Using a vanilla DRL algorithm, the maximum drag reduction of 8 % was achieved with the use of 147 sensors (scheme $\mathcal{L}_I$), and the maximum reward value is 19.13. Moreover, the std of C_L is reduced to 0.15 when learning has converged. For the case of 4 sensors with the vanilla DRL method (scheme $\mathcal{L}_{II}^4$), the maximum drag reduction is only 6.4 %, and the std of C_L decreases to only 0.29, as shown in figure 5. These two cases show that, with few surface sensors, AFC performance worsens. This is due to the inability of DRL to estimate the flow field and limited observable data correctly.

As for the results using the DF-DRL method, it is observed that both $\mathcal{L}_I$ and $\mathcal{L}_{II}^4$ achieve similar drag-reduction amplitudes of 8 %, corresponding to the maximum drag reduction

also observed using the 147 pressure sensors. However, the latter approach achieves a higher reward value, as indicated by the decrease in the std of C_L due to the lift penalty term within the reward function. This performance improvement is happening even though both the $\mathcal{L}_I$ and $\mathcal{L}_{II}^4$ approaches undergo a temporary rebound in their learning. Therefore, when utilizing the DF-DRL method, fewer sensors, as used in scheme $\mathcal{L}_{II}^4$, can achieve the same drag reduction as scheme $\mathcal{L}_I$, which achieves 25 % better than the vanilla model based on direct sensor feedback. This can be obtained while also improving the reduction of lift fluctuations. These observations suggest that the DF-DRL method can maintain drag-reduction performance while reducing the number of sensors required. The difference in reward observed between the two cases can be attributed to the reduction in the std of C_L achieved in scheme $\mathcal{L}_{II}^4$.

For scheme $\mathcal{L}_I$, with many sensors distributed around the cylinder and wake region, the DRL controller can obtain exhaustive flow field information. Thus, the inclusion of historical data provided by the DF-DRL method has a minor impact. However, scheme $\mathcal{L}_{II}^4$, with limited sensor numbers and sparse information on the cylinder surface, has more information available to perform effective flow control with the DF-DRL method. These results suggest, unsurprisingly, that using more sensors leads to better C_D reduction effects and more stable reward convergence with naive DRL agents. Moreover, when the flow field information is limited in quantity and placement of the sensors (due to physical restrictions), the DF-DRL method demonstrates better convergence and yields a superior control policy.

4.2. *Control performance and learning convergence with DF-DRL method*

To further investigate the impact of sensor quantity and placement azimuth on the control effectiveness and convergence performance of the DF-DRL-based controller, we conducted case studies with different quantities of sensors, i.e. 1, 4, 8, 12, 24 and 36, as well as various placement azimuth layouts of 0° to 180° with a 15° spacing.

4.2.1. *Sensor quantity*

Five typical layout schemes $\mathcal{L}_{II}$ of surface pressure sensors are investigated in this section. The arrangement of these sensors is depicted in figure 4(*b*), where the sensors are evenly distributed, and all the leading edge sensors are removed, as described in 3.3. Since the state includes the jet actions component, the pressure sensors distributed around the jet will not significantly impact the results.

The impact of adding more surface pressure sensors on training performance is illustrated in figure 6. Results show that increasing the number of sensors does not lead to a significant improvement in drag and C_L reduction, which remains around 8 % across all schemes. Additionally, all cases converge at approximately 200 episodes. Figure 6(*b*) displays learning curves that follow the same trend as the C_D, indicating that the final DF-DRL performance is very similar to the benchmark case for different numbers of pressure sensors on the surface of the cylinder.

As depicted in figure 6(*c*), the standard deviation of the lift exhibits a comparable declining pattern to that observed in the previous section. Following an initial decrease and temporary increase at around episode 200, all five groups undergo a consistent decrease until the completion of DF-DRL training.

This result can be explained by the fact that the pressure on the cylinder surface, as a surrogate for lift, is a better DF than wake measurements, where varying vortex shedding destroys the phase relationship. The pressure on the cylinder surface provides

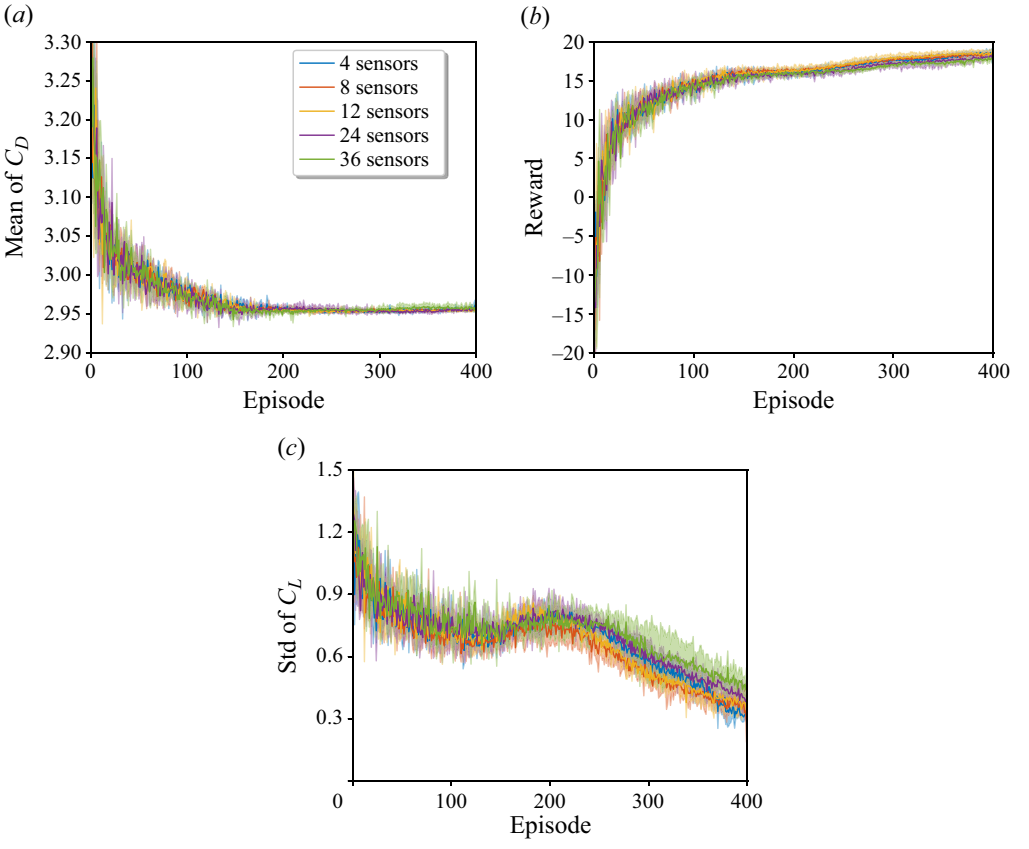


Figure 6. Evolution of (*a*) mean C_D reduction, (*b*) reward and (*c*) std lift coefficient for different numbers of sensors with DF-DRL method. Sensors are uniformly located on the surface of the cylinder as indicated in figure 4(*c*).

a more accurate representation of the dynamic behaviour of the flow, leading to a better understanding of the underlying physics. This conclusion agrees with previous work by Loiseau *et al.* (2018). Using pressure as a DF highlights the potential for feature-based approaches in reduced-order modelling of fluid flows.

Even though the final drag-reduction effect is very similar, an interesting subtle difference can be observed from figure 6, where the DRL algorithm with DF lifting does not show a monotonically increasing relationship between the number of sensors and control performance.

The counterintuitive scenario mainly arises from the potentially redundant information (features) in the state (Roghair *et al.* 2022). This can be explained by the phenomenon of the curse of dimensionality (Altman & Krzywinski 2018) commonly observed in deep learning. The curse of dimensionality refers to a series of problems and challenges encountered in high-dimensional spaces. It describes the difficulties and obstacles faced when dealing with data with a large number of dimensions.

The main issues associated with the curse of dimensionality in high-dimensional spaces include: (i) the distances between data points become very large, leading to data sparsity. This means that it becomes challenging to find other similar data points near a given data point, making data analysis and modelling difficult; (ii) selecting

and extracting meaningful features becomes complex. Due to the large number of dimensions, it is difficult to determine which features are most important for solving a specific problem; (iii) statistical inference and model estimation in high-dimensional data become challenging. Due to data sparsity and computational complexity, obtaining reliable statistical results from limited data samples becomes difficult, leading to issues like overfitting or underfitting.

In this study, the data collected by each sensor can be regarded as a feature. For a case with i sensors and two actuators, considering the commonly used DF lifting, which includes the past M time steps of pressure and the magnitude of the actuator jet velocity, the dimensionality increases from $\mathbb{R}^{1\times i}$ to $\mathbb{R}^{M\times(i+1)}$. When the dimensionality of the state increases, the complexity and variability of the data also increase. If there are too few sensors, important information and patterns may be missed, leading to suboptimal control performance. Insufficient sensors may result in incomplete data coverage or inadequate representation of the system dynamics, making it difficult to capture crucial features or relationships necessary for effective control.

On the other hand, if there are too many sensors, the curse of dimensionality comes into play. The high-dimensional space poses challenges in terms of data sparsity, computational complexity and feature selection. The abundance of sensors may introduce noise, redundancy or irrelevant information, which can hinder the control process. It becomes difficult to discern which sensors are truly informative and contribute meaningfully to the control objective.

Given that the disparities in C_D and C_L fluctuation reduction among the experiments are not substantial, to further demonstrate the potential of using surface pressure as a DF for a nonlinear system and its combination with DRL, we will reduce the number of sensors to one in the next section.

4.2.2. *Placement azimuth*

Based on the results above, it is apparent that an increase in the number of sensors utilized in DF-DRL-based AFC tasks does not necessarily result in better performance, including drag and lift reduction. To explore the maximum performance potential of DF-DRL and obtain the optimal sensor layout scheme, single-sensor schemes $\mathcal{L}_{III}$ are selected in the following study, as illustrated in figure 4(c). According to the symmetry of the geometry of the set-up and the boundary conditions, this study only considers deploying sensors on the upper semicircle region for training and analysis purposes. The cylinder surface features a single pressure sensor located every 15°, covering a range of 0° to 180°, with a total of 13 configurations. Three training repetitions are performed for each case with the same hyperparameters to eliminate randomness. As shown in figure 7(a), the mean C_D indicates that AFC performance with only one pressure sensor is almost as optimal as baseline scheme $\mathcal{L}_I$. The results suggest that scheme $\mathcal{L}_{III}$ on cylindrical surfaces using the DF-DRL method can achieve the best AFC performance.

It can also be observed from figure 7(b) that the trailing edge sensor of the cylinder has a higher reward value than the leading edge sensor, resulting in a lower mean C_D. Furthermore, a sudden reduction in C_D can be observed between $\mathcal{L}_{III}^{75}$ and $\mathcal{L}_{III}^{90}$. This can be attributed to the influence of jet actuators situated on the top side of the cylinder, where changes in jet velocity can lead to abnormal pressure fluctuations on the surface at 90° that confuse the DRL controller. Additionally, a significant jump in performance occurs between $\mathcal{L}_{III}^{90}$ and $\mathcal{L}_{III}^{150}$, characterized by a marked decrease in the mean C_D. A decreasing trend can be observed in the std of C_L from $\mathcal{L}_{III}^{0}$ to $\mathcal{L}_{III}^{180}$, as depicted in figure 7(c).

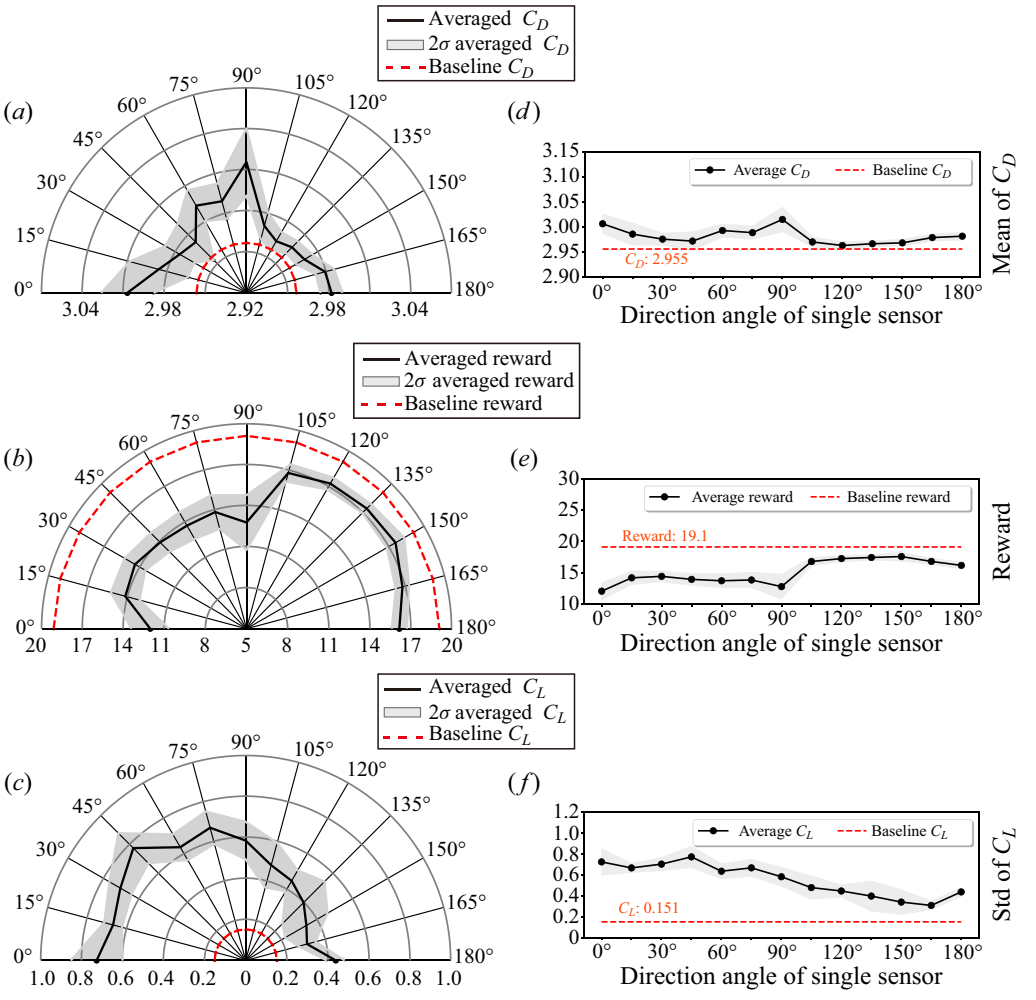


Figure 7. One pressure sensor is placed every 15° on the cylinder surface for training, from 0° to 180°, for a total of 13 sets of training. (*a–c*) Are the mean C_D, reward and the std of C_L in the last 10 training episodes, respectively, (*d–f*) are the changing trends of mean C_D, reward and the std of C_L, respectively.

This further emphasizes that a sensor located closer to the trailing edge of the cylinder can effectively generate information that can be used to mitigate the std of C_L.

To explain this phenomenon more comprehensively, figure 8 depicts the time-averaged vorticity field of the uncontrolled flow around the cylinder, with the red dots representing the 13 different single-sensor layout schemes. The sensors located at the trailing edge of the cylinder, namely $\mathcal{L}_{III}^{105}$ to $\mathcal{L}_{III}^{180}$, are positioned at the vortex-shedding location, indicating that these pressure sensors contain crucial information about vortex shedding compared with the windward side of the cylinder. This observation explains why the trailing edge pressure sensors outperform the leading edge sensors in the overall training outcomes, which include the reduction of C_D and C_L.

To summarize, a general tendency of mean C_D, reward and std of C_L is presented in figure 7(*d*). Notably, a single sensor situated between 0° and 180° demonstrates near

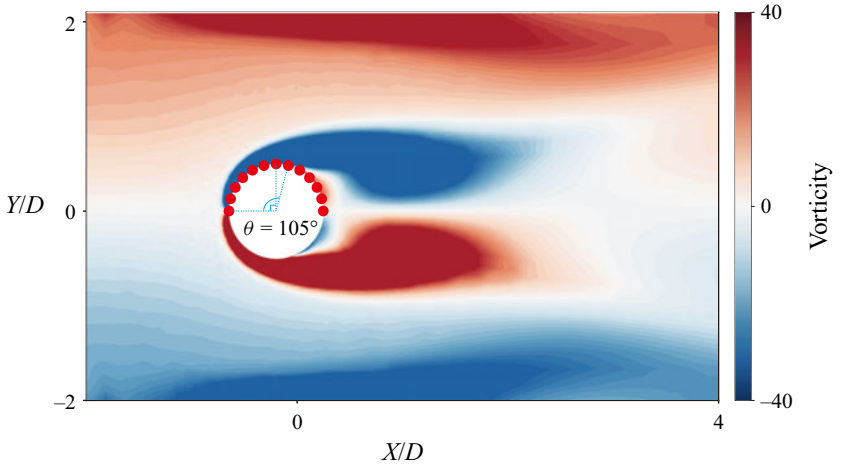


Figure 8. Contours of time-averaged vorticity of a plain case, the red points are surface pressure sensors. Vortex separation occurs between sensors 90° and 105°.

optimal reduced C_D and C_L, corresponding to an elevated reward value. These results inspire the training strategy of a single-sensor system $\mathcal{L}_{III}$ at higher Reynolds numbers.

4.3. *Robustness of DF-DRL-based plant under more complex flow scenario*

4.3.1. *Case 1: 2-D flow around a circular cylinder at higher Reynolds number*

Based on the promising performance of AFC demonstrated in § 4.2.2, scheme $\mathcal{L}_{III}^{150}$ is chosen to investigate the robustness of a single surface sensor at higher Reynolds numbers ($Re = 500$ and 1000). The policy network architecture is the same as before, consisting of two dense layers of 512 fully connected neurons, with the input layer receiving data from a single pressure sensor and the output layer representing the jet velocity. As the vortex-shedding frequency of the cylinder increases with the rise in Reynolds numbers, the SAC agent interacts with the environment every 44 and 46 time steps t at $Re = 500$ and 1000, respectively.

Figure 9 shows the evolution of the mean C_D, reward value and C_L obtained from three repeated training processes at Reynolds numbers of $Re = 500$ and 1000. After approximately 400 episodes at $Re = 500$ and 600 episodes at $Re = 1000$, C_D approached convergence, demonstrating that a stable control strategy was achieved. Meanwhile, the reward curves gradually increased with each episode, with the std of C_L declining and stabilizing. As described in (2.4), C_D and C_L are first-order terms, where lift has a weight of 1, and drag has a weight of 0.1. For the DRL agent, this implies that the reduction of drag has a higher reward. When it is reduced to its maximum value (2.1 at $Re = 500$ and 1.9 at $Re = 1000$), inhibiting lift becomes the only viable option. However, as the Reynolds number increases, the learning requires more episodes to converge, and the agents need more trial-and-error steps to comprehend the nonlinear relationships inherent in the dynamic system.

An interesting phenomenon is observed when comparing the final performance of drag reduction across different Reynolds numbers. As the Reynolds number increases, the drag-reduction effect improves. This contradicts our intuition, as we would normally expect the flow field to become more complex at higher Reynolds numbers, with increased turbulence and vortex shedding, making it difficult for the DRL agent to learn an effective

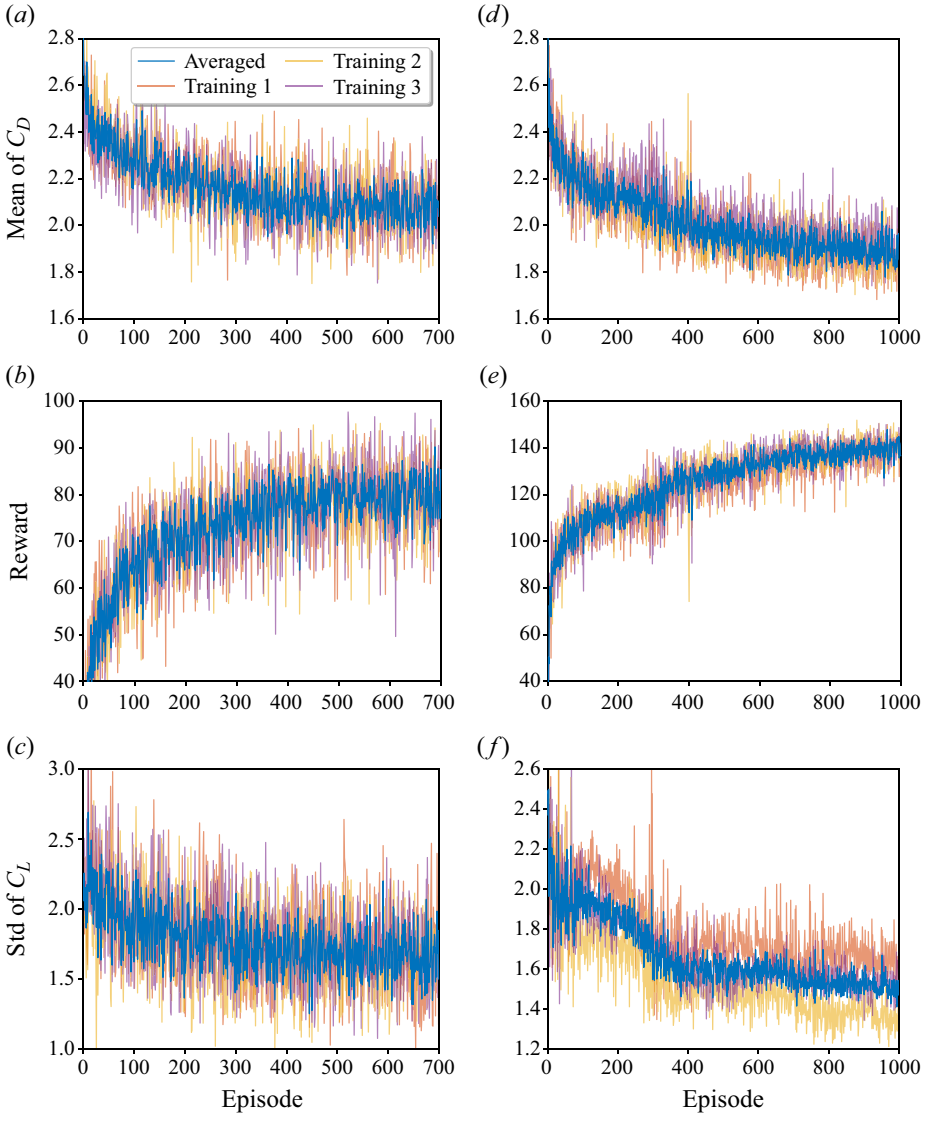


Figure 9. Evolution of the mean C_D, reward and the std of C_L during the learning process at Reynolds numbers corresponding to (*a–c*) $Re = 500$, and (*d–f*) $Re = 1000$. The averaged curves labelled in blue refer to the moving average of the mean of all three training curves in each panel. All results are obtained under scheme $\mathcal{L}_{III}^{150}$.

strategy for flow control. However, this is not the case. The main reason lies in the drag force component (Achenbach 1968). The overall drag F_d on the circular cylinder submerged in a Newtonian fluid can be calculated by

$$F_d = \underbrace{\oint p \cdot \cos(\theta) \cdot \mathrm{d}A}_{\textit{Pressure drag}} + \underbrace{\oint \tau_w \cdot \sin(\theta) \cdot \mathrm{d}A}_{\textit{Skin friction drag}}, \tag{4.1}$$

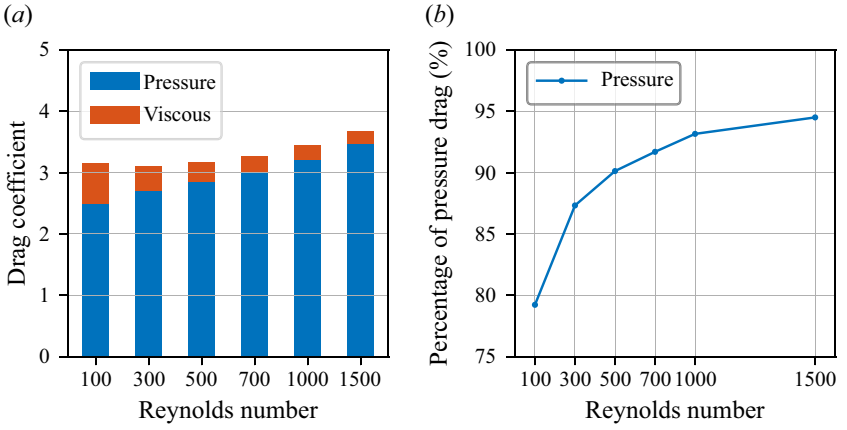


Figure 10. Component analysis of (*a*) drag force caused by pressure and skin friction at different Reynolds numbers, and (*b*) the proportion of pressure drag in the total drag.

$$\tau_w = \mu \cdot \left(\frac{\partial v_t}{\partial n} \right)_{\text{Surface}}, \tag{4.2}$$

where p and τ_w are the normal stress and shear stress on the cylinder surface, respectively, v_t is the velocity along the cylinder surface and n is the normal direction.

Figure 10 shows the magnitudes of pressure drag, skin friction drag and the proportion of pressure drag in total drag at different Reynolds numbers. It can be observed that, as the Reynolds number increases, the proportion of pressure drag monotonically increases, growing from 79.2 % at $Re = 100$ to 94.5 % at $Re = 1500$. Under an invariant aerodynamic shape and inflow velocity, the primary mechanism of AFC was to suppress the shedding of vortices at the rear end of the cylinder (Wang & Feng 2018). This indicates that the significant drag-reduction effect was mainly attributed to the reduction of pressure drag caused by flow separation. As the Reynolds number increased, the proportion of pressure drag in the overall drag force increased, leading to a more pronounced drag-reduction effect if the DRL controller effectively manipulates the flow separation.

Tang *et al.* (2020) also proposed a compelling explanation. The flow around a cylinder can be decomposed into a superposition of steady base flow and vortex-shedding components. The base flow is numerically simulated using a symmetric boundary condition at the equatorial plane of the cylinder. The results showed that the drag force on the cylinder controlled by DRL was consistent with the drag force of the base flow, which indicates that the drag reduction of AFC using DRL mainly originates from vortex shedding, and the drag generated by vortex shedding is primarily attributable to the pressure drag component. Straightforwardly, under high Reynolds number conditions, the increased drag caused by the pressure component (both in absolute value and proportion) allows the DRL agent greater potential for flow control. When the DRL agent finds the optimal control rate, it leads to a decrease in C_D.

The results of the DF-based SAC algorithm are presented in figure 11. The entire training process is parallelized across five environments provided by DRLinFluids. The algorithm successfully learned to perform active flow control, resulting in a continuous reduction of drag and suppression of lift. In the absence of actuation, C_D oscillates periodically around a mean value, as shown in figure 11(*a*). The mean value of C_D is 3.20, with a std of 0.283 for C_D and 2.17 for C_L. With DF-DRL-based AFC, the mean C_D is reduced to 2.17,

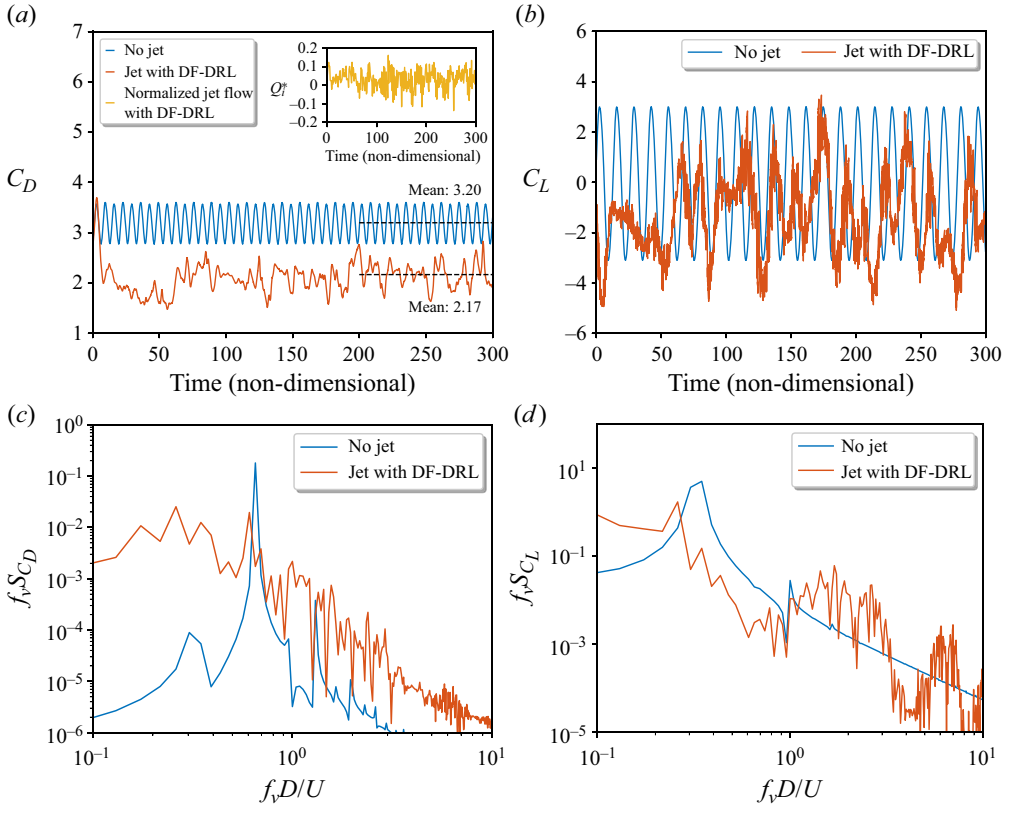


Figure 11. (*a*) Evolution of C_D for the cylinder without (no jet) and with (DF-DRL cylinder) AFC at $Re = 500$, as well as the associated normalized flow rate of the jet flow. (*b*) Temporal variations in smoothed C_L for the cylinder without (no jet) and with (DRL cylinder) AFC in $Re = 500$. (*c*) Power spectral density (PSD) of C_D during the period of non-dimensional time ranging from 200 to 300 and (*d*) PSD of C_L during the period of non-dimensional time ranging from 200 to 300.

corresponding to a drag reduction of approximately 32.2 %. Furthermore, the fluctuation of C_D is suppressed, as indicated by the reduced std value of 0.252. Meanwhile, the std value of C_L is decreased slightly to 1.61.

Power spectrum analyses are conducted to compare C_D and C_L of the cylinder with and without AFC, and the results are presented in figures 11(*c*) and 11(*d*). The power spectrum curves for both C_D and C_L of the plain cylinder exhibit a distinct peak. This indicates a series of distinct vortex shedding at this frequency, contributing to the majority of the energy required for the mean drag and the fluctuation. By contrast, the peaks disappear in the power spectrum curves of C_D and C_L of the cylinder with DF-DRL-based AFC.

The results presented in figure 12 demonstrate that the turbulent conditions are relatively weak at a Reynolds number of 1000. Precisely, for the plain cylinder, the mean C_D is measured to be 3.48 with a std of 0.455. In addition, C_L exhibits a std of 2.76. However, when AFC is implemented, a significant reduction in the mean C_D is achieved, resulting in a value of 1.86, corresponding to a drag reduction of approximately 46.55 %. Moreover, the std of C_D decreases to 0.31, indicating a more consistent behaviour of the cylinder under flow control. Notably, the std of C_L is also markedly reduced to 1.61, which is highly desirable for suppressing the lift force and mitigating flow-induced instability of the cylinder.

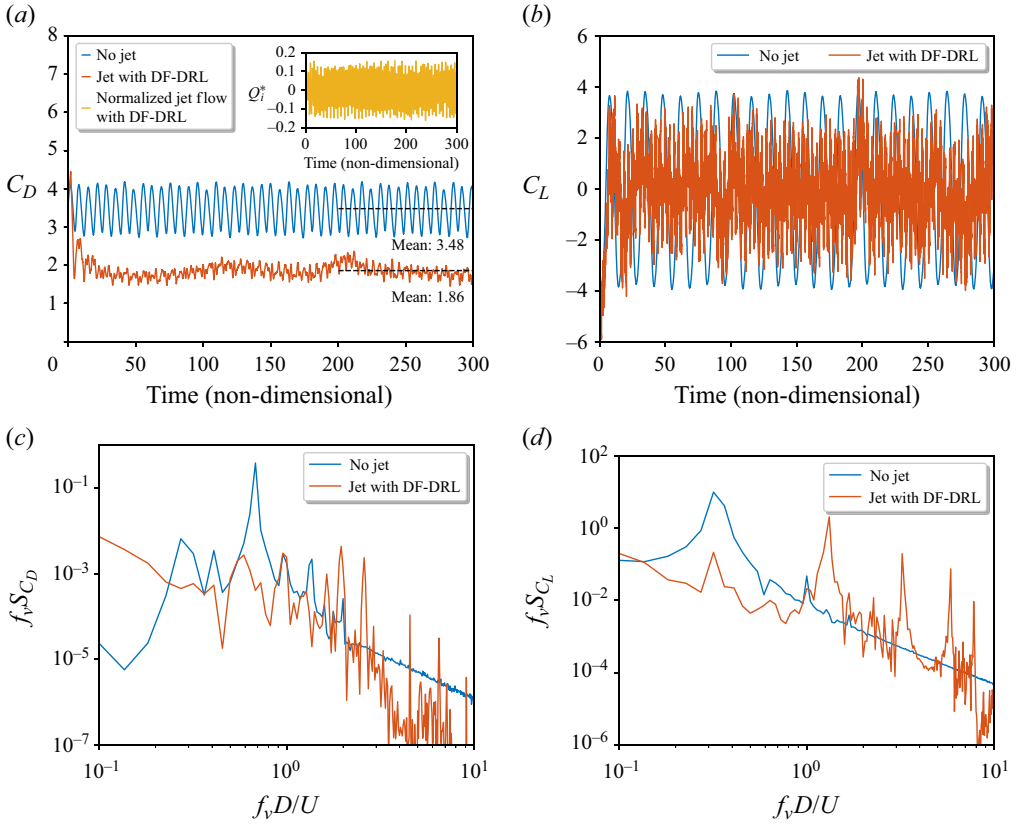


Figure 12. (*a*) Evolution of C_D for the cylinder without (no jet) and with (DF-DRL cylinder) AFC at $Re = 1000$, as well as the associated normalized flow rate of the jet flow. (*b*) Temporal variations in smoothed C_L for the cylinder without (no jet) and with (DF-DRL cylinder) AFC at $Re = 1000$. Panels (*c*,*d*) show the PSD of C_D and C_L, respectively, during the period of non-dimensional time ranging from 200 to 300.

The power spectrum analyses of C_D and C_L for the cylinder with and without active flow control are presented in figures 12(*c*) and 12(*d*). The power spectrum curves for both C_D and C_L of the plain cylinder show an obvious peak, indicating the presence of a regular vortex shedding of significant energy. In contrast, when AFC is implemented, the peak in the power spectrum curves of C_D and C_L for the cylinder is eliminated, indicating that the jet actuation has completely disrupted the regular vortex shedding.

Figure 13 displays the instantaneous flow field around a circular cylinder, with and without active control. The impact of controlled jet flow on reducing the aerodynamic force acting on the cylinder is explained in terms of the flow pattern. In figures 13(*a*) and 13(*c*), which represent conditions for $Re = 500$ and 1000, respectively, a vortex-shedding pattern is observed for the plain cylinder, as expected. This alternate vortex-shedding pattern directly contributes to fluctuations in both C_D and C_L, as demonstrated in figures 11 and 12.

Figures 13(*b*) and 13(*d*) illustrate the impact of fluctuating actuation on the vortex-shedding pattern. The alternate vortex shedding is suppressed, reducing fluctuations of both C_D and C_L. Meanwhile, an elongated recirculation bubble is formed in the near wake, associated with increased pressure and a reduction in drag force. The elongated wake implies a reduced curvature of the shear layer, corresponding to increased

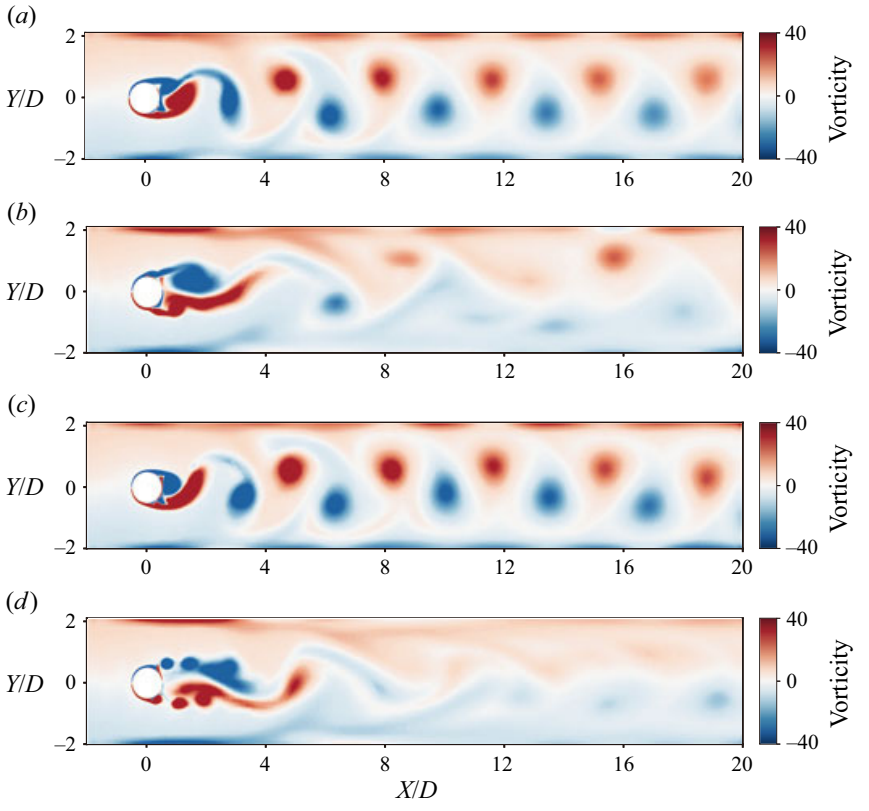


Figure 13. Typical snapshots of the vorticity channel around the cylinder at $Re = 500$ corresponding to the flow without (a) and with (b) control for $Re = 1000$ corresponding to the flow without (c) and with (d) control.

pressure at the rearward cylinder side. As a result, the cylinder with AFC experiences less drag.

Figure 14 depicts the mean vorticity contours around a cylinder, with and without active control. The increase in the recirculation zone is obvious at $Re = 500$ and 1000 and illustrates the effective control strategy learned by the DRL agent. The results demonstrate that well-trained DRL agents, which utilize a single surface pressure sensor's temporal information as the state, can achieve efficient control even under flow conditions with strong nonlinearity and various Reynolds numbers.

It is worth noting that learning a DRL-based control law for the AFC task presents a significant challenge in utilizing a single surface pressure sensor as the state in weak turbulent conditions. However, these results demonstrate the efficacy of DF-DRL-based AFC of a circular cylinder with sparse surface pressure sensing and offer a promising avenue for reducing drag and enhancing AFC performance in fluid dynamics systems.

4.3.2. *Case 2: 3-D turbulent flow around a circular cylinder*

Further consideration is given to a more complex flow scenario that closely resembles real-world conditions. Turbulent flow around a 3-D cylinder with a circular cross-section at $Re = 10\,000$ is studied in this section. The entire computational domain was discretized using polyhedral meshes of varying sizes. The mesh is directly generated from the 3-D simulation region to accurately simulate the physics and minimize numerical diffusion,

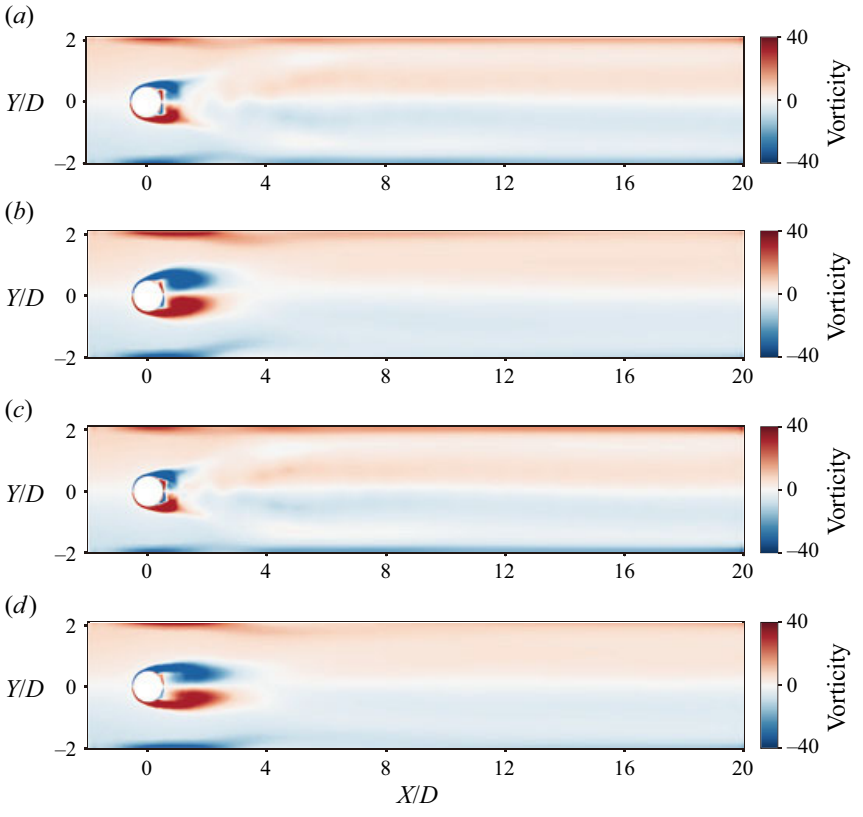


Figure 14. Mean vorticity contours around the cylinder at $Re = 500$ corresponding to the flow without (a) and with (b) control and $Re = 1000$ corresponding to the flow without (c) and with (d) control.

consisting of a total of 1.33 million grids and 15 layers of boundary mesh on the surface of the cylinder. The dimensions of the computational domain are set following the study conducted by Navrose & Mittal (2013). The distances from the inlet (upstream) and outlet (downstream) to the surface of the cylinder are set at $10D$ and $25.5D$, respectively. Additionally, the distances from the top and bottom surfaces to the cylinder surface are both $10D$. The cylinder has a spanwise distance of $4D$, which has been depicted in figure 15.

The velocity on the surface of the cylinder is subject to a no-slip condition, ensuring that the fluid adheres to the cylinder's surface. At the upstream boundary, free-stream values are assigned to the velocity. The stress vector is set to zero at the downstream boundary. On the remaining boundaries, the normal component of the velocity and the tangential component of the stress vector are prescribed a zero value in both directions. Throughout the time-marching solution process, the position of the cylinder, its velocity and the boundary conditions are updated at each nonlinear iteration to capture the evolving flow dynamics accurately. The results of mesh convergence are listed in table 2.

Subsequently, we conducted AFC experiments on three different configurations for 3-D flow around a cylinder. These configurations include: (scheme A) flow control using a single surface pressure sensor as the state input for a DF-DRL-based actuator; (scheme B) flow control using a single surface pressure sensor as the state input for a vanilla DRL-based actuator; and (scheme C) flow control using wake filed velocity sensors as

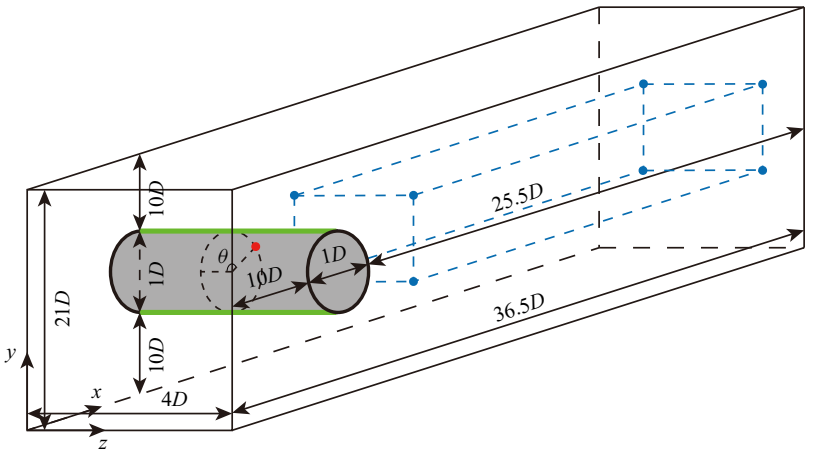


Figure 15. Schematics of the computational domain for 3-D turbulent flow around a cylinder. The red dot represents the location of a surface pressure sensor, with 0 degrees corresponding to the most front of the cylinder and θ equal to 135 degrees. The green bar-shaped regions on either side of the cylinder represent the jet actuators. The blue boxes depict sensor clusters deployed in the wake region. The wake sensors are deployed at equal intervals along the x, y and z directions, forming an $8 \times 3 \times 3$ grid configuration.

	Mean of C_d	r.m.s. of C_L	S_t
Gopalkrishnan (1993)	1.186	0.384	0.193
Bishop, Hassan & Saunders (1997)	—	0.463	0.201
Norberg (2003)	—	0.394	0.202
Dong *et al.* (2006)	1.143	0.448	0.203
Fan *et al.* (2020)	1.192	0.482	0.204
Present	1.151	0.431	0.200

Table 2. Comparison of integral flow quantities in the flow past a 3-D circular cylinder at $Re = 10\,000$. Here, C_d is the drag coefficient, C_L is the root-mean-square (r.m.s.) value of lift coefficient and S_t is the Strouhal number.

the state input for a vanilla DRL-based actuator. The results of the learning curves during the training stage are shown in figure 16.

Figure 16(*a*) shows the evolution of the mean drag coefficient along with the increase of DRL training episodes with and without DF lifting. The results indicate that both the method and the vanilla DRL method achieve significant drag-reduction effects. Compared with the case without control, the mean drag was reduced to 0.822, 0.867 and 1.01 for schemes *A*, *B* and *C*, respectively. However, some significant differences can be observed from the reward learning curves, as described in figure 16(*b*). The reward curve based on the DF-DRL method shows a steadily increasing trend throughout the entire episode, eventually reaching around -76. On the other hand, vanilla DRL, especially with a surface pressure sensor, exhibits a noticeable instability between episodes 50 and 300, characterized by large fluctuations, followed by a relatively low value near 320. Another factor contributing to the difference in reward values is the standard deviation of the lift coefficient, as shown in figure 16(*c*). Schemes *B* and *C* increase lift fluctuations to 0.626 and 0.526, while scheme *A* with the DF-DRL method is more effective in suppressing lift fluctuations, resulting in 0.394.

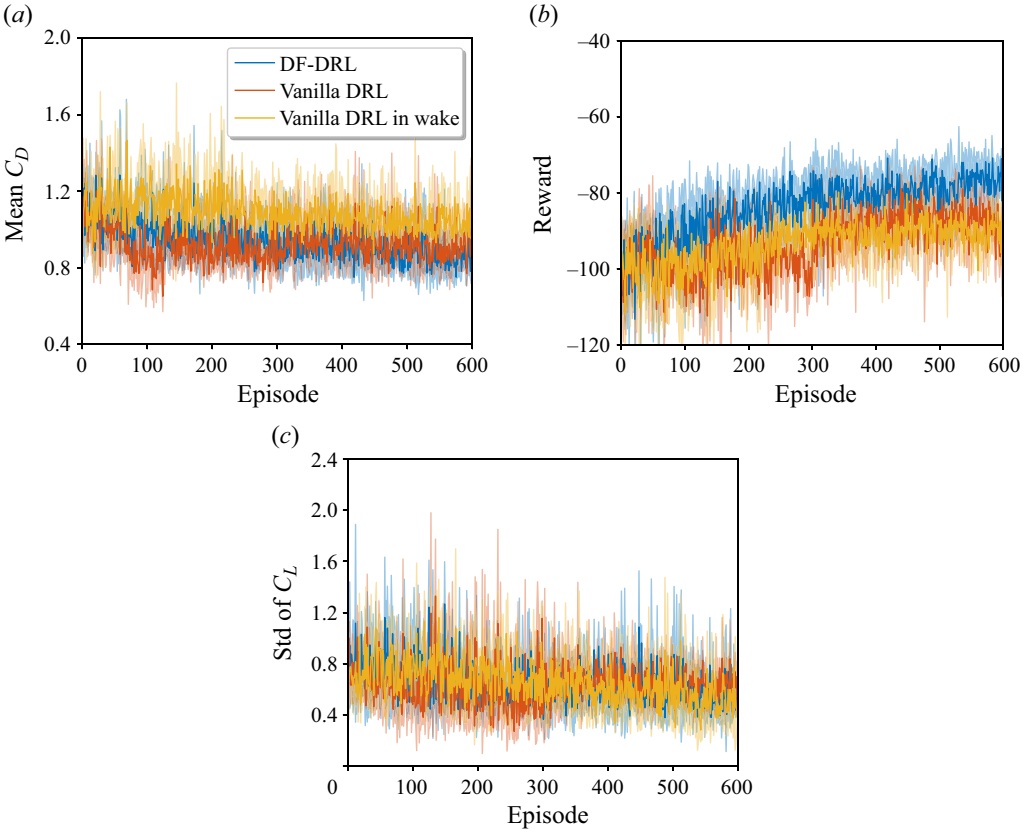


Figure 16. Evolution of (*a*) mean drag reduction, (*b*) reward and (*c*) std lift coefficient of vanilla DRL and DF-DRL method with surface pressure sensing, and vanilla DRL method with wake sensing, respectively. A detailed description of the layout schemes for the two types of sensors is provided in figure 15.

To further clarify the minor differences in drag reduction observed between the scheme *A* and scheme *B*, additional experiments are conducted under different reward function settings, notably those excluding the lift coefficient penalty in vanilla DRL training. The outcomes of these experiments align with the findings of Rabault *et al.* (2019), wherein the omission of the penalty term leads to the adoption of a 'cheating' control strategy by the agent. This strategy involves a significant shift towards unilateral continuous jet actuation, resulting in a decrease in the lift coefficient to 0.62, marking a reduction of 43.6 %. Concurrently, an increase in the r.m.s. of lift from 1.77 to 6.08 is observed, representing an increase of 243.5 %. The implementation of such a control strategy, while beneficial for reducing drag, induces fluctuations in lift that could be deemed unacceptable for certain practical applications. This highlights the critical need for a well-balanced reward function in DRL applications, ensuring that the control strategies employed do not compromise the structural stability and performance in practical scenarios.

After a training phase consisting of 600 episodes, the DF-DRL agent (scheme *A*) is further utilized for testing purposes. The results, as shown in figure 17, reveal a decrease in the drag coefficient from its initial value of 1.151 to 0.822, achieving a drag reduction of 28.6 %, meanwhile, the r.m.s. of lift coefficient decreases from 0.431 to 0.394. Figures 17(*c*) and 17(*d*) demonstrate that DRL-based AFC primarily suppresses drag and lift forces by inhibiting energy in the relatively low-frequency region, which

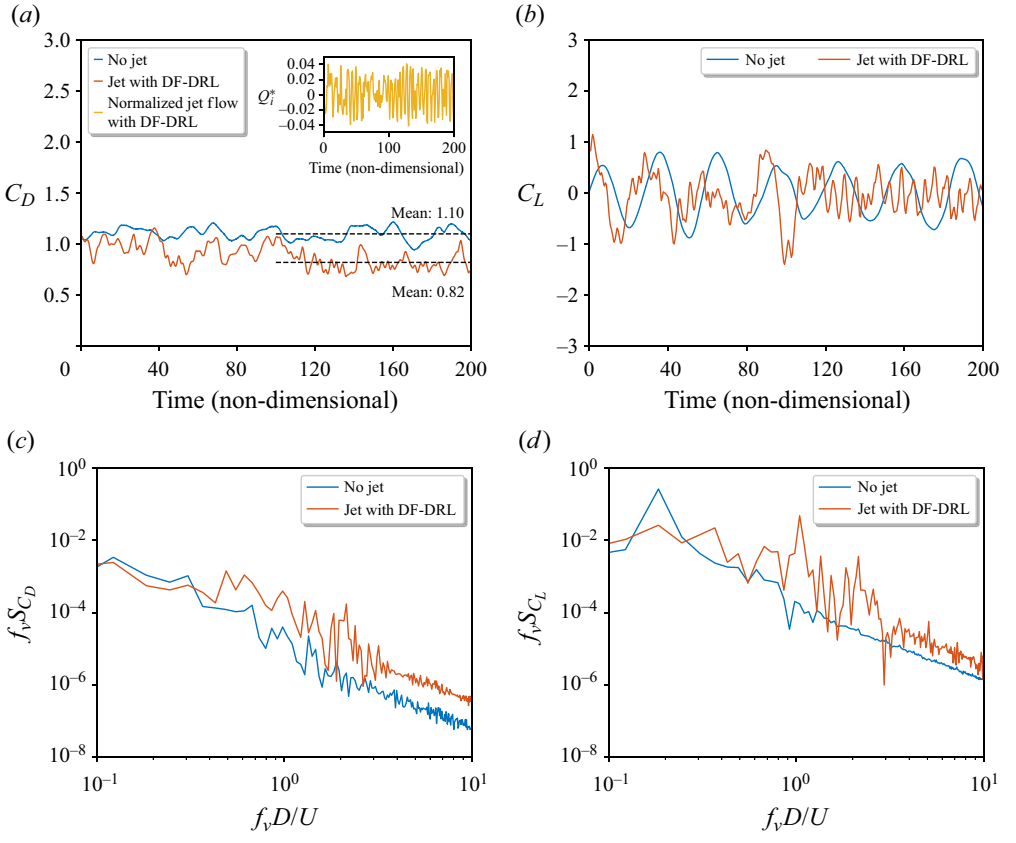


Figure 17. (*a*) Evolution of C_D for the cylinder without (no jet) and with (DF-DRL cylinder) AFC at $Re = 10\,000$, as well as the associated normalized flow rate of the jet flow Q_i^* (yellow line), which is defined in (3.9). (*b*) Temporal variations in smoothed C_L for the cylinder without (no jet) and with (DF-DRL cylinder) active flow control at $Re = 10\,000$. Panels (*c*,*d*) show the PSDs of C_D and C_L, respectively, during the period of non-dimensional time ranging from 100 to 200.

is the dominant source of the energy associated with drag and lift forces. It also should be emphasized that the actuator's behaviour significantly surpasses the basic dichotomy of either continuous blowing or suction (at a constant value) and diverges from the typical operation of a bang-bang controller at its maximum output. Instead, the actuator matures into an adaptive, real-time controller, adept at dynamically responding to the environment's changing conditions.

From the perspective of 3-D vortex structures in the wake field, as shown in figures 18(*a*) and 18(*b*), it is evident that, in the case without any jet injection, small and fragmented vortices are generated in the wake region near the cylinder. As the flow moves away from the cylinder, the smaller-scale vortex structures gradually dissipate, making way for larger-scale vortices. However, when the DF-DRL-based actuator is employed for control, a more regular elongated vortex structure is formed as a result of the blowing and suction of air on both sides, combined with the incoming flow.

This elongated vortex structure, resembling a strip created by the actuator, plays a role in mitigating the generation of fragmented vortices to some extent. As a result, the mid to far wake region exhibits a more regular and alternating pattern of elongated vortices.

The utilization of the DF-DRL-based actuator introduces a controlled airflow that has a significant impact on the wake flow characteristics. The formation of the elongated

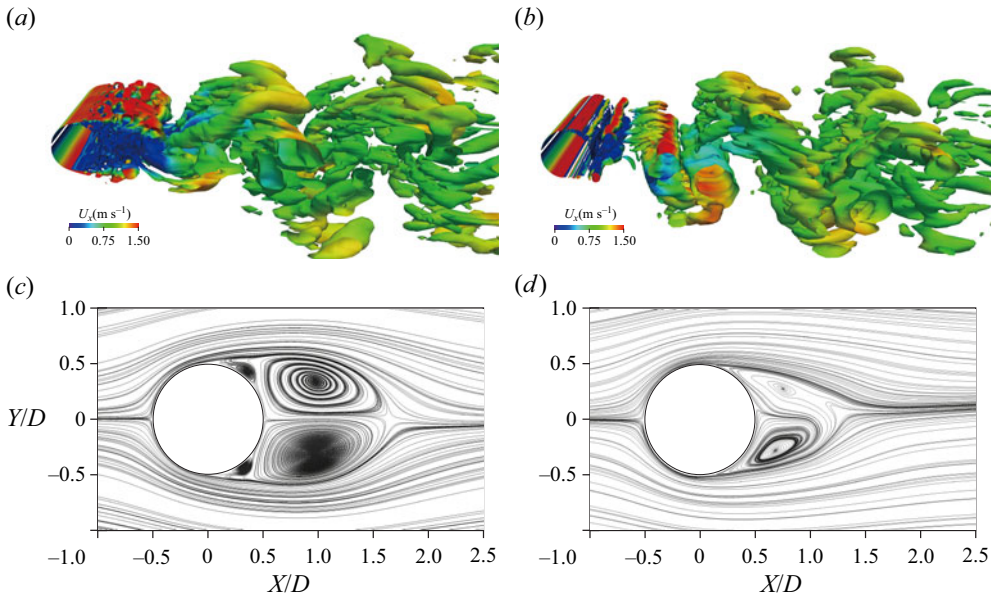


Figure 18. Turbulent flow past a circular cylinder at $Re = 10\,000$. Instantaneous vortical structures close to the cylinder surface coloured by the streamwise velocity (*a*) without and (*b*) with DF-DRL-based control, and streamlines of mean flow (*c*) without and (*d*) with DF-DRL-based control.

vortex structure with more regularity and coherence indicates an improved flow control capability. This controlled vortex structure has the potential to enhance aerodynamic performance, reduce drag or achieve other desired flow control objectives in various engineering applications.

The dimensions of the wake recirculation zone, specifically its width and length, directly influence the base pressure behind a 2-D bluff body (He, Li & Wang 2014; Roy *et al.* 2019). As shown in figures 18(*c*) and 18(*d*), the wake region of the cylinder, controlled by the DF-DRL-based actuator, exhibits characteristics compared with the uncontrolled cylinder that include (i) the disappearance of secondary vortices, (ii) a reduction of 37.1 % in the maximum width of the wake and (iii) a slight decrease of 15 % in the length of the wake. The first two factors lead to a significant decrease in drag, which outweighs the adverse effect of a shorter recirculation zone on drag. This overall effect manifests as a reduction in drag, resulting in a decrease of approximately 28.6 %.

5. Conclusions and outlook

This study presents significant progress towards practical AFC with surface pressure sensors located on a circular cylinder as the sole input for a DRL agent. This approach can potentially advance DRL for real-world applications, such as drag and lift reduction for vehicles and high-rise buildings. The main results of this study are summarized as follows.

Firstly, a novel DRL method called DF-DRL is introduced. Essentially, DF-DRL utilizes prior knowledge to extract one or several features of a nonlinear dynamic system, enabling it to estimate the complete state of the system to the fullest extent possible. This concept aligns with the ideas of pattern recognition and reduced-order modelling. The DF-DRL model combines identification and control, and is not limited to the traditional DRL tasks

that take the state at a certain moment as input. Instead, suitable DF states are selected and lifted to a higher-dimensional vector based on the characteristics of different dynamic systems. Following this step, the vector is used as the state input to the agent. Results show that DF-DRL with a single surface pressure sensor can achieve the same drag-reduction performance as the vanilla DRL method using 147 velocity sensors that fully sample the cylinder wake region.

Secondly, the study investigates the distribution of sensors needed for AFC of the cylinder wake. We conclude that, in low-to-moderate Reynolds number scenarios, a single surface pressure sensor can achieve control results comparable to those obtained with 147 wake sensors under AFC when DF-DRL is used. Additionally, we find that the reward value obtained with a single trailing edge sensor on the cylinder is higher than if the sensor is located at the leading edge, resulting in a lower mean C_D and the std of C_L.

Thirdly, three different flow configurations were examined to verify the effectiveness and robustness of the proposed sensor configuration and DF-DRL method. Results show that the DRL agent utilizing a single surface pressure sensor is capable of controlling wake development behind the circular cylinder, even under more complex scenarios corresponding to higher Reynolds numbers or 3-D turbulent inflow.

Sparse reduced-order modelling (Brunton, Proctor & Kutz 2016*b*; Loiseau *et al.* 2018) is a highly popular research field in which selecting appropriate DF lifting methods for different fluid dynamic systems can enable more accurate estimation using fewer sensor data. Processing these features and using them as DRL states is a promising approach. In the present study, significant reductions in C_D of a cylinder are achieved through two distinct approaches. Specifically, the use of typical DRL resulted in a reduction of 6.4 % utilizing a 4 sensor layout scheme, while dynamic feature sensing with lifting yielded a reduction of 8 % compared with the benchmark performance under a low Reynolds number. Under a turbulent flow around a cylinder at high Reynolds numbers, there is a more significant reduction in drag coefficient, reaching as high as 28.6 % with a DF-DRL controller. The results of this investigation demonstrate that the C_D value of the dynamic feature sensing with lift and DRL (DF-DRL) model is impressively lower than the vanilla model that relies solely on direct sensor feedback, highlighting the efficacy of this approach for improving aerodynamic performance. The DF-DRL method presents a promising approach to significantly reducing the number of required sensors while achieving optimal C_D and C_L reduction performance, which offers a promising pathway for taming complex fluid dynamic systems.

Funding. This study is supported by the National Key R&D Program of China (2021YFC3100702), National Natural Science Foundation of China (52278493, 52108451), Shenzhen Science and Technology Program (SGDX20210823103202018, GXWD20201230155427003-20200823230021001, KQTD20210811090112003) and Guangdong-Hong Kong-Macao Joint Laboratory for Data-Driven Fluid Mechanics and Engineering Applications (2020B1212030001). This work is also supported by the National Science Foundation of China (NSFC) through grants 12172109 and 12172111, by Guangdong province, China, via the Natural Science and Engineering grant 2022A1515011492 and by the Shenzhen Research Foundation for Basic Research, China, through grant JCYJ20220531095605012.

Declaration of interests. The authors report no conflict of interest.

Data availability statement. The DF-DRL model will be uploaded into the DRLinFluids repository. Please check at the following URL: https://github.com/venturi123/DRLinFluids. We invite all users to discuss further and ask for help directly on GitHub through the issue system, and we commit to helping develop a community around the DRLinFluids framework by providing in-depth documentation and help to new users.

Parameter	Value
Numerical time step (non-dimensional dt)	5×10^{-4}
Maximum action amplitude (non-dimensional)	1.5
Action duration	25/44/46 dt for $Re = 100/500/1000$
Number of action steps per episode	100 (Wang *et al.* 2022*a*)

Table 3. Configurations of flow simulation.

Parameter	Value
Actor architecture	512×512 (two fully connected layers)
Critic architecture	512×512 (two fully connected layers)
Actor leaning rate	3×10^{-4}
Critic leaning rate	2×10^{-4}
Discount factor	0.97
Alpha	0.2
Optimizer	Adam (Kingma *et al.* 2015)
Dynamic feature lifted states $\boldsymbol{S}$	$\mathbb{R}^{30\times 2}$ (double vortex-shedding periods in time dimension)
CPU time per episode	Up to 20 min
Total CPU time of training stage	≈5200 CPUh

Table 4. Hyper-parameters of the present DF-DRL model.

Author ORCIDs.
Qiulei Wang https://orcid.org/0000-0002-1612-8274;
Lei Yan https://orcid.org/0000-0002-0503-7561;
Gang Hu https://orcid.org/0000-0001-6284-0812;
Wenli Chen https://orcid.org/0000-0002-7471-815X;
Jean Rabault https://orcid.org/0000-0002-7244-6592;
Bernd R. Noack https://orcid.org/0000-0001-5935-1962.

Appendix. Hyperparameters

Tables 3 and 4 present the main numerical parameters of both the simulations and the learning algorithm.

References

Achenbach, E. 1968 Distribution of local pressure and skin friction around a circular cylinder in cross-flow up to $Re = 5 \times 10^6$. *J. Fluid Mech.* **34** (4), 625–639.

Altman, N. & Krzywinski, M. 2018 The curse(s) of dimensionality. *Nat. Meth.* **15** (6), 399–400.

Andrychowicz, M., Wolski, F., Ray, A., Schneider, J., Fong, R., Welinder, P., McGrew, B., Tobin, J., Abbeel, P. & Zaremba, W. 2018 Hindsight experience replay. In *Advances in Neural Information Processing Systems*, vol. 30 (ed. I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan & R. Garnett). Curran Associates.

Bellemare, M.G., Dabney, W. & Munos, R. 2017 A distributional perspective on reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning* (ed. D. Precup & Y.W. Teh), pp. 449–458. PMLR.

Bishop, R.E.D., Hassan, A.Y. & Saunders, O.A. 1997 The lift and drag forces on a circular cylinder oscillating in a flowing fluid. *Proc. R. Soc. Lond.* A **277** (1368), 51–75.

Blanchard, A.B., Cornejo Maceda, G.Y., Fan, D., Li, Y., Zhou, Y., Noack, B.R. & Sapsis, T.P. 2021 Bayesian optimization for active flow control. *Acta Mechanica Sin.* **37** (12), 1786–1798.

BRUNTON, S.L., BRUNTON, B.W., PROCTOR, J.L., KAISER, E. & KUTZ, J.N. 2017 Chaos as an intermittently forced linear system. *Nat. Commun.* **8** (1), 19.

BRUNTON, S.L., BRUNTON, B.W., PROCTOR, J.L. & KUTZ, J.N. 2016*a* Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control. *PLoS ONE* **11** (2), e0150171.

BRUNTON, S.L. & NOACK, B.R. 2015 Closed-loop turbulence control: progress and challenges. *Appl. Mech. Rev.* **67** (5), 050801.

BRUNTON, S.L., PROCTOR, J.L. & KUTZ, N.J. 2016*b* Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl Acad. Sci. USA* **113** (5), 3932–3937.

DONG, S., KARNIADAKIS, G.E., EKMEKCI, A. & ROCKWELL, D. 2006 A combined direct numerical simulation–particle image velocimetry study of the turbulent near wake. *J. Fluid Mech.* **569**, 185–207.

DURIEZ, T., BRUNTON, S.L. & NOACK, B.R. 2017 *Machine Learning Control-Taming Nonlinear Dynamics and Turbulence*, vol. 116. Springer.

FAN, D., YANG, L., WANG, Z., TRIANTAFYLLOU, M.S. & KARNIADAKIS, G.EM. 2020 Reinforcement learning for bluff body active flow control in experiments and simulations. *Proc. Natl Acad. Sci. USA* **117** (42), 26091–26098.

FRANÇOIS-LAVET, V., HENDERSON, P., ISLAM, R., BELLEMARE, M.G. & PINEAU, J. 2018 An introduction to deep reinforcement learning. *Found. Trends Mach. Learn.* **11** (3–4), 219–354.

FUJIMOTO, S., VAN HOOF, H. & MEGER, D. 2018 Addressing function approximation error in actor-critic methods. In *Proceedings of the 35th International Conference on Machine Learning* (ed. J. Dy & A. Krause), vol. 80, pp. 1587–1596. PMLR.

GARNIER, P., VIQUERAT, J., RABAULT, J., LARCHER, A., KUHNLE, A. & HACHEM, E. 2021 A review on deep reinforcement learning for fluid mechanics. *Comput. Fluids* **225**, 104973.

GAUTIER, N., AIDER, J.-L., DURIEZ, T., NOACK, B.R, SEGOND, M. & ABEL, M. 2015 Closed-loop separation control using machine learning. *J. Fluid Mech.* **770**, 442–457.

GOPALKRISHNAN, R. 1993 Vortex-induced forces on oscillating bluff cylinders. PhD thesis, Massachusetts Institute of Technology.

GUASTONI, L., RABAULT, J., SCHLATTER, P., AZIZPOUR, H. & VINUESA, R. 2023 Deep reinforcement learning for turbulent drag reduction in channel flows. *Eur. J. Phys.* **46** (4), 27.

HA, D. & SCHMIDHUBER, J. 2018 Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems* (ed. S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi & R. Garnett), vol. 31. Curran Associates.

HAARNOJA, T., ZHOU, A., ABBEEL, P. & LEVINE, S. 2018 Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning* (ed. J. Dy & A. Krause), vol. 80, pp. 1861–1870. PMLR.

HE, G.S., LI, N. & WANG, J.J. 2014 Drag reduction of square cylinders with cut-corners at the front edges. *Exp. Fluids* **55** (6), 1745.

JASAK, H., JEMCOV, A. & TUKOVIC, Z. 2007 OpenFOAM: a C++ library for complex physics simulations. In *International Workshop on Coupled Methods in Numerical Dynamics*, vol. 1000, pp. 1–20. IUC Dubrovnik Croatia.

JI, T., JIN, F., XIE, F., ZHENG, H., ZHANG, X. & ZHENG, Y. 2022 Active learning of tandem flapping wings at optimizing propulsion performance. *Phys. Fluids* **34** (4), 047117.

KINGMA, D.P., BA, J., BENGIO, Y. & LECUN, Y. 2015 In *3rd International Conference on Learning Representations* (ed. Y. Bengio & Y. LeCun), *7–9 May, ICLR, San Diego, CA. USA*.

KORKISCHKO, I. & MENEGHINI, J.R. 2012 Suppression of vortex-induced vibration using moving surface boundary-layer control. *J. Fluids Struct.* **34**, 259–270.

LEE, C., KIM, J., BABCOCK, D. & GOODMAN, R. 1997 Application of neural networks to turbulence control for drag reduction. *Phys. Fluids* **9** (6), 1740–1747.

LI, S., SNAIKI, R. & WU, T. 2021 A knowledge-enhanced deep reinforcement learning-based shape optimizer for aerodynamic mitigation of wind-sensitive structures. *Comput.-Aided Civil Infrastructure Engng* **36** (6), 733–746.

LILLICRAP, T.P., HUNT, J.J., PRITZEL, A., HEESS, N., EREZ, T., TASSA, Y., SILVER, D. & WIERSTRA, D. 2019 Continuous control with deep reinforcement learning. arXiv:1509.02971.

LOISEAU, J.-C., NOACK, B.R. & BRUNTON, S.L. 2018 Sparse reduced-order modeling: sensor-based dynamics to full-state estimation. *J. Fluid Mech.* **844**, 459–490.

MEI, Y.-F., ZHENG, C., AUBRY, N., LI, M.-G., WU, W.-T. & LIU, X. 2021 Active control for enhancing vortex induced vibration of a circular cylinder based on deep reinforcement learning. *Phys. Fluids* **33** (10), 103604.

MEZIĆ, I. 2005 Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dyn.* **41** (1), 309–325.

MNIH, V., BADIA, A.P., MIRZA, M., GRAVES, A., LILLICRAP, T., HARLEY, T., SILVER, D. & KAVUKCUOGLU, K. 2016 Asynchronous methods for deep reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning* (ed. M.F. Balcan & K.Q. Weinberger), pp. 1928–1937. PMLR.

MNIH, V., KAVUKCUOGLU, K., SILVER, D., GRAVES, A., ANTONOGLOU, I., WIERSTRA, D. & RIEDMILLER, M. 2013 Playing atari with deep reinforcement learning. arXiv:1312.5602.

MNIH, V., *et al.* 2015 Human-level control through deep reinforcement learning. *Nature* **518** (7540), 529–533.

NAVROSE & MITTAL, S. 2013 Free vibrations of a cylinder: 3-D computations at $Re = 1000$. *J. Fluids Struct.* **41**, 109–118.

NORBERG, C. 2003 Fluctuating lift on a circular cylinder: review and new measurements. *J. Fluids Struct.* **17** (1), 57–96.

NOVATI, G., VERMA, S., ALEXEEV, D., ROSSINELLI, D., VAN REES, W.M. & KOUMOUTSAKOS, P. 2017 Synchronisation through learning for two self-propelled swimmers. *Bioinspir. Biomim.* **12** (3), 036001.

PINO, F., SCHENA, L., RABAULT, J. & MENDEZ, M.A. 2023 Comparative analysis of machine learning methods for active flow control. *J. Fluid Mech.* **958**, A39.

PINTÉR, J.D. 1995 *Global Optimization in Action: Continuous and Lipschitz Optimization: Algorithms, Implementations and Applications*, vol. 6. Springer Science & Business Media.

RABAULT, J., KUCHTA, M., JENSEN, A., RÉGLADE, U. & CERARDI, N. 2019 Artificial neural networks trained through deep reinforcement learning discover control strategies for active flow control. *J. Fluid Mech.* **865**, 281–302.

RABAULT, J. & KUHNLE, A. 2019 Accelerating deep reinforcement learning strategies of flow control through a multi-environment approach. *Phys. Fluids* **31** (9), 094105.

RABAULT, J., REN, F., ZHANG, W., TANG, H. & XU, H. 2020 Deep reinforcement learning in fluid mechanics: a promising method for both active flow control and shape optimization. *J. Hydrodyn.* **32**, 234–246.

REN, F., RABAULT, J. & TANG, H. 2021*a* Applying deep reinforcement learning to active flow control in weakly turbulent conditions. *Phys. Fluids* **33** (3), 037121.

REN, F., WANG, C. & TANG, H. 2019 Active control of vortex-induced vibration of a circular cylinder using machine learning. *Phys. Fluids* **31** (9), 093601.

REN, F., WANG, C. & TANG, H. 2021*b* Bluff body uses deep-reinforcement-learning trained active flow control to achieve hydrodynamic stealth. *Phys. Fluids* **33** (9), 093602.

ROGHAIR, J., NIARAKI, A., KO, K. & JANNESARI, A. 2022 A vision based deep reinforcement learning algorithm for UAV obstacle avoidance. In *Intelligent Systems and Applications* (ed. K. Arai), Lecture Notes in Networks and Systems, pp. 115–128. Springer International.

ROY, S., GHOSHAL, S., BARMAN, K., DAS, V.K., GHOSH, S. & DEBNATH, K. 2019 Modulation of the recirculation region due to magneto hydrodynamic flow. *Engng Sci. Technol., Intl J.* **22** (1), 282–293.

SCHAARSCHMIDT, M., KUHNLE, A., ELLIS, B., FRICKE, K., GESSERT, F. & YONEKI, E. 2018 LIFT: reinforcement learning in computer systems by learning from demonstrations. *CoRR*. arXiv:1808.07903.

SCHÄFER, M., TUREK, S., DURST, F., KRAUSE, E. & RANNACHER, R. 1996 Benchmark computations of laminar flow around a cylinder. In *Flow Simulation with High-Performance Computers II* (ed. E.H. Hirschel), pp. 547–566. Springer.

SCHULMAN, J., LEVINE, S., MORITZ, P., JORDAN, M.I. & ABBEEL, P. 2015 Trust region policy optimization. In *Proceedings of the 32nd International Conference on Machine Learning* (ed. F. Bach & D. Blei), pp. 1889–1897. PMLR.

SCHULMAN, J., WOLSKI, F., DHARIWAL, P., RADFORD, A. & KLIMOV, O. 2017 Proximal policy optimization algorithms. arXiv:1707.06347.

SILVER, D., *et al.* 2017 Mastering Chess and Shogi by self-play with a general reinforcement learning algorithm. arXiv:1712.01815.

TAKENS, F. 1981 Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980* (ed. D. Rand & L.-S. Young), Lecture Notes in Mathematics, pp. 366–381. Springer.

TANG, H., RABAULT, J., KUHNLE, A., WANG, Y. & WANG, T. 2020 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.

VARELA, P., SUÁREZ, P., ALCÁNTARA-ÁVILA, F., MIRÓ, A., RABAULT, J., FONT, B., GARCÍA-CUEVAS, L.M., LEHMKUHL, O. & VINUESA, R. 2022 Deep reinforcement learning for flow control exploits different physics for increasing Reynolds number regimes. *Actuators* **11** (12), 359.

VERMA, S., NOVATI, G. & KOUMOUTSAKOS, P. 2018 Efficient collective swimming by harnessing vortices through deep reinforcement learning. *Proc. Natl Acad. Sci. USA* **115** (23), 5849–5854.

VIGNON, C., RABAULT, J. & VINUESA, R. 2023 Recent advances in applying deep reinforcement learning for flow control: perspectives and future directions. *Phys. Fluids* **35** (3), 031301.

VIQUERAT, J., RABAULT, J., KUHNLE, A., GHRAIEB, H., LARCHER, A. & HACHEM, E. 2021 Direct shape optimization through deep reinforcement learning. *J. Comput. Phys.* **428**, 110080.

WANG, J. & FENG, L. 2018 *Flow Control Techniques and Applications*. Cambridge Aerospace Series. Cambridge University Press.

WANG, Q., YAN, L., HU, G., LI, C., XIAO, Y., XIONG, H., RABAULT, J. & NOACK, B.R. 2022*a* Drlinfluids: an open-source python platform of coupling deep reinforcement learning and openfoam. *Phys. Fluids* **34** (8), 081801.

WANG, Y.-Z., HUA, Y., AUBRY, N., CHEN, Z.-H., WU, W.-T. & CUI, J. 2022*b* Accelerating and improving deep reinforcement learning-based active flow control: transfer training of policy network. *Phys. Fluids* **34** (7), 073609.

WEAVER, L. & TAO, N. 2013 The optimal reward baseline for gradient-based reinforcement learning. In *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence (2001)*, pp. 538–545. Morgan Kaufmann.

WEBER, T., *et al.* 2018 Imagination-augmented agents for deep reinforcement learning. In *Advances in Neural Information Processing Systems* (ed. I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan & R. Garnett). Curran Associates.

WENG, J., CHEN, H., YAN, D., YOU, K., DUBURCQ, A., ZHANG, M., SU, Y., SU, H. & ZHU, J. 2022 Tianshou: a highly modularized deep reinforcement learning library. *J. Mach. Learn. Res.* **23** (267), 1–6.

XU, H., ZHANG, W., DENG, J. & RABAULT, J. 2020 Active flow control with rotating cylinders by an artificial neural network trained by deep reinforcement learning. *J. Hydrodyn.* **32**, 254–258.

ZHENG, C., JI, T., XIE, F., ZHANG, X., ZHENG, H. & ZHENG, Y. 2021 From active learning to deep reinforcement learning: intelligent active flow control in suppressing vortex-induced vibration. *Phys. Fluids* **33** (6), 063607.

ZHOU, Y., FAN, D., ZHANG, B., LI, R. & NOACK, B.R. 2020 Artificial intelligence control of a turbulent jet. *J. Fluid Mech.* **897**, A27.