Learning in two dimensions and controlling in three: Generalizable drag reduction strategies for flows past circular cylinders through deep reinforcement learning

Michail Chatzimanolakis,[1,2] Pascal Weber,[1,2] and Petros Koumoutsakos[2,*]
[1]*Computational Science and Engineering Laboratory, ETH Zürich, CH-8092, Switzerland*
[2]*Computational Science and Engineering Laboratory, Harvard University, Cambridge, Massachusetts 02138, USA*

(Received 18 August 2023; accepted 10 January 2024; published 4 April 2024)

We investigate drag reduction mechanisms in flows past two- and three-dimensional cylinders controlled by surface actuators using deep reinforcement learning. We investigate 2D and 3D flows at Reynolds numbers up to 8000 and 4000, respectively. The learning agents are trained in planar flows at various Reynolds numbers, with constraints on the available actuation energy. The discovered actuation policies exhibit intriguing generalization capabilities, enabling open-loop control even for Reynolds numbers beyond their training range. Remarkably, the discovered two-dimensional controls, inducing delayed separation, are transferable to three-dimensional cylinder flows. We examine the trade-offs between drag reduction and energy input while discussing the associated mechanisms. The present paper demonstrates discovery of transferable and interpretable control strategies for bluff body flows through deep reinforcement learning with limited computational cost.

DOI: 10.1103/PhysRevFluids.9.043902

I. INTRODUCTION

The identification and utilization of drag reduction mechanisms in bluff body flow constitute fundamental elements in the design of aircraft and ships. This has a direct and profound impact on their energy consumption and environmental emissions [1].

Since the early stages of flow control, following the development of boundary layer theory by Prandtl in 1904, circular cylinders have been a standard model for studying drag reduction mechanisms [2]. These approaches can be broadly categorized into passive and active methodologies. Passive methods suggest modifications to the body's surface, such as wall protrusions or surface roughness [3,4]. Active methods involve the use of surface actuators, including mass transpiration, tangential belt actuators, or plasma actuators [5–8]. The discovery of control strategies has heavily relied on computer simulations. In Ref. [9], a series of simulations investigated passive drag reduction through partial leeward porous coatings on a cylinder's surface. In another work, active and passive drag reduction strategies were derived for a three-dimensional flow around a cylinder by examining the sensitivity of drag with respect to perturbations of the surface velocity and roughness. The control reduced drag by 20% with a maximum control velocity 2% of the free-stream velocity at Reynolds number $\mathrm{Re} = 190, 300, 1000$ [10]. In later work, optimization of control parameters through evolutionary algorithms has also proven to be a reliable approach. Automated discovery of control strategies for cylinder flows was first introduced to optimize tangential actuators on cylinder surfaces, exhibiting up to 50% reduction in drag at a Reynolds number of 500 [5]. Further

*Corresponding author: petros@seas.harvard.edu

investigations extended these findings to three-dimensional settings, achieving a 40% reduction in drag [5].

Reinforcement learning (RL) has been introduced in fluid dynamics over the last decade for developing control strategies for synchronizing multiple hydrodynamically interacting swimmers [11–15]. The scope of these applications has been extended in several directions, including effective navigation in vortical flows [16] and the semisupervised discovery of subgrid-scale models in turbulent flows [17–19].

In recent years, there have been several works that apply RL to find optimal control strategies for active drag reduction. In Ref. [20], the authors used RL to control a cylinder's angular velocity in order to reduce drag, at a relatively modest Reynolds number of 100. A crucial component in the formulation of an RL problem is the choice of state and action. In the context of active flow control, the magnitude and therefore the required power for the control has to be carefully considered. The work presented in Ref. [21] reports successful drag reduction for the planar flow past a cylinder at $\mathrm{Re} = 100$; however, the proposed approach seems to suffer from sampling multiple quantities from the cylinder's wake for the RL state and from using unrealistically large actuation velocities for the employed actions. The active control with mass flow actuators using RL was also examined in different works for Reynolds numbers ranging from 100 to 1000 in Refs. [22,23], where the achieved drag reductions range from 17%, to 38%. Similarly, other work in Ref. [24] has used RL in order to learn active control policies for elliptical cylinders at $\mathrm{Re} = 100$, achieving drag reductions up to 10%. The effect of the chosen actuation strengths is not discussed and the feasibility of the chosen values are largely ignored. A similar approach to Ref. [21] is chosen in Ref. [25], where the previous work from is extended to Reynolds numbers up to 2000. That work touches on the generalization capabilities of the learned policies to higher Reynolds numbers through Transfer Learning, which would be a desirable trait of a computed optimal policy. Reference [26] moved in that direction, by extensively training an RL agent at low Reynolds numbers and then by complementing the training process with a few simulations at higher Reynolds numbers. Their results successfully reduced drag for a three-dimensional cylinder whose wake was controlled by two smaller ones. Recent work by Ref. [27] also proposes strategies in a three-dimensional setting, but with the goal of reducing skin friction in a channel flow. The authors from [28] combined simulations and experiments to discover control policies for increasing the system power gain efficiency of a system of three cylinders, showing the transferability of the insights from simulation to reality. The authors of this study reported a wall-clock time of more than three weeks for their training to complete, exemplifying the high computational cost of such tasks.

The present work aims to tackle the four aforementioned issues when RL is applied for active flow control: generalizing to high Reynolds numbers, doing so with realistic actuation strengths, transferring of the results to realistic three-dimensional flows and achieving a manageable computational cost. By randomizing the Reynolds number and the control strength during training, we showcase how the generalization capabilities can be improved and what effect the choice of the actuation strength has. Using this framework, we explore drag reduction mechanisms for the flow past a circular cylinder for Reynolds numbers in the range $\mathrm{Re} = [500, 8000]$ using RL. For the strongest actuation, the drag reduction achieved is competitive to existing results that required stronger actuation. Furthermore, the generalisation capabilities to three-dimensional flows at $Re = 1000$, 2000 and 4000 is demonstrated. In these cases a drag reduction from 14.1%, 18.2% and 16.1% was achieved, respectively. Compared to the—relatively scarce—results for three-dimensional flows, our approach seems to be advantageous in terms of actuation strength and computational cost. For example, Ref. [5] reports a 40% reduction in drag (for a lower Reynolds number of 500), but uses moving tangential actuator belts. Similarly, the configurations suggested in Refs. [26,28] utilize additional cylinders to control the wake, achieving reductions in drag in the order of 30%. The computational cost is alleviated by leveraging an an efficient implementation of Adaptive Mesh Refinement [29] for the fluid dynamics simulations and by using a parallel implementation of V-RACER with Remember and Forget for Experience Replay (ReF-ER) [30] in Korali [31] for the stochastic search involved

in DRL. Our approach allows for parallel training with direct numerical simulations (DNSs) at a manageable cost.

This paper is structured as follows: In Sec. I A, we introduce the governing equations and numerical method used for our direct numerical simulations. The RL method is discussed in Sec. I B. Building on that background, we formulate drag reduction as a RL problem in Sec. II. Our results, presented in Sec. III, are organized as follows: First, we introduce the experimental setup and examine the mechanism for drag reduction on one example in Sec. III A. Then, in Sec. III B, we analyze the effect of the actuation velocity on the results. The transferability of the learned policy to different Reynolds numbers is discussed in Sec. III C, and we further investigate the influence of the two factors on the actions taken in Sec. III D. We explore extending the results to 3D in Sec. III E and conclude in Sec. IV.

A. Direct numerical simulations

We perform two- and three-dimensional DNSs of the flow past a cylinder by solving the incompressible Navier-Stokes equations

$$\nabla \cdot \boldsymbol{u} = 0, \quad \frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \boldsymbol{u}, \tag{1}$$

where $\boldsymbol{u}$, ρ, p, and ν are the fluid velocity, density, pressure, and kinematic viscosity. The no-slip boundary condition is enforced on the cylinder surface with a prescribed velocity $\boldsymbol{u}^s$ through the penalization approach [32–34], which augments the Navier-Stokes equations with a penalty term $\lambda\chi(\boldsymbol{u}^s - \boldsymbol{u})$. Here $\lambda \in \mathbb{R}$ is the penalization coefficient and χ is the characteristic function that takes values $\chi = 1$ inside the cylinder and $\chi = 0$ outside. The simulations are performed with CubismAMR software, an adaptive version of the Cubism library, which partitions the simulation domain into cubic blocks of uniform resolution that are distributed to multiple compute nodes for cache-optimized parallelism [35]. CubismAMR organizes these blocks in an octree data structure (for three-dimensional simulations) or a quadtree data structure (for two-dimensional simulations), allowing for adaptive mesh refinement in different regions. We refer to Refs. [29,36] for details on the implemented numerical scheme and code validation results.

B. Reinforcement learning

RL algorithms solve Markov decision processes (MDPs), which are defined by the tuple $(\mathcal{S},\ \mathcal{A},\ r,\ p)$ consisting of a state-space $\mathcal{S}$, an action-space $\mathcal{A}$, a function $r : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ which is the reward of transitioning to state $\boldsymbol{s}' \in \mathcal{S}$ from state $\boldsymbol{s} \in \mathcal{S}$ by taking action $\boldsymbol{a} \in \mathcal{A}$, and an unknown, stochastic transition map $p(\boldsymbol{s}'|\boldsymbol{a}, \boldsymbol{s})$, which is the probability of transitioning to $\boldsymbol{s}'$ from $\boldsymbol{s}$ by taking action $\boldsymbol{a}$.

On the MDP, we define a stochastic policy via a probability distribution $\pi(\boldsymbol{a}|\boldsymbol{s})$, which allows sampling an action for a given state. The goal of RL is to find the optimal policy:

$$\pi^\star = \arg\max_\pi V^\pi(\boldsymbol{s}), \quad \forall \boldsymbol{s} \in \mathcal{S}. \tag{2}$$

that maximizes the state-value function, defined as

$$V^\pi(\boldsymbol{s}) = \mathbb{E}_{p,\pi}\left[\sum_{i=0}^{N_i-1} \gamma^i r(\boldsymbol{s}_i, \boldsymbol{s}_{i+1}, \boldsymbol{a}_i)|\boldsymbol{s}_0 = \boldsymbol{s}\right], \tag{3}$$

where $\gamma^i \in [0, 1)$ is known as the discount factor and N_i are the total number of transitions between states.

The optimal policy is computed based on interactions of an RL agent with the environment. At every step i, the agent chooses an action $\boldsymbol{a}_i$ based on the observation of the state $\boldsymbol{s}_i$ from the environment. The environment then transitions to a new state $\boldsymbol{s}_{i+1}$ and returns a reward $r(\boldsymbol{s}_i, \boldsymbol{s}_{i+1}, \boldsymbol{a}_i)$.

In off-policy methods, transitions are collected in a replay memory and an approximation to the optimal policy $\pi(\boldsymbol{a}|\boldsymbol{s};\boldsymbol{\vartheta})$ with parameters $\boldsymbol{\vartheta}$ is learned. In actor-critic methods, a value function $V(\boldsymbol{s};\boldsymbol{\vartheta})$ approximation is learned as well. State-of-the-art DRL employs neural networks $\mathrm{NN}(\boldsymbol{s};\boldsymbol{\vartheta})$ as universal function approximators, where the weights $\boldsymbol{\vartheta}$ of the neural network are typically optimized using stochastic gradient descent. For the present paper, we use V-RACER with ReF-ER [30] implemented in Korali [31]; this is an off-policy actor-critic DRL method, proven successful in several scientific applications [17,18] and recently generalized to multiple RL agents [37].

II. REINFORCEMENT LEARNING FOR DRAG REDUCTION

The effective deployment of RL requires an appropriate choice of states, actions, and reward function. Here we deploy an RL agent that interacts with the environment at discrete times $t_i = t_0 + i\Delta t$, for $i = 0, \ldots, N_i - 1$, where N_i is the total number of actions taken before a set of interactions, also referred to as an episode, terminates. The times at which the actions are taken are equally spaced in time, with spacing Δt, and the agent starts the interaction after a transient time t_0. We set $\Delta t = 0.1U/D$, where U is the cylinder velocity and D its diameter.

A. Actions

We deploy $N_a = 8$ uniformly distributed mass transpiration actuators on the cylinder surface [5,22], each with a time-dependent strength $a_j^{t_i} \in [-1, 1]$, $j = 0, \ldots, N_a - 1$. Actuator j imposes a radial velocity on the surface of the cylinder,

$$v_j^r(\theta) = cU a_j^{t_i} \cos\left(\frac{\pi(\theta - \theta_j)}{\theta_a^j}\right), |\theta - \theta_j| \leqslant \theta_a^j/2, \tag{4}$$

where c is a constant that varies during training, θ_a^j is the angle that corresponds to the arc length over which actuator j is active, $\theta_j = j\frac{\pi}{4}$ is the angle on the cylinder surface where each actuator is centered. One set of actions $\boldsymbol{a}_i$ consists of picking the actuator strengths under the constraint that their mean value $\sum_{j=1}^{N} a_j^{t_i}/N_a$ is zero, ensuring a zero total mass flux caused by actuation.

B. State

We observe the cylinder lift and drag coefficients as well as the pressure and vorticity on its surface in $N_s = 16$ uniformly placed sensors. For each of the sensors, pressure and vorticity are averaged over the cylinder surface in an area covered by an arc length of 10 °. We also include the Reynolds number and c value from Eq. (4) in our state representation, resulting in a 36−dimensional state.

C. Reward

The reward function entails the averaged total drag and a penalty term for the actuation strengths, expressed as

$$r(\boldsymbol{s}_i, \boldsymbol{s}_{i+1}, \boldsymbol{a}_i) = -\frac{1}{\Delta t}\int_{t_i}^{t_{i+1}} C_D(t)\,\mathrm{d}t - \frac{w}{N_a}\sqrt{\sum_{j=0}^{N_a}\left(a_j^{t_i}\right)^2}, \tag{5}$$

where C_D denotes the cylinder drag coefficient. The first term implies that the agent minimizes the mean drag. The second term is a regularization term with coefficient w that penalizes strong actions and therefore balances the trade-off between drag reduction and energy required for the actuation.

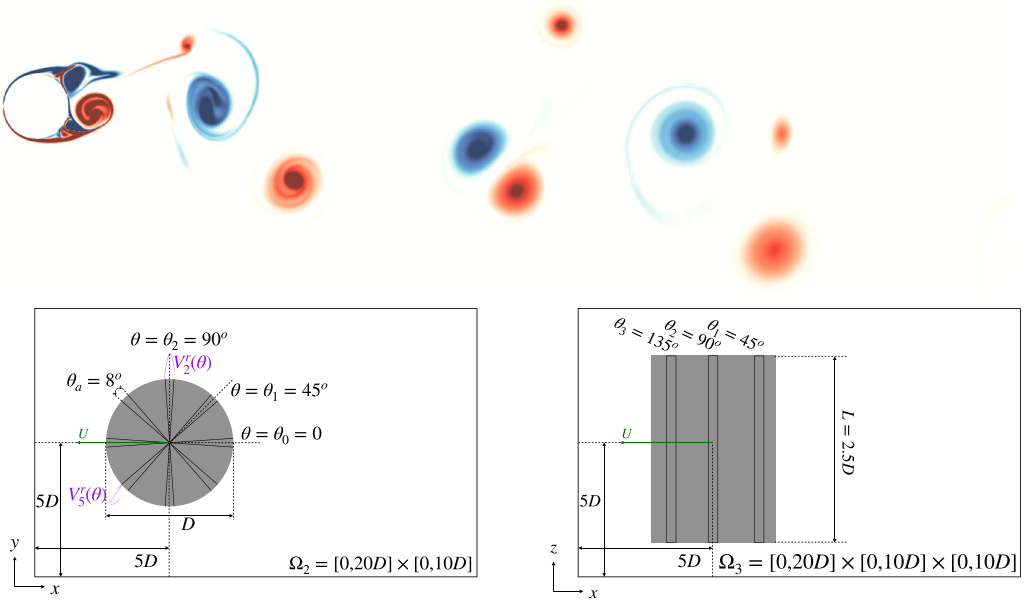

FIG. 1. Illustration of the RL setup. Top: Snapshot of the vorticity field at $T = 200$ for the two-dimensional flow at Re $= 4000$. Bottom left: Sketch of the actuators and 2D simulation domain. Bottom right: Sketch of the actuators and 3D simulation domain, top view.

The computed drag reduction for each case refers to the time-averaged drag coefficient throughout each simulation, and is computed from

$$\frac{\int_{T_s}^{T_e} C_D(t)dt - \int_{T_{s,b}}^{T_{e,b}} C_{D,\text{baseline}}dt}{\int_{T_{s,b}}^{T_{e,b}} C_{D,\text{baseline}}(t)dt}, \tag{6}$$

where $C_{D,\text{baseline}}$ and $C_D(t)$ are the instantaneous drag coefficients without control and when using a trained policy and respectively. Also, we use $T_{s,b} = T_s = 0.1$ and $T_e = T_{e,b} = 30$ for the 3D simulations and $T_{s,b} = 50$, $T_s = T_{e,b} = 200$ and $T_e = 300$ for the 2D simulations.

III. RESULTS

We perform two- and three-dimensional simulations of flows past circular cylinders. The two-dimensional simulations are performed in a rectangular domain $\Omega_2 = [0, 20D] \times [0, 10D]$, with a cylinder of diameter D placed at $(5D, 5D)$. The three-dimensional simulations use a rectangular domain $\Omega_3 = [0, 20D] \times [0, 10D] \times [0, 10D]$, with a cylinder of diameter D and length $L = 2.5D$ placed at $(5D, 5D, 5D)$. In both cases, the cylinder is impulsively set into motion with velocity U. Figure 1 shows the aforementioned setup. Adaptive mesh refinement takes place according to the magnitude of the vorticity field and the finest resolution depends on the minimum grid spacing allowed (denoted by h). We find that decreasing the value of h below $D/200$ yields a smaller than 2% change in the computed mean drag, and thus choose to use this value for our simulations. Nondimensional time is scaled as $T = tU/D$ while the time step is determined according to the Courant Friedrichs Lewy (CFL) condition, with a Courant number of 0.5. Commencement of vortex shedding is accelerated by adding a small rotation of the cylinder along the z direction. For $0.25 < T < 0.5$, the z component of its angular velocity is

$$\omega_z = \frac{0.04U}{D} \sin(8\pi T). \tag{7}$$

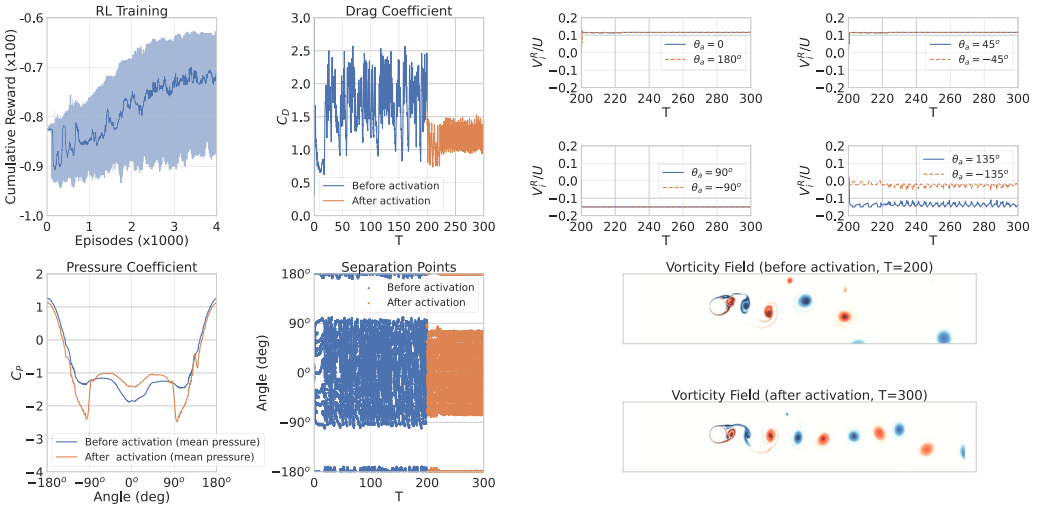

FIG. 2. Results for policy π (no regularizer weight) for a two-dimensional flow past a cylinder at Re = 4000, with maximum actuation velocity up to 15% of the cylinder velocity. Top left: Cumulative reward as a function of episodes simulated, plotted with a 95% confidence interval. Top center: Drag coefficient as a function of time for the converged policy. Top right: Actions as fraction of cylinder velocity as a function of time. Bottom left: Time-averaged cylinder pressure coefficient, before and after actuators are activated. Bottom center: Angle of the points with zero shear (separation points) on cylinder surface. Bottom right: Vorticity field at $T = 300$.

We train the RL agent in 2D simulations with a Reynolds number randomly sampled from Re = 1000, 2000, and 4000. For each episode, the maximum actuation velocity is determined by sampling c, Eq. (4), in the interval [0.05,0.15]. We examine two policies π and π^w. The first policy does not employ a regularizer ($w = 0$) whereas the second policy does (we set $w = 0.1$). Each episode consists of a simulation where actuation starts at $T = 200$, when the wake of the cylinder is developed and vortex shedding has commenced (also see Fig. 1 for a snapshot of the vorticity field) and ends at $T = 250$. Each training was run on 32 compute nodes, each equipped with two AMD EPYC 7763 of 64 cores and lasts 12 h. As can be seen in the top left plot of Fig. 2, this allows simulating approximately 4000 episodes and achieves a converged policy approximately after episode 3000. Note that the resolution used during training used a minimum grid spacing of $2h$ as a compromise between time-to-solution and accuracy of each episode simulated. After training, the found policies are tested in a fully resolved simulation for different Reynolds numbers and maximum actuation velocities (determined by the value of c). Table I compares the mean drag for the uncontrolled case $T \in [100, 200]$ and for the controlled case where $T \in [200, 300]$.

A. Mechanism for drag reduction

To understand the mechanism for drag reduction, we present the results for Re = 4000 and $c = 0.15$ in Fig. 2. In the top middle plot, the drag coefficient time series before and after the actuators are activated is presented; drag is decreased by almost 33% (see Table I). The fluid velocity imposed by each actuator expressed as a fraction of the cylinder velocity is displayed in the top right panel. Positive values correspond to blowing and negative values to suction. The actuators at the front half of the cylinder ($\pi/2 \leqslant |\theta_a| \leqslant \pi$) suction fluid, while the other actuators blow fluid, which helps the boundary layer remain attached for longer. The vorticity field with actuation (bottom right plot of Fig. 2) at $T = 300$ can be compared with the initial condition at $T = 200$ shown in Fig. 1. The comparison reveals that the cylinder wake becomes narrower and more symmetric. The width of the wake is closely associated with the location of the flow separation

TABLE I. Summary of the two-dimensional simulation results. Each row shows the mean drag coefficient without actuation and the mean drag coefficient for three cases with different values for the maximum actuation velocity c. Results are shown for both policies π and π^w. The percentage written in parentheses indicates the drag reduction.

		Policy π (no regularizer weight)			Policy π^w (regularizer weight $w = 0.1$)		
Re	C_D	$c = 0.05$	$c = 0.10$	$c = 0.15$	$c = 0.05$	$c = 0.10$	$c = 0.15$
500	1.52	1.42 (6%)	1.33 (12%)	1.25 (18%)	1.43 (6%)	1.34 (12%)	1.26 (17%)
1000	1.61	1.45 (10%)	1.31 (19%)	1.20 (26%)	1.45 (10%)	1.32 (18%)	1.21 (25%)
2000	1.75	1.52 (14%)	1.28 (27%)	1.14 (35%)	1.53 (13%)	1.30 (26%)	1.16 (34%)
4000	1.75	1.70 (3%)	1.35 (23%)	1.18 (33%)	1.70 (3%)	1.42 (19%)	1.20 (32%)
8000	1.92	1.91 (1%)	1.67 (13%)	1.29 (33%)	1.67 (13%)	1.74 (10%)	1.33 (31%)

point on the cylinder surface, where early separation can cause wider wakes and increased drag [29]. The policy successfully delays separation, allowing the flow to remain attached for a longer duration and resulting in reduced drag. This delay of the separation is quantified in the bottom left and center plots of Fig. 2. The first plot shows the time-averaged pressure coefficient on the cylinder surface before and after activation of the actuators. Regions of separated flow are characterized by flat pressure profiles, which significantly contribute to drag increases. After the actuators are activated, the regions of constant pressure become visibly smaller, indicating that the flow remains attached for a longer duration. The center plot shows the polar angle of the points of zero vorticity on the cylinder surface over time, which is indicative of flow separation. The found policy effectively moves the separation angle towards the back of the cylinder, reducing the maximum separation angle (at the top part of the cylinder) from approximately 90°to around ±75°. Overall, the results from policy π demonstrate the effectiveness of the implemented drag reduction strategy, as evidenced by the significant reduction in drag, improved wake symmetry, and delayed separation.

B. Effect of actuation velocities

We examine the effect of the actuation strength and of introducing a regularizer preferring low actuation to the reward function. The actuation velocity effectively influences the amount of energy injected into the system. To quantify this added energy, we introduce the energy cost metric, denoted as E_c. It is defined as follows:

$$E_c = \frac{1}{(\tau_2 - \tau_1)N_a} \int_{\tau_1}^{\tau_2} \sqrt{\sum_{j=0}^{N_a-1} \left(a_j^{t_i}\right)^2} \mathrm{d}t \tag{8}$$

where $\tau_1 = 200T$ corresponds to the start of actuator activation and $\tau_2 = 300T$ to the end time of each of our simulations. We also define $E_c = 0$ for simulations with inactive actuators. This metric provides a measure of the overall cost associated with reducing drag by activating the actuators within a given time interval. Figure 3 shows the mean drag coefficient, mean squared lift coefficient, and mean separation angle as functions of the energy cost. Notably, as the energy cost increases, we observe more substantial reductions in drag. These reductions are achieved by generating narrower and more symmetric wakes. Consequently, the mean squared lift coefficient and mean separation angle decrease with increasing energy cost. Note that an energy cost of zero corresponds to the baseline case before the actuators were activated. As anticipated, smaller actuation velocities result in less drag reduction, refer also to Table I. It is worth noting that the inclusion of a regularizer in the reward function yields comparable drag reductions to the case without a regularizer. However, a significant difference lies in the energy cost, particularly when higher actuation velocities ($c = 0.10$ or $c = 0.15$) are permitted. In both instances, the energy cost is approximately 25% lower.

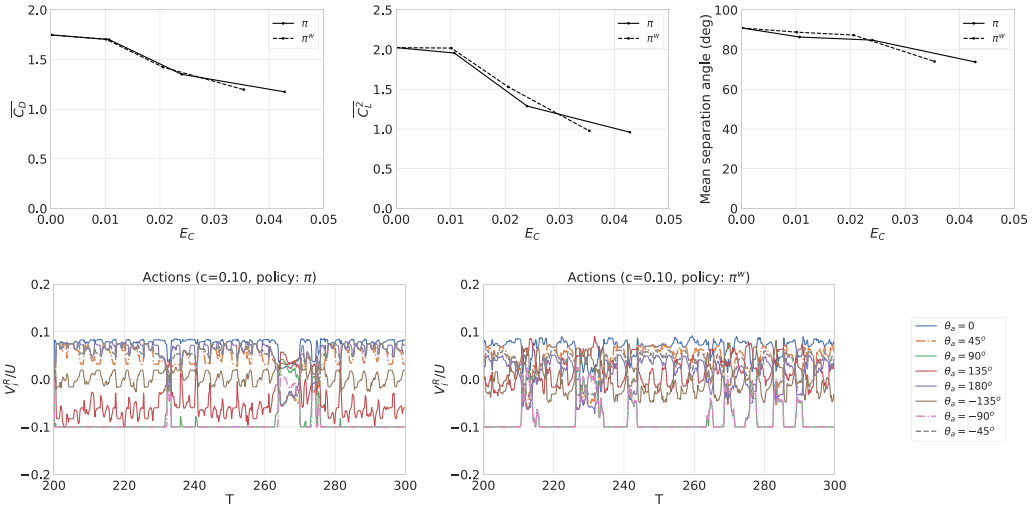

FIG. 3. Summary of results when a regularizer is introduced into the reward function for Re = 4000. Top row: Mean drag coefficient, mean squared lift coefficient, and mean flow separation angle as functions of the energy cost for policy π (no regularizer) and policy π^w (with regularizer). Bottom: Comparison of actions taken by the two policies, expressed as a fraction of the cylinder velocity, for $c = 0.10$.

To understand how varying the maximum actuation velocity affects our policy, we plot the drag coefficient for Re = 4000 and different values of c in Fig. 4. We see that for $c = 0.05$, drag is not reduced until $t \approx 280T$. The vorticity field for $c = 0.05$ is presented in Fig. 4 for various time instances. Like the stronger actuation case, the wake width eventually becomes narrower here. However, this does not happen until $t \approx 280T$, which coincides with the time instance during which drag is actually reduced. It is only after that time that the separation angle becomes acute, yielding a narrower wake and a decrease in drag of about 8%. When averaged throughout the whole simulation, the final decrease ends up being only 2.7%. Interestingly, for $c = 0.05$, the actions taken by the RL agent exhibit greater variance over time compared to when $c = 0.15$, as depicted in both Figs. 4 and 2. This variation can be attributed to the RL agent's efforts to control the wake of the cylinder,

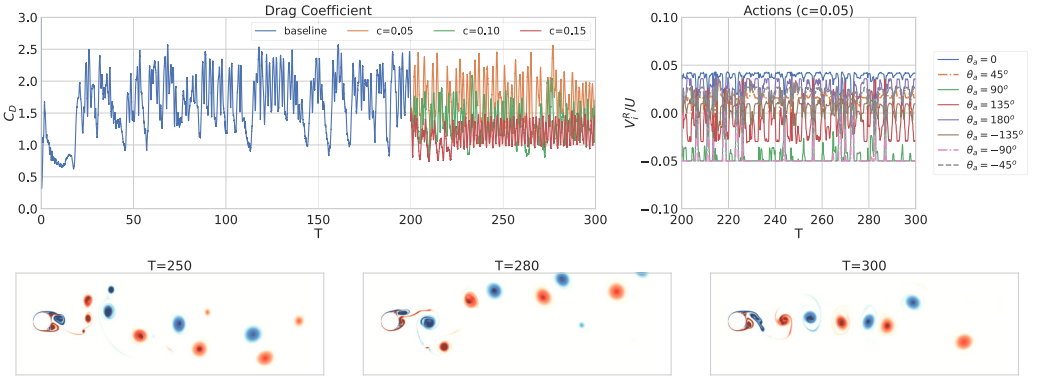

FIG. 4. Summary of results for policy π with varying maximum actuation velocities (varying c) and a fixed Reynolds number of 4000. Top left: Drag coefficient time series for different values of c. Top right: Actions time series for $c = 0.05$, expressed as fraction of cylinder velocity. Bottom: vorticity field at several time instances, for $c = 0.05$.

aiming to make it narrower and more symmetrical. For instance, we observe that the actuator located at $\theta_a = -90^o$ is occasionally turned off (the actuation strength briefly reaches zero, as indicated by the pink curve in Fig. 4). This typically occurs when vortices are shed in the positive y direction, such as at $t \approx 280T$. A similar behavior is observed when vortices are shed in the negative y direction, concerning the actuator located at $\theta_a = 90^o$. Switching off or momentarily reducing the strength of these two actuators appears to be crucial in controlling the direction of vortices shed by the cylinder. Ultimately, successful control leads to a narrower wake and reduced drag.

At the bottom of Fig. 4, we visualize the impact of introducing a regularizer on the actions taken by the RL agent for $c = 0.1$. On the left, we display the actions taken by policy π, while on the right we present the actions taken by π^w. Notably, we observe intermittent deactivation of the actuators positioned at $\theta_a = \pm 90^o$. As discussed earlier, strategic deactivation of these influential actuators at opportune moments enhances wake symmetry. In addition, the transition from π to π^w gives the RL agent the ability to identify safe time instances for deactivating these actuators without compromising drag reduction performance. Furthermore, when employing π^w, the actuator located at $\theta_a = 135^o$ exhibits periodic oscillations around a small value, effectively reducing the energy cost. Conversely, when utilizing π, this particular actuator maintains a nonzero mean value without necessarily contributing to drag reduction significantly.

C. Effect of Reynolds number

Here we assess the effectiveness of our policy at varying Reynolds numbers. During training, we used Re = 1000, 2000, 4000 and we extend this range by also testing Re = 500 and Re = 8000.

For $c = 0.15$, we observe a significant reduction in drag across all Reynolds numbers, as depicted in Fig. 5. Similar to the case of Re = 4000, the wake becomes narrower and flow separation is delayed, as evident by the comparison between the initial and final vorticity field for Re = 8000 in the middle of Fig. 5. The location of separation points and pressure coefficient profiles also show similar trends; the middle right of Fig. 5 shows how the mean separation point location varies with the Reynolds number for the uncontrolled scenario and when policy π is applied for $c = 0.15$. The displacement of the separation point is greater for the larger Reynolds numbers examined. For Re = 1000, 2000, 4000, and 8000, the mechanism for drag reduction is mostly the same for $c = 0.15$. To elucidate the situation at lower Reynolds numbers, we examine the time-averaged pressure coefficient in the bottom right plot of Fig. 5. Although the size of constant pressure regions does not change significantly, the pressure values do, leading to the reduction in total drag. These subtle changes are not clearly visible in the vorticity field before and after actuation, which is shown at the bottom left and center of Fig. 5.

As the Reynolds number increases, the actions taken by the RL agent demonstrate greater variance with time. This is evident as we compare the top left plot of Fig. 5 with the top left plot of Fig. 2. We have established that a strategic deactivation of the actuators placed at $\theta_a = \pm 90^o$ can help with enforcing symmetry in the cylinder's wake. So far, this has only been necessary for smaller values of the actuation velocity ($c = 0.05$ or $c = 0.10$). This is not the case for Re = 8000, where we see that large actuation values are not sufficient and need to be combined with the aforementioned periodic deactivation of the actuators to manage to reduce drag and maintain more symmetric conditions in the wake compared to the uncontrolled case.

When transitioning to the comparatively conservative scenario (with regard to action/actuator velocity magnitudes) characterized by $c = 0.05$, our policy demonstrates notable success in mitigating drag, specifically for Re = 500, 1000, and 2000 (refer to Table I). In the case of Re = 4000, it has been demonstrated that control eventually proves effective, resulting in an approximate 8% reduction in drag. However, it is worth noting that the average reduction amounts to around 2% due to the time required for the actions taken to manifest their full impact. At Re = 8000, allowing actuation with only up to 5% of the cylinder's velocity does not seem to suffice to achieve a notable drag reduction. The introduction of a regularizer does not seem to alter results significantly for $c = 0.05$. This is valid across all Reynolds numbers, with the exception of Re = 8000 where a 13%

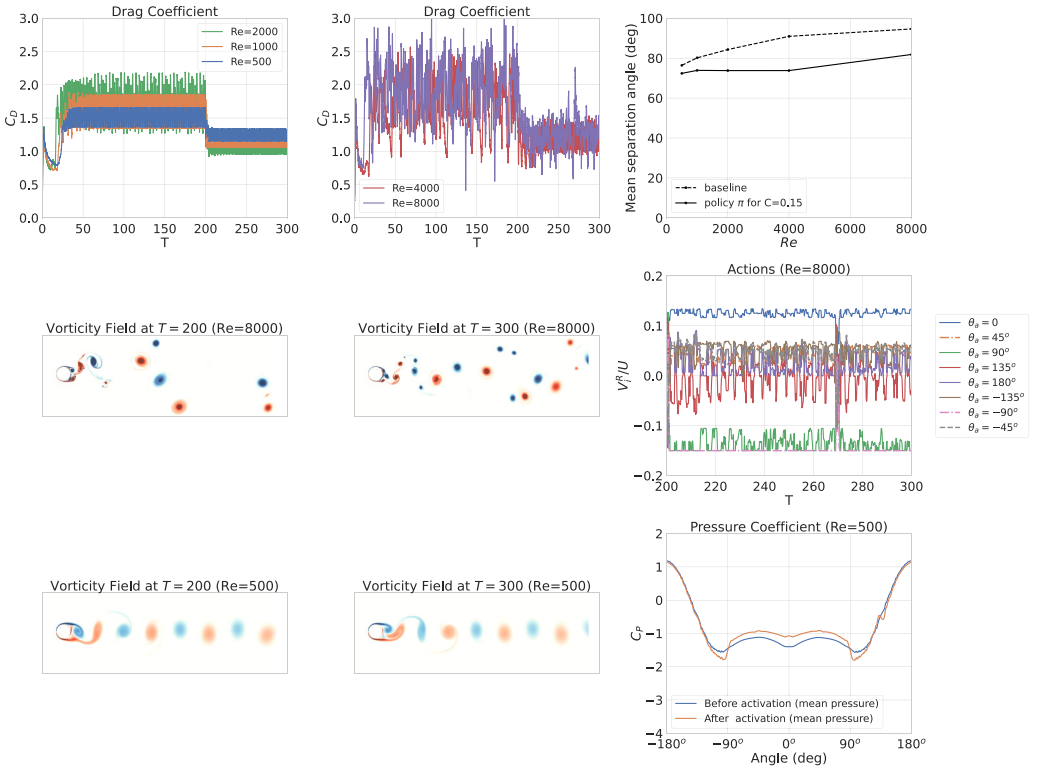

FIG. 5. Summary of results for different Reynolds numbers and $c = 0.15$. Top left and top center: Drag coefficient time series. Top right: Mean separation angle as a function of the Reynolds number for the uncontrolled case and a controlled case. Middle left and center: Vorticity field at Re $= 8000$ before ($T = 200$) and after actuation ($T = 300$). Middle right: Actions as fraction of cylinder velocity for Re=8000. Bottom left and center: Vorticity field at Re $= 500$ before ($T = 200$) and after actuation ($T = 300$). Bottom right: Cylinder pressure coefficient for Re $= 500$ before and after actuators are activated.

reduction is observed. A possible reason for this outlier could be the chaotic nature of the flow at this Reynolds number, which can change drastically from small perturbations; it is, however, hard to pinpoint the exact reasons for this success.

Note that when testing at Reynolds numbers different from the ones used during training, we have chosen to use the same number of actuators. The introduction of a regularizer in one of the two computed policies should result in weak actuation for actuators that do not contribute to drag reduction significantly, indicating that they could possibly be omitted (which proves to be the case for the actuators located at $\pm 135^o$, as discussed in the next section). Furthermore, the location of the pressure and vorticity measurements is kept the same. Our uniform distribution of 16 sensors on the cylinder surface could in theory be altered when changing Reynolds numbers to accommodate maximum information gain and result in an optimal sensor placement, which is beyond the scope of this paper.

D. Discussion: Action space

RL policies can be effective, but at the same time they are complex and do not readily render themselves to interpretation. Here, to enhance our understanding of the discovered policies, we examine their action space. For each Reynolds number, value of c, and computed policy (π and

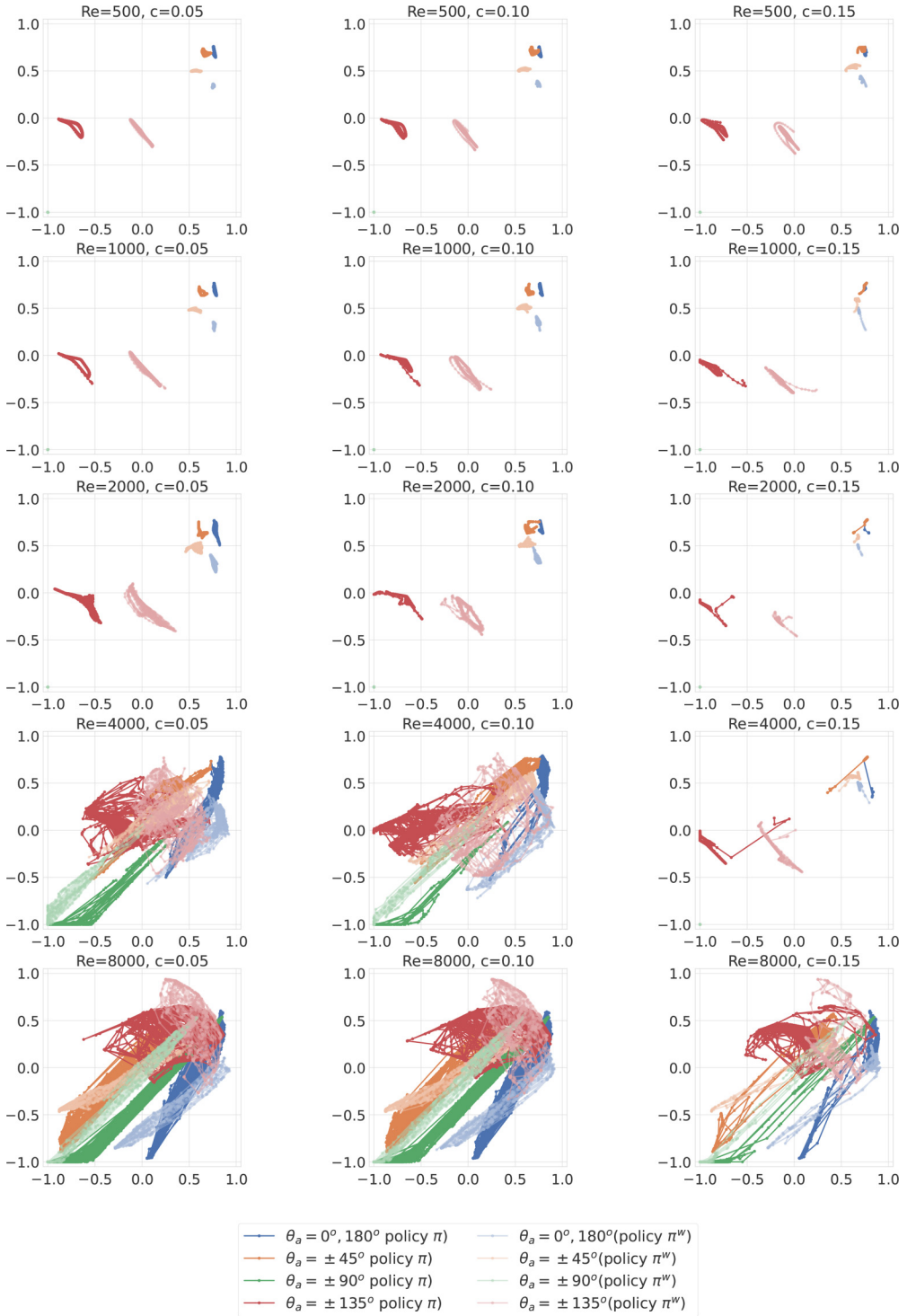

FIG. 6. Action pairs across Reynolds numbers and maximum actuation velocities for the two policies π and π^w. For each plot, the horizontal axis corresponds to $\theta_a \in \{0^o, 45^o, 90^o, 135^o\}$ and the vertical axis to $\theta_a \in \{180^o, -45^o, -90^o, -135^o\}$.

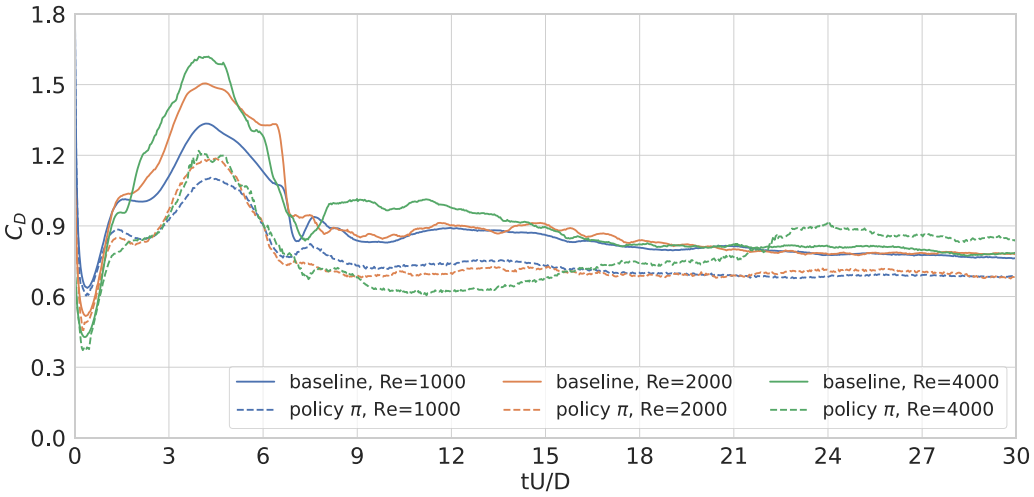

FIG. 7. Flow past three-dimensional cylinder at Re = 1000, 2000, and 4000. Drag coefficient time series for baseline case and for three different policies.

π^w), we plot four pairs of actions; each pair corresponds to all actions taken by two actuators plotted against one another. This is shown in Fig. 6.

At low Reynolds numbers, we observe mostly constant actions. However, as the Reynolds number increases, more complex behavior emerges. At higher Reynolds numbers, we even observe actuator pairs that switch from blowing to suction.

A similar trend can be observed as the maximum actuation velocity (c) decreases. Larger values of c demonstrate a more simplistic approach to drag reduction, where each actuator pair predominantly performs either suction or blowing, without significant variance. On the other hand, reducing the actuation velocity calls for a more sophisticated approach, resulting in increased variance in the actions taken.

Figure 6 also demonstrates how introducing a regularizer affects the policy and the action space. In most cases, the trajectories plotted are shifted closer to the origin, indicating a reduction in the action magnitudes for actuators that do not contribute significantly to drag reduction, such as those placed at $\theta_a = \pm 135^\circ$. Conversely, the other pairs of actuators, whose actuation velocity magnitude is not significantly reduced by the regularizer, display a strong correlation. This correlation is expected due to symmetry; there is no apparent reason for our policy to prefer one direction over another, especially for the key locations at $\theta_a = \pm 45^\circ$ and $\theta_a = \pm 90^\circ$, which strongly influence flow separation.

E. Three-dimensional flow

The transition from two-dimensional to three-dimensional vortex shedding has been observed at Reynolds numbers as low as 200 [38]. Yet, two-dimensional simulations could provide insights relevant to their three-dimensional counterparts at a greatly reduced computational cost; each three-dimensional simulation utilizes about half of the computational resources required to perform a complete training for a single policy (with thousands of two-dimensional simulations).

Hence, direct training of a three-dimensional model at the range of Reynolds numbers considered in this paper is currently not feasible. Nevertheless, we can assess the performance of our policies in a three-dimensional setting. To this end, we apply our computed policy π directly to the three-dimensional flow past a cylinder at Re = 1000, 2000, and 4000 for a maximum actuation velocity up to 15% of the cylinder velocity ($c = 0.15$).

TABLE II. Summary of the three-dimensional controlled and uncontrolled cases.

Re	C_D	C_D, policy π (% reduction)
1000	0.882	0.757 (14.1%)
2000	0.916	0.750 (18.2%)
4000	0.940	0.788 (16.1%)

To implement this, we define a state and a set of actions. Similar to the two-dimensional model, we place $N_a = 8$ mass transpiration actuators on the cylinder surface, but in this case, the actuators are extended in the z direction on the cylinder surface for $|z| < 0.5L$. Our choice of state involves sampling pressure and the z component of vorticity at 16 locations on the cylinder surface, with the quantities averaged in the z direction. Additionally, we include the cylinder lift and drag coefficients in our state, as well as the Reynolds number and maximum actuation velocity (c). The actuator activation time is set to $T = 0.1$ (instead of $T = 200$), and the simulations are terminated at $T = 30$ (instead of $T = 300$). The setup is illustrated in Fig. 1. The resulting drag coefficient for all three cases is plotted in Fig. 7 and the results are summarized in Table II. The policy successfully

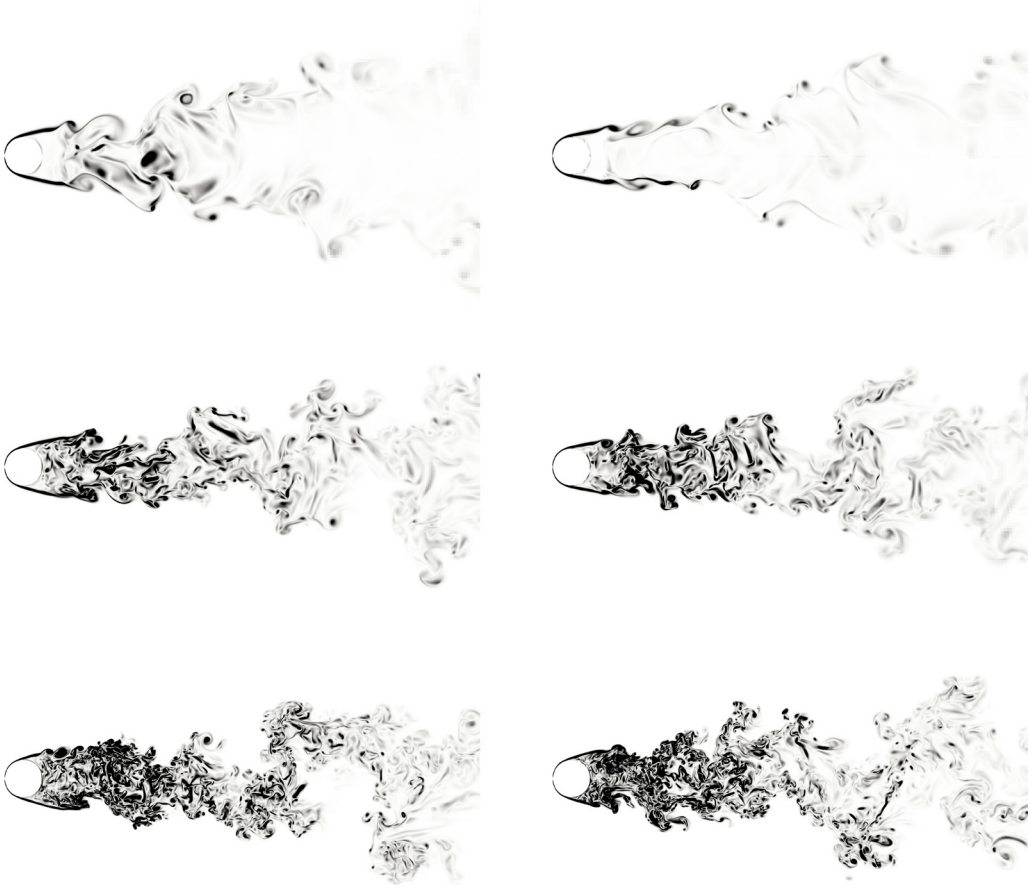

FIG. 8. Left column: Slice through the xy plane showing vorticity magnitude at $T = 30$ for the uncontrolled cases (top to bottom: Re = 1000, 2000, 4000). Right column: Slice through the xy plane showing vorticity magnitude at $T = 30$ for the controlled cases (top to bottom: Re = 1000, 2000, 4000).

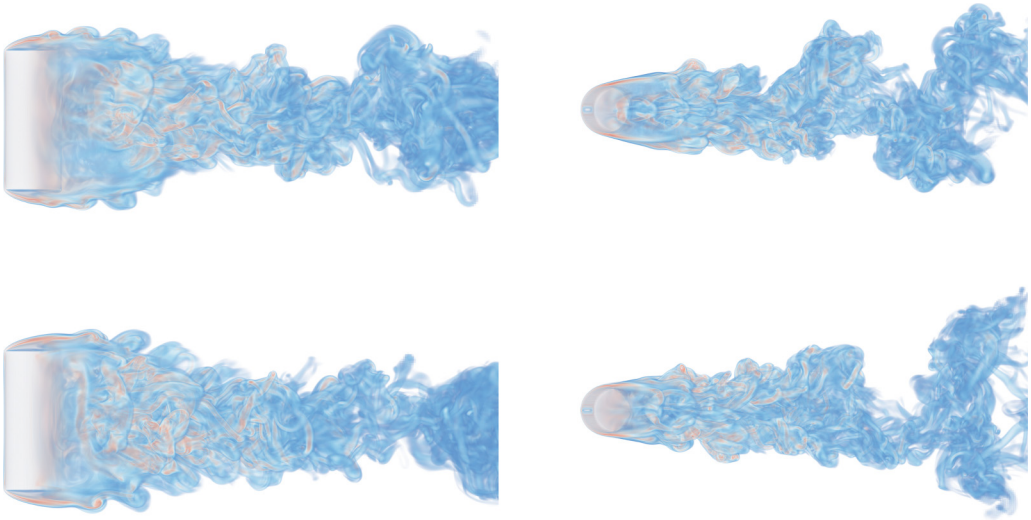

FIG. 9. Vorticity magnitude volume rendering for Re = 2000; left column shows the top views and right column shows the side views. The uncontrolled case corresponds to the first row, while the second row shows the controlled case.

reduces drag, on average, for all three cases. However, the drag we obtain by applying the computed policy is not strictly lower than the baseline case for Re = 4000. At about $T = 22$, drag starts to increase. Upon closer examination of the actions taken by our policy, it becomes apparent that the actuation velocities are significantly reduced during this time period. Consequently, the reduction in flow control strength leads to the observed increase in drag. This discrepancy can be attributed to the divergence between the observed state by our RL agent and its two-dimensional counterpart, particularly at this Reynolds number. Our policy seems to not be trained well in this regime and would require further training with three-dimensional simulations to further improve. Still, many observations that were valid in the two-dimensional regime also hold here. As can be seen in Fig. 8, the wake for the controlled cases at Re = 1000 and 2000 is narrower and separation is delayed. The same is true for Re = 4000, but only before $T = 22$. Finally, a volume rendering of the vorticity magnitude at $T = 30$ for Re = 2000 is shown in Fig. 9 for both the uncontrolled and controlled cases.

IV. CONCLUSIONS

We investigate drag reduction for 2D and 3D cylinder flows through active control mechanisms discovered by deep RL. The study reveals an intriguing trade-off between drag reduction and actuation energy expenditure. Aggressive actuation leads to significant delays in separation on the cylinder surface and up to 35% drag reduction. On the other hand, a more conservative approach in terms of energy expenditure identifies instances when the actuators can be turned off. Similarly, when limited resources are available (in terms of maximum actuation velocity), the discovered policies manipulate the flow field and eventually reduce drag by turning on and off, at select time instances, the actuators.

We find that the identified control policies exhibit generalization to a wider range of Reynolds numbers than the ones used during training. Notably, despite being trained for two-dimensional planar flows, the computed policy is effective for three-dimensional flows as well. We also describe efforts to interpret the complex policies developed by RL. At the same time, we note that RL requires length evaluations and there is significant room for improvement by developing effective

surrogate models. The exhibited generalization from 2D to 3D flows is intriguing and deserves further examination. We argue that further work in interpretable RL is required.

The present paper identifies effective and interpretable control strategies while promoting efficient resource utilization. We believe that the proposed RL strategies for flow control can be extended to a broader range of unsteady separated flows, providing insights into the drag reduction mechanisms under energy and other constraints.

[1] J. Wang and L. Feng, Introduction, in *Flow Control Techniques and Applications*, Cambridge Aerospace Series (Cambridge University Press, Cambridge, 2018), pp. 1–22.

[2] L. K. Doreti and L. Dineshkumar, Control techniques in flow past a cylinder—a review, IOP Conf. Ser.: Mater. Sci. Eng. **377**, 012144 (2018).

[3] L. Sirovich and S. Karlsson, Turbulent drag reduction by passive mechanisms, Nature (London) **388**, 753 (1997).

[4] Y. Ran, Z. Deng, H. Yu, W. Chen, and D. Gao, Review of passive control of flow past a circular cylinder, J. Visualization **26**, 1 (2023).

[5] M. Milano and P. Koumoutsakos, A clustering genetic algorithm for cylinder drag optimization, J. Comput. Phys. **175**, 79 (2002).

[6] R. Sosa, J. D'Adamo, and G. Artana, Circular cylinder drag reduction by three-electrode plasma actuators, J. Phys.: Conf. Ser. **166**, 012015 (2009).

[7] L. Wang, Md. M. Alam, and Y. Zhou, Drag reduction of circular cylinder using linear and sawtooth plasma actuators, Phys. Fluids **33**, 124105 (2021).

[8] W.-L. Chen, Y. Huang, C. Chen, H. Yu, and D. Gao, Review of active control of circular cylinder flow, Ocean Eng. **258**, 111840 (2022).

[9] I. Guinness and T. Persoons, Passive flow control for drag reduction on a cylinder in cross-flow using leeward partial porous coatings, Fluids **6**, 289 (2021).

[10] X. Mao and B. Wang, Spanwise localized control for drag reduction in flow passing a cylinder, J. Fluid Mech. **915**, A112 (2021).

[11] M. Gazzola, B. Hejazialhosseini, and P. Koumoutsakos, Reinforcement learning and wavelet adapted vortex methods for simulations of self-propelled swimmers, SIAM J. Sci. Comput. **36**, B622 (2014).

[12] M. Gazzola, A. A. Tchieu, D. Alexeev, A. de Brauer, and P. Koumoutsakos, Learning to school in the presence of hydrodynamic interactions, J. Fluid Mech. **789**, 726 (2016).

[13] G. Novati, S. Verma, D. Alexeev, D. Rossinelli, W. van Rees, and P. Koumoutsakos, Synchronisation through learning for two self-propelled swimmers, Bioinspir. Biomim. **12**, 036001 (2017).

[14] S. Verma, G. Novati, and P. Koumoutsakos, Efficient collective swimming by harnessing vortices through deep reinforcement learning, Proc. Natl. Acad. Sci. USA **115**, 5849 (2018).

[15] I. Mandralis, P. Weber, G. Novati, and P. Koumoutsakos, Learning swimming escape patterns for larval fish under energy constraints, Phys. Rev. Fluids **6**, 093101 (2021).

[16] P. Gunnarson, I. Mandralis, G. Novati, P. Koumoutsakos, and J. Dabiri, Learning efficient navigation in vortical flow fields, Nat. Commun. **12**, 7143 (2021).

[17] G. Novati, H. L. de Laroussilhe, and P. Koumoutsakos, Automating turbulence modelling by multi-agent reinforcement learning, Nat. Mach. Intell. **3**, 87 (2021).

[18] H. J. Bae and P. Koumoutsakos, Scientific multi-agent reinforcement learning for wall-models of turbulent flows, Nat. Commun. **13**, 1443 (2022).

[19] D. Zhou, M. P. Whitmore, K. P. Griffin, and H. J. Bae, Large-Eddy simulation of flow over Boeing Gaussian bump using multi-agent reinforcement learning wall model, in *AIAA AVIATION 2023 Forum* (AIAA, Reston, VA, 2023), https://arc.aiaa.org/doi/pdf/10.2514/6.2023-3985.

[20] M. Tokarev, E. Palkin, and R. Mullyadzhanov, Deep reinforcement learning control of cylinder flow using rotary oscillations at low Reynolds number, Energies **13**, 5920 (2020).

[21] J. Rabault, M. Kuchta, A. Jensen, U. Ré glade, and N. Cerardi, Artificial neural networks trained through deep reinforcement learning discover control strategies for active flow control, J. Fluid Mech. **865**, 281 (2019).
[22] H. Tang, J. Rabault, A. Kuhnle, Y. Wang, and T. Wang, Robust active flow control over a range of Reynolds numbers using an artificial neural network trained through deep reinforcement learning, Phys. Fluids **32**, 053605 (2020).
[23] J. Rabault, F. Ren, W. Zhang, H. Tang, and H. Xu, Deep reinforcement learning in fluid mechanics: A promising method for both active flow control and shape optimization, J. Hydrodyn. **32**, 234 (2020).
[24] B. Wang, Q. Wang, Q. Zhou, and Y. Liu, Active control of flow past an elliptic cylinder using an artificial neural network trained by deep reinforcement learning, Appl. Math. Mech. **43**, 1921 (2022).
[25] P. Varela, P. Suárez, F. Alcántara-Ávila, A. Miró, J. Rabault, B. Font, L. M. García-Cuevas, O. Lehmkuhl, and R. Vinuesa, Deep reinforcement learning for flow control exploits different physics for increasing Reynolds number regimes, Actuators **11**, 359 (2022).
[26] Z. Wang, D. Fan, X. Jiang, M. S. Triantafyllou, and G. E. Karniadakis, Deep reinforcement transfer learning of active control for bluff body flows at high Reynolds number, J. Fluid Mech. **973**, A32 (2023).
[27] T. Sonoda, Z. Liu, T. Itoh, and Y. Hasegawa, Reinforcement learning of control strategies for reducing skin friction drag in a fully developed turbulent channel flow, J. Fluid Mech. **960**, A30 (2023).
[28] D. Fan, L. Yang, Z. Wang, M. S. Triantafyllou, and G. E. Karniadakis, Reinforcement learning for bluff body active flow control in experiments and simulations, Proc. Natl. Acad. Sci. USA **117**, 26091 (2020).
[29] M. Chatzimanolakis, P. Weber, and P. Koumoutsakos, Vortex separation cascades in simulations of the planar flow past an impulsively started cylinder up to Re = 100000, J. Fluid Mech. **953**, R2 (2022).
[30] G. Novati and P. Koumoutsakos, Remember and forget for experience replay, in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, USA*, Proceedings of Machine Learning Research, Vol. 97, edited by K. Chaudhuri and R. Salakhutdinov (PMLR, Long Beach, California, 2019), pp. 4851–4860.
[31] S. M. Martin, D. WÃ¤lchli, G. Arampatzis, A. E. Economides, P. Karnakov, and P. Koumoutsakos, Korali: Efficient and scalable software framework for Bayesian uncertainty quantification and stochastic optimization, Comput. Methods Appl. Mech. Eng. **389**, 114264 (2022).
[32] P. Angot, C. H. Bruneau, and P. Fabrie, A penalization method to take into account obstacles in incompressible viscous flows, Numer. Math. **81**, 497 (1999).
[33] Y. Ueda and T. Kida, Asymptotic analysis of initial flow around an impulsively started circular cylinder using a Brinkman penalization method, J. Fluid Mech. **929**, A31 (2021).
[34] D. Rossinelli, B. Hejazialhosseini, W. van Rees, M. Gazzola, M. Bergdorf, and P. Koumoutsakos, MRAG-I2D: Multi-resolution adapted grids for remeshed vortex methods on multicore architectures, J. Comput. Phys. **288**, 1 (2015).
[35] D. Rossinelli, B. Hejazialhosseini, P. Hadjidoukas, C. Bekas, A. Curioni, A. Bertsch, S. Futral, S. J. Schmidt, N. A. Adams, and P. Koumoutsakos, 11 PFLOP/s simulations of cloud cavitation collapse, in *International Conference for High Performance Computing, Networking, Storage and Analysis, SC* (Association for Computing Machinery, Denver, Colorado, 2013).
[36] M. Chatzimanolakis, P. Weber, F. Wermelinger and P. Koumoutsakos, Cubism—a C++ library for distributed block-structured adaptive mesh refinement, arXiv:2206.07345.
[37] P. Weber, D. Wälchli, M. Zeqiri, and P. Koumoutsakos, Remember and forget experience replay for multi-agent reinforcement learning, arXiv:2203.13319.
[38] N. Kanaris, D. Grigoriadis, and S. Kassinos, Three dimensional flow around a circular cylinder confined in a plane channel, Phys. Fluids **23**, 064106 (2011).