Unsteady cylinder wakes from arbitrary bodies with differentiable physics-assisted neural network

Shuvayan Brahmachary ⊙* and Nils Thuerey ⊙

Department of Mathematics and Informatics, Technical University of Munich Boltzmannstrasse 3, 85748 Garching, Germany

This work describes a hybrid predictive framework configured as a coarse-grained surrogate for reconstructing unsteady fluid flows around multiple cylinders of diverse configurations. The presence of cylinders of arbitrary nature causes abrupt changes in the local flow profile while globally exhibiting a wide spectrum of dynamical wakes fluctuating in either a periodic or chaotic manner. Consequently, the focal point of the present study is to establish predictive frameworks that accurately reconstruct the overall fluid velocity flowfield such that the local boundary layer profile, as well as the wake dynamics, are both preserved for long time horizons. The hybrid framework is realized using a base differentiable flow solver combined with a neural network, yielding a differentiable physics-assisted neural network (DPNN). The framework is trained using bodies with arbitrary shapes, and then it is tested and further assessed on out-of-distribution samples. Our results indicate that the neural network acts as a forcing function to correct the local boundary layer profile while also remarkably improving the dissipative nature of the flowfields. It is found that the DPNN framework clearly outperforms the supervised learning approach while respecting the reduced feature space dynamics. The model predictions for arbitrary bodies indicate that the Strouhal number distribution with respect to spacing ratio exhibits similar patterns with existing literature. In addition, our model predictions also enable us to discover similar wake categories for flow past arbitrary bodies. For the chaotic wakes, the present approach predicts the chaotic switch in gap flows up to the mid-time range.

I. INTRODUCTION

Machine learning has drawn significant attention towards the field of fluid simulations [1,2], particularly in the past two decades. Owing to its successful inception in the field of turbulence modeling [3–6], it has spawned multiple avenues such as super-resolution reconstruction of fluid flows [7,8], detection of turbulent interface [9], active flow control [10], fluid-particle interaction [11], reduced-order modeling (ROM) for predicting the dynamical state of a fluid system [12–16], aerodynamic shape optimization [17], among others. To incorporate human knowledge in the form of physics models, methods were introduced that integrate them into the training procedure, e.g., physics-informed neural network [18,19], graph neural network [20,21], and generative adversarial network [22] with physical loss functions [23]. While custom loss functions satisfy the boundary conditions and minimize the residual of the underlying governing partial differential equations, they only allow for partial coupling of the flow solver with the neural network. Another alternative that allows full integration of the flow solver within the training loop is via differentiable models [24], e.g., differentiable physics for turbulence modeling [25]. This approach has demonstrated its effectiveness as a cost-efficient strategy for turbulence modeling compared to the cost-intensive direct numerical simulations. Such approaches have also shown the potential to serve as a coarse-grained surrogate for unsteady fluid flows. For instance, the works of Refs. [26,27] demonstrated that differentiable flow solvers coupled with a neural network can yield satisfactory reconstruction of flowfields for representative test cases. However, the full merits of the hybrid strategy as a generic and accurate approach are far from established.

A primary task when evaluating the merits of a predictive framework is to identify a fluid test problem that is of significant practical interest and complexity. One such problem that has been of significant interest to the fluid dynamics community is that of flow past body [28] or multiple bodies [29]. Typical factors and challenges that plague the analysis of such fluid flows may include computational time or the turn-around time, low to large temporal scales in the flow, unsteadiness in the flow, the presence of multiple bodies, a large spectrum of wake dynamics, etc. While coarse-grained surrogates reduce the computational burden [30], the effectiveness of such a strategy to reliably reconstruct the flowfields for multiple wake regimes, besides ensuring accuracy in local boundary profiles and global wake dynamics, is yet to be examined. For instance, authors of Ref. [31] examine flow past two side-by-side cylinders to qualitatively evaluate the generalizability aspect of their models without considering the local fluid behavior around the body boundary. In addition, the work by Ref. [28] examined flow past a single bluff-body of various shapes using a convolutional neural network-based auto-encoder (CNN-AE) model with an emphasis on global fluid dynamics. In a similar vein, the investigation by Ref. [23] points out that their model predictions for flow past a single cylinder exhibited the largest errors in either the body boundary or the wake region. Clearly, this presents a significant limitation that should be addressed by advancing performance measuring metrics based on both the local as well as global fluid properties.

Flow past objects are of significant interest to the engineering community from the hydrodynamics as well as structural

---

*s.brahmachary@tum.de

design point of view. The potential implications of the underlying instabilities [32], wake dynamics [33], wake-induced vibration [34], and its control [35] have led to establishing a significant body of research. One of the focal points concerning this research is understanding the spectrum of wake flow exhibited by multiple cylinders placed relative to each other. Some of the investigations that present a comprehensive study for two cylinders are Refs. [36–38], etc., and Refs. [39–42] for three or more cylinders. These investigations point towards the direct dependency of the resulting wake regimes with the spacing ratio (i.e., length L to diameter D ratio) L/D. More recently, the investigation by Ref. [43] showed the presence of up to nine distinct flow regimes for cylinders arranged in an equilateral-triangle position at various spacing ratios, $1 \leqslant L/D \leqslant 6$ for multiple Reynolds numbers $50 \leqslant Re \leqslant 175$. While these investigations provide fundamental insights about wake flow regimes and flow states, their analysis is limited to the use of equidiameter circular cylinders or rectangular cylinders with varying aspect ratios. Consequently, this begs the question if the observations related to the universal nondimensional parameters such as Strouhal number and critical spacing ratio (as reported by Ref. [43]) still hold for arbitrarily shaped cylinders.

Analyzing large datasets of wake flow exhibited by arbitrarily shaped complex configurations offers potential for novel physical insights; however, employing traditional body-conforming continuum computational fluid dynamics (CFD) flow solvers can be time-consuming and warrants special treatment for catering to changing geometries via the underlying computational grid. While nonconformal immersed boundary (IB) approaches [44,45] allow for the use of a fixed Cartesian mesh, exploring a large design space of possible spacing ratios could significantly ramp up the turn-around time. Consequently, neural network-based coarse-grained surrogate models for unsteady fluid flows provide a cost-effective alternative approach [30].

In this work, we build on the ideas by Ref. [26] and present a hybrid differentiable physics-assisted neural network (or DPNN) framework that acts as a coarse-grained surrogate for accurate unsteady fluid simulations for low Reynolds number flows past multiple bodies. This framework requires a base solver integrated into the neural network (NN) along with ground truth data to train the resulting hybrid framework. We employ a Cartesian grid-based immersed boundary method (i.e., FoamExtend) as the *Reference* solver. In addition, we employ an in-house developed differentiable flow solver (i.e., PhiFlow) as the *Source* solver. For faster training and inference, the *Source* intentionally resorts to a first-order spatial accuracy along with masked stair-step representation of the underlying body boundary. This inherently reduces the computational burden of the base differentiable flow solver. In contrast, however, the *Reference* solver utilizes an accurate local algebraic reconstruction for fluid flow around the body boundary with second order spatial accuracy. Given this stark contrast between the *Reference* and *Source* solvers, besides the highly multimodal nature of the present problem, this naturally poses a very challenging task for NNs, i.e., to act as a forcing function to learn the nontrivial corrections. We comprehensively evaluate the hybrid approach for multiple wake categories while drawing necessary comparisons with various

benchmarks. Moreover, our approach of utilizing arbitrarily shaped bodies in a staggered equilateral triangle position allows for a large data diversity and sheds light on the physics of the wake dynamics resulting from the arbitrary cylindrical configurations.

Our focus is thus twofold. First, we train and test the DPNN framework on a dataset comprising unsteady flow past arbitrary bodies placed in a staggered equilateral triangle position, each at a unique spacing ratio. These tests were performed once the *Reference* solver was thoroughly validated and verified to generate accurate solutions. Second, after establishing the effectiveness of our approach using multiple performance measuring metrics, we evaluate the out-of-distribution generalization capabilities of our framework. These evaluations is based on configurations that align closely with the current state of existing literature. Throughout the study, we strive to understand the physical implications of the solutions derived from the predictive framework, examining both local boundary layer profiles and global wake dynamics. We also highlight the framework's performance in both physical and reduced feature spaces to demonstrate its long-term temporal stability.

II. NUMERICAL SETUP AND VALIDATION

In this section, we illustrate the computational setup underpinning the entire hybrid framework. This is followed by validation test cases undertaken to affirm the correctness of the setup as well as verify the ability to render accurate solutions.

A. Reference solver

The present study employs the open source `FoamExtend` V 4.0 immersed boundary (IB) approach as the *Reference* solver. As mentioned earlier, the IB-based nonconformal approach serves a dual purpose. First, it greatly simplifies the use of a fixed Cartesian mesh for all underlying configurations, irrespective of the geometric complexity. This methodology circumvents the necessity for labor-intensive remeshing for each geometric configuration and the corresponding coordinate transformation from physical to computational space. Second, the use of a fixed Cartesian mesh ensures seamless integration to the deep-learning pipeline wherein one can leverage the benefits of the Eulerian (i.e., fixed cell in space) viewpoint. `FoamExtend` utilizes a discrete forcing approach-based strategy for enforcing the boundary condition for the immersed bodies [see Fig. 1(a)] and has been built on top of the well-established OpenFoam [46] framework. This method uses a higher order interpolation strategy to interpolate the fluid properties at the immersed cells by using the properties in the neighboring fluid cells.

In the present work, we consider the 2D Navier Stokes equations as mentioned below:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}.\nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \nu\nabla^2\mathbf{u}$$

$$\nabla.\mathbf{u} = 0, \tag{1}$$

where $\mathbf{u} = (u, v)$ is the velocity field, ρ is the density, p is the pressure, and ν is the dynamic viscosity. The first equation represents the momentum conservation equations, whereas the second equation represents the continuity
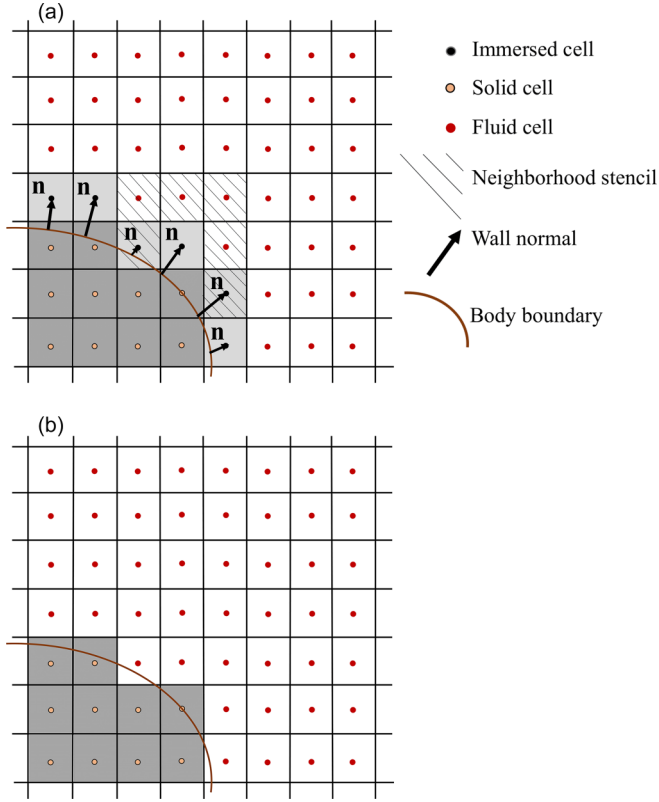
FIG. 1. Boundary reconstruction approach used by (a) `FoamExtend` (*Reference* solver) (b) `Phiflow` (*Source* solver).

equation, which also serves as a constraint on the velocity fields, otherwise known as the "divergence-free" constraint.

The flow scenario describes the incompressible flow past cylinders at low Reynolds number Re $\approx$100. For the present case, the Reynolds number is evaluated based on the cylinder height of the primary upstream cylinder as Re $= \frac{UH}{\nu}$, where U and H represent the freestream velocity and height of the upstream cylinder respectively. For the present setup, $\nu = 0.01$ Ns/m^2 and $U = 1$ m/s are used. A second-order accurate Crank Nicolson scheme is used to march the solutions in time, while a second-order limited linear (central-differencing) scheme is used for convective flux discretization. Pressure-implicit with splitting of operators (PISO) [47] is employed to solve for the discretized momentum and continuity equations in one predictor and two corrector steps. In particular, preconditioned conjugate gradient (PCG)-based solver is used for p linear equations along with a diagonal-based incomplete Cholesky (DIC) precondition for symmetric matrices. In addition, for U linear equations, BiCGSTAB is used along with diagonal-based incomplete LU (DILU) preconditioned for asymmetric matrices.

B. Validation: Strouhal number and force characteristics

We execute a comprehensive grid independence study to corroborate the *Reference* solver used to acquire the ground truth data. We initiate our numerical experiments with flow past a single cylinder at Reynolds number Re $= 100$. A computational domain of size [0,24D] $\times$ [0,16D] is chosen using a uniformly spaced Cartesian grid. In particular, we choose

medium and fine spatial resolutions to undertake a grid-independence study, i.e., $\Delta x/D = \Delta y/D \in \{1/32, 1/50\}$, where D represents the diameter of the cylinder. This leads to a blockage ratio, $B = D/H = 0.0625$. Further, we use a constant time-step $\Delta t = 0.1$ s for this validation test case and perform the simulations until a total time $T = 200$ s.

The time-averaged drag coefficient $\bar{C}_d$ as well as the root-mean-square (r.m.s.) lift coefficient C'_1 obtained from the *Reference* flow solver is compared with existing results. Specifically, Table I shows the $\bar{C}_d$, C'_1, and Strouhal number St values for flow past a single cylinder at various grid resolutions. It is found that both $\bar{C}_d$ and C'_l obtained from the medium as well as the fine spatial resolutions compare really well with the existing solutions of Ref. [48], who also employ a nonconformal immersed boundary-based numerical solver for the computations. It must be noted that a change in the underlying grid from medium to fine spatial resolution results in an increase in the total number of control volumes n_c from 393 216 to 960 000. Consequently, to allow for a reasonable computational time while retaining accuracy, we choose the medium grid for the remainder of the computations in this section as well as for obtaining the ground truth solution.

Table II presents the mean as well as the r.m.s. values for the coefficient of drag and lift for the upstream and downstream cylinder (only one) for flow past three cylinders placed in an equilateral triangle position at spacing ratio $L/D = 2.5$. It is found that the present solutions compare really well with the ones obtained by Ref. [42], who employ a body-conformal finite-volume approach using a time-step size $\Delta t = 0.05$ s. The influence of the time-step size is also analyzed in Table III. It is found that the choice of time-step size results in a marginal difference in force quantities as well as the Strouhal number. Consequently, a moderate value of $\Delta t = 0.1$ s is adopted for the present study. It can now be remarked that the grid resolution and time-step size chosen for the present study allow for the accurate computation of flow past multiple bodies. In addition, we also qualitatively verify the wake regimes produced using the flow solver, as highlighted in Appendix B.

C. Differentiable flow solver

The present study uses the open source simulation toolkit `Phiflow` (ϕ_{flow}) as the base differentiable flow solver (or *Source*). It leverages automatic differentiation (AD) to enable an end-to-end recurrent training paradigm. This incorporation allows for the computation of gradients of the simulation outcomes with respect to its inputs or parameters in an efficient manner. The key distinction here is not just the ability to solve fluid flow governing equations but to do so in a way that is inherently differentiable with respect to the parameters and states involved in the simulation. By leveraging AD, we can directly integrate fluid flow simulations into the training loop of neural network models, enabling them to learn from and adapt to the underlying physics of fluid dynamics. Phiflow in this study serves as the computational foundation that allows us to simulate fluid flow scenarios and directly incorporate these simulations into a recurrent neural network training process. This approach is fundamentally different from traditional CFD methods, which typically focus on solving the

TABLE I. Coefficient of lift C_l, drag C_d, and Strouhal number St for flow past a single cylinder at Re = 100.

Re = 100 (Single cylinder)	$\bar{C}_d$	C_l'	St
Constant *et al.* 2017 [48] ($\Delta x = \Delta y = 0.02D$)	1.38	—	0.165
Constant *et al.* 2017 [48] ($\Delta x = \Delta y = 0.010D$)	1.37	—	0.165
Present ($\Delta x = \Delta y = 0.0312D$)	1.412	0.276	0.156
Present ($\Delta x = \Delta y = 0.02D$)	1.403	0.285	0.156

fluid flow governing equations without providing the capability to compute gradients of the simulation outcomes with respect to its inputs.

While it is crucial that the base solver (i.e., Source) be differentiable, the *Reference* solver (i.e., FoamExtend in this case) need not be differentiable. We employ the projection method to decouple the momentum and continuity equation in a two predictor-corrector step. First, we use the operator splitting of the diffusion, advection, and pressure terms in the predictor phase. Specifically, the MacCormack advection step for advecting the initial velocity fields. Second, in the corrector phase, the spatial gradients of pressure are used to compute the divergence-free velocity field [49].

Finally, the choice of Cartesian grids in both *Reference* and *Source* solver allows for a one-to-one mapping between each control cell. Figure 1 elaborates further on the nuances and differences between the reconstruction strategies employed by *Reference* solver [Fig. 1(a)] and *Source* solver [Fig. 1(b)]. It is clear that the former employs a sharp interface algebraic reconstruction approach locally for the near body boundary cells whereas the later resorts to a stair-step representation of the underlying geometry. This is especially important given that the use of masked stair-step representation of the body boundary for faster training by the *Source* results in a very approximate reconstruction of near-body fluid properties. This limitation allows for us to build a training setup wherein the network learns to faithfully reconstruct the local boundary flow profile based on the accurate *Reference* solver.

D. Cylinder layout

We carefully choose a setup for cylinder cluster arrangement that allows us to mimic the traditional approach adopted in open literature. However, as we seek to generate and exploit a multitude of complexities in the training data to harness the full capabilities of the present hybrid learning strategy, we introduce some randomness to the setup. Some specifications related to the cylinder arrangements are as follows:

(1) *Cylinder arrangement.* We place three cylinders in an equilateral triangle position, i.e., one upstream cylinder followed by two downstream cylinders, placed apart from each other via the spacing ratio L/D. This setup mimics the "regular triangular" cluster setup used in Ref. [36] and has also been investigated recently by Ref. [43]. Additionally, this arrangement allows us to investigate *proximity* interference (P), *wake* interference (W), a combination of (P+W) interference as well as no-interference.

(2) *Upstream composite cylinder.* The upstream cylinder (cylinder C_1) is framed as a composite body consisting of a base rectangle (at its center) and secondary rectangle or semicircle on its sides (see Fig. 2). The selection between a secondary rectangle or a secondary semicircle as one of the sides of the base rectangle is random. We define a function $f_1 : R_{u,i} \to S$, such that it maps a random integer $R_{u,i} \in \{0,1\}$ to geometric shapes, S $\in$ {semicircle, rectangle}, where $i \in \{1,2,3,4\}$. The outcome can be represented as follows:

$$R_{u,i} = \begin{cases} 0 & f_i = \text{semicircle} \\ 1 & f_i = \text{rectangle} \end{cases}$$

The length of the base rectangle, as well as the secondary rectangles, are randomly chosen between fixed upper and lower limits (see Table IV). The diameter of the semicircle is equal to the edge length of the base rectangle to which it is attached.

(3) *Downstream cylinders.* The choice of the downstream cylinders is either a rectangular cylinder of a certain length and height or a circular cylinder of a certain diameter. This selection is partially random. We define a function $f_{2,3} : R_d \to S$, such that it maps a random integer $R_d \in \{0,1\}$ to geometric shapes, S $\in$ {circle, rectangle}. Once the upper downstream cylinder (cylinder C_2) is chosen (say a rectangular cylinder), the lower downstream cylinder (cylinder C_3) gets fixed (a circular cylinder). This outcome can be represented as follows:

$$R_d = \begin{cases} 0 & f_2 = \text{circle}; \ f_3 = \text{rectangle} \\ 1 & f_2 = \text{rectangle}; \ f_3 = \text{circle} \end{cases}.$$

This forces the training dataset to always contain a rectangle and a circular cylinder as the two downstream cylinders. In addition, once the height of the downstream rectangular cylinder, H_{C2} (or the diameter of the downstream circular cylinder, D_{C3}) is chosen, the dimension (i.e., diameter or cylinder height) of the other downstream cylinder is fixed to satisfy the following criterion:

$$H_{C2} + D_{C3} = 2H_{C1}. \tag{2}$$

TABLE II. Coefficient of lift C_l, drag C_d, and Strouhal number St for flow past three cylinders at various time-step sizes.

Re = 100 (Three cylinders, $L/D = 2.5$)	$\bar{C}_{d,1}$	$\bar{C}_{l,1}$	$C_{l,1}'$	$\bar{C}_{d,2}$	$\bar{C}_{l,2}$	$C_{l,2}'$
Zheng *et al.* [42]	1.23	0.0	−0.002	1.53	−0.087	0.335
Present ($\Delta x = \Delta y = 0.0312D$)	1.249	0.0	0.0	1.561	0.166	0.433

TABLE III. Coefficient of lift C_l, drag C_d, and Strouhal number St for flow past three cylinders.

Re = 100 (Three cylinders, $L/D = 2.5$)	$\bar{C}_{d,1}$	$\bar{C}_{l,1}$	$C'_{l,1}$	$\bar{C}_{d,2}$	$\bar{C}_{l,2}$	$C'_{l,2}$	St
Present ($\Delta t = 0.25$)	1.254	0.0	0.0	1.584	0.172	0.49	0.148
Present ($\Delta t = 0.1$)	1.249	0.0	0.0	1.561	0.166	0.433	0.156
Present ($\Delta t = 0.05$)	1.244	0.0	0.0	1.545	0.166	0.359	0.156

This restricts the cylinders from overlapping. The dimensions of the rectangle and the circular cylinder are randomly chosen between a fixed upper and lower limit (see Table IV).

(4) *Spacing ratio*. The distance between center-to-center distance between the downstream cylinders, as well as the downstream and upstream cylinders, is controlled by the spacing ratio L/D. The L/D values are chosen such that it covers a wide spectrum of possible wake flow regimes (i.e., $1.2 \leqslant L/D \leqslant 5.5$). Care is taken to avoid overlapping bodies for low L/D values. Finally, to ensure an unbiased selection of the spacing ratio, the Latin hypercube sampling (LHS) is chosen as a design of the experiment method.

Points 2 and 3 above intentionally introduce arbitrariness to the training data to serve a dual purpose. First, we analyze the role of arbitrariness in the wake flow regime to offer new insight into the present body of knowledge around flow past equidiameter cylinders. Second, this introduces data diversity while also allowing us to evaluate the generalizability of the hybrid learning framework to more representative problems. Further, point 4 allows us to uncover various wake categories that have been introduced recently by Ref. [43] for similar spacing ratios. For the present work, we employ a python script which pregenerates a set of cylinder configurations using a fixed random number seed. The configurations of cylinder generated is done in a manner that always satisfies the bounds mentioned in Table IV. At the end of this step, we have 100 set of cylinder layouts, each available in a .stl format that is used by the Reference flow solver. These serve as the basis to classify the solid cells from the fluid cells. Once the flowfield for all the 100 experiments (based on 100 unique cylinder layouts) are computed, we choose the first 50 as training and the remaining 50 as inference or testing samples.

E. Wake categories

This section introduces the various wake categories obtained from the *Reference* solver at multiple spacing ratios L/D for equidiameter cylinders placed in an equilateral triangle position. These representative spacing ratios for equidiameter cylinders are for illustration purposes and are not part of the training data. We purposely choose

equidiameter circular cylinders so that fair comparisons can be drawn with Ref. [43] who also evaluate it on identical conditions.

Single bluff-body wake. At extremely small spacing ratio, i.e., $L/D = 1.2$, the three cylinders are close to each other such that they exhibit vortices as if it were a "single bluff-body." This is further demonstrated pictorially in Fig. 32 in the Appendix B, along with other representative cases. This kind of wake is periodic in nature.

Deflected gap wake. At slightly larger spacing ratio, i.e., $L/D = 1.5$, the fluid in between the two downstream cylinders (or gap flow) switches direction to either the upper cylinder (i.e., cylinder 2) or the lower cylinder (i.e., cylinder 3), hence the name "deflected gap." This switch is constant with time. The kind of wake is periodic, with some modulation over time.

Flip-flopping wake. For marginally larger spacing ratio, i.e., $L/D = 2.25$, the gap flow in between the two downstream cylinders switches direction erratically between the upper cylinder (i.e., cylinder 2) or the lower cylinder (i.e., cylinder 3), hence the name "flip-flopping." This chaotic switch is a challenging case for any predictive framework, and the kind of wake is irregular in nature.

Antiphase wake. For moderately larger spacing ratio, i.e., $L/D = 3.5$, the vortices shed by the two downstream cylinders rotate in opposing directions, hence the name "antiphase." This wake category is periodic in nature and exhibits a symmetry about the centerline streamwise direction. For arbitrarily shaped bodies, this symmetry is lost and is consequently referred to in this work as "quasi-anti-phase" wake.

Fully developed in-phase wake. For larger spacing ratio, i.e., $L/D = 5$, the vortices shed by the two downstream cylinders are in same direction; hence the name "in-phase." Further, the upstream cylinder also sheds vortices which merges with the vortices shed by the downstream cylinders, hence the term "fully developed." This kind of wake is periodic in nature. For arbitrarily shaped bodies, this is referred to in this work as "quasi-in-phase" wake.

III. HYBRID PREDICTIVE FRAMEWORK

This section introduces the methodology behind the DPNN-based hybrid framework used in the present work.

TABLE IV. Upper and lower limits for the dimensions of the cylinders.

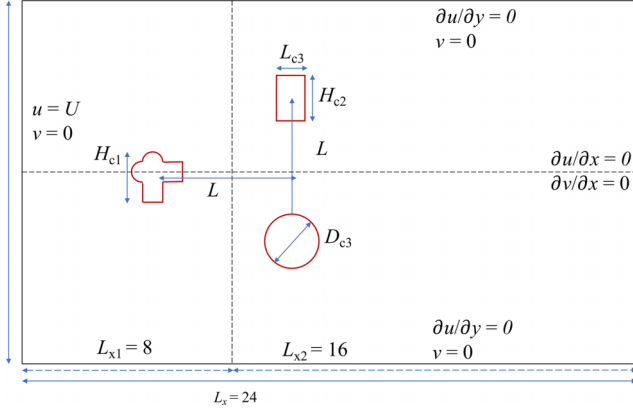| Component		Length (L)		Height (H)		Radius (R)	
		Lower limit	Upper limit	Lower limit	Upper limit	Lower limit	Upper limit
Upstream cylinder	(Primary) Base rectangle	0.40	0.50	0.40	0.50	—	—
	(Secondary) Rectangle	0.15	0.25	0.15	0.25	—	—
Downstream cylinder	Rectangle	0.40	0.50	0.70	1.00	—	—
	Circular cylinder	—	—	—	—	0.40	0.50

FIG. 2. Computational domain including the three bodies placed in an equilateral triangle position (not to scale).

As alluded earlier, such a strategy embeds the base differentiable flow solver `Phiflow` with a neural network (NN) architecture, mimicking the *solver-in-the-loop* strategy previously employed by Ref. [26]. Such a strategy reaps the benefit of long-term training feedback while enabling an end-to-end training paradigm. We build on the above by enabling a robust and efficient coarse-grained surrogate model for unsteady flows for investigating one of the core fluid flow problems, i.e., wake flows caused by arbitrarily shaped objects. This serves a dual purpose, i.e., first, the arbitrarily shaped bodies allow for a rich and diverse set of datasets. Second, while it is well known that the shape of the body is responsible for the resulting wake characteristics, such an issue has not prompted an investigation into a large spectrum of spacing ratios to the best of our knowledge.

A. Neural network architecture

In this section, we highlight the neural network (NN) architectures that were explored in the present study. We explore a range of popular architectures while keeping the total number of parameters in each network architecture within reasonable limits to facilitate a fair comparison.

The network takes as input of channel size = 3, i.e., velocity fields, $\mathbf{u} = u, v$ along with the shape mask or marker field $m_f \in R^{n_x \times n_y}$, where n_x and n_y represents the number of grid points along the longitudinal and lateral directions, respectively. The shape mask m_f uniquely identifies the shape of the enclosed body and is fixed for a given test configuration. While the ground truth data is obtained on a high-resolution grid of size $n_x, n_y = (768 \times 512)$, the network only receives a downsampled ($3\times$) velocity and shape mask as inputs (i.e., 96×64), for faster training. For a training sample, there are a total of $t_f = 3000$ frames or snapshots (of which the first 1500 are discarded to only consider statically stable snapshots), each representing velocity fields generated sequentially using a constant time step of $\Delta t = 0.1$ s (or, total time $t = 300$ s). The entire dataset consists of 100 different experiments each at a unique spacing ratio, resulting in 100 unique shape masks m_f. We use the default "glorot uniform" to initialize the weights and "zeros" for its biases. As mentioned in Sec. II D and Fig. 2, to allow for a fair comparison, we employ the same

cylinder cluster arrangement as also employed by Ref. [43], with the equidiameter cylinders employed by Ref. [43] being replaced by arbitrarily shaped cylinders in the present work. Among the entire dataset, 50 are used for training, and the remaining 50 are used for testing. The training is executed using a batch size of 50 for 50 epochs (see Appendix C1, Fig. 42) using Adam [50] as the optimizer. A variable learning rate α is employed, which is sequentially reduced with epochs. The output from the network is constrained to be of channel size = 2, i.e., the velocity field at the next time step. In this work, we predominantly employ residual neural network (ResNet)-based architecture. Figure 3 shows a typical ResNet based architecture used alongside the *Source* solver as the hybrid approach. This architecture comprises multiple convolution blocks in series with residual connections in between them. The architecture uses rectified linear unit (or ReLU) as the activation function. Appendix C provides the details related to each architecture.

B. Present hybrid framework

The DPNN-based hybrid predictive framework is a *recursive* learning strategy wherein the base flow solver `PhiFlow` (ϕ_{flow}) is coupled with the neural network as shown in Fig. 3. In the present study, the *Reference* solver (i.e., `FoamExtend`) provides the precomputed ground truth data. The hybrid framework receives an initial guess $\mathbf{r}^t$ from the *Reference* solver and feeds it to the base solver (i.e., `PhiFlow`). The base solver then updates the velocity field to the next time-step $\mathbf{u}^*$. At this stage, no learning is involved. The output from `PhiFlow` is then fed to the NN architecture as input, i.e., velocity fields u, v, and shape marker field m_f. The output from the NN, $\mathbf{f}(\mathbf{u}^*; \theta_W)$ acts as the forcing function which is used to correct the output from the base solver `PhiFlow` (θ_w represents the network weights). This step is made recursive for m solver unrollment steps signifying as many forward steps by base solver in time by feeding the cumulative output back to the base solver, i.e., $\mathbf{u}^{t+1} = \mathbf{u}^* + \mathbf{f}(\mathbf{u}^*; \theta_W)$. At the end of the m unrolling steps, the computed or predicted velocity flowfield is compared with the Reference solution to determine the $\mathcal{L}_2$ loss.

We remark that both the *Reference* and the base solvers are subject to identical values for the dynamic viscosity ν, time-step size Δt, and freestream conditions, besides making sure to incorporate the same object under consideration. The critical difference here lies in their respective solution strategies and the techniques to handle body boundary conditions, i.e., while the *Reference* solver is spatially second-order accurate and algebraically reconstructs the fluid properties along the body boundary to locally satisfy the no-slip boundary conditions, the *Source* solver is spatially first-order accurate and resorts to a masked stair-step representation for computational efficiency. This fact serves as the key motivation to learn the global wake dynamics as well as the local boundary representation via the loss formulation, as will be discussed in the next section.

C. Loss formulation

The loss formulation provides the central goal of the learning process. In this study, this means recovering the
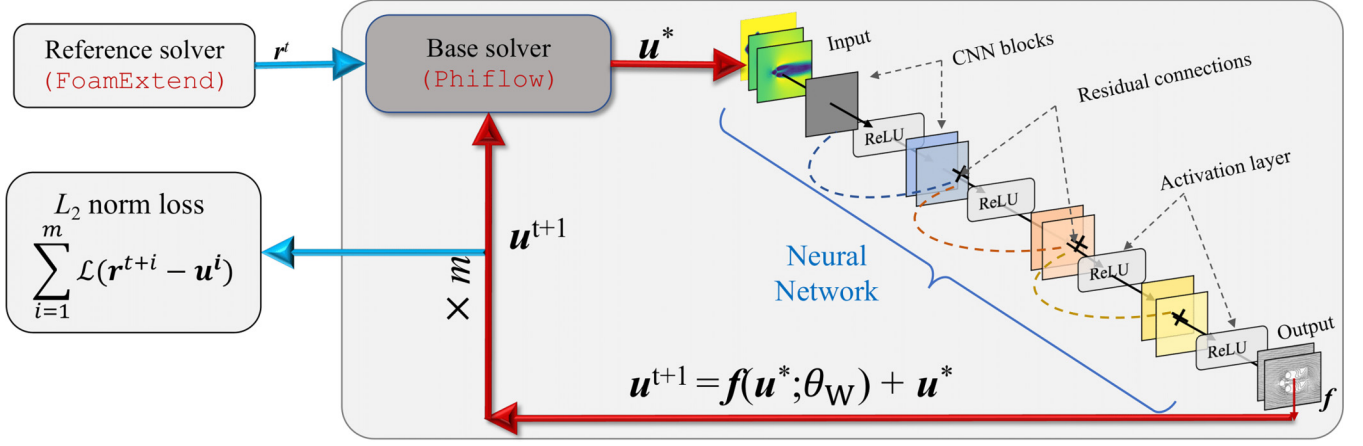
FIG. 3. DPNN-based hybrid predictive framework.

ground truth velocity flowfields as accurately as possible. The trainable parameters of the model are iteratively updated by minimizing the loss function. This allows the optimizer to navigate the nonconvex energy landscape with the goal to arrive at a global/local minima. Typical choices of a loss function $\mathcal{L}$ use some error norm to compare network prediction $\mathbf{u}$ with the ground truth $\mathbf{r}$. For instance, the $\mathcal{L}_2$ norm for a purely supervised training is given as

$$\mathcal{L} = \mathcal{L}_2(\mathbf{r} - \mathbf{u}). \quad (3)$$

Such a formulation minimizes the Euclidean distance between $\mathbf{u}$ and $\mathbf{r}$. Although popular, such a loss is devoid of additional information from the physics point of view. For instance, Ref. [26] showed that one can formulate physics-based loss formulations by taking advantage of the temporal unrolling during the training. This was followed by an independent investigation [25] wherein they built a custom loss function to accurately predict turbulent flows. Consequently, a greater emphasis is placed on rendering a loss yielding a physically consistent solution. Such a physics-based loss could take the form

$$\mathcal{L} = \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}_2(\mathbf{r}^{t+i} - \mathbf{u}^i), \quad (4)$$

where m represents the number of unrollment steps used during training and t represents the instant of time. This loss is strictly not computed at every time step, however, the aggregated history of the loss of every intermediate step (while unrolling for m steps) is summed up to obtain the final $\mathcal{L}$ norm error.

D. Benchmarks

In addition to our present DPNN framework, we also compare it to existing benchmarks. As part of this comparative evaluation, we employ the data-driven supervised learning (SL) method. In addition, we evaluate results produced by only running the base solver, i.e., *Source*. For the SL approach, the network directly receives the input $\mathbf{r}^t$ from the *Reference* solver and produces the output as the flowfield for the subsequent time step, i.e., $\mathbf{f}(\mathbf{r}^t; \theta_W)$. As this SL framework does not contain an integrated *Source* solver, the output generated

by the network serves as the final prediction for the next time step [i.e., $\mathbf{u}^{t+1} = \mathbf{f}(\mathbf{r}^t; \theta_W)$]. This makes the SL approach faster. Notably, to ensure a fair comparison, the outputs from both DPNN and SL frameworks utilize identical underlying networks and receive the same input during training/testing. For the output produced from the *Source* solver, the next time-step predictions can be written as $\mathbf{u}^{t+1} = \mathbf{u}^*$. Finally, For reasons of completeness and validation, we also juxtapose our predictions with those from the *Reference* solver. Comparisons with the *Source* solver highlight the starting point on which the DPNN solver learns to apply its corrections. Further details related to the network architecture may be found in Appendix C.

IV. RESULTS AND DISCUSSIONS

In this section, we present a detailed analysis of the results obtained from the present hybrid learning framework as well as other benchmarks. As mentioned earlier, we employ multiple neural network architectures within the hybrid learning approach. Henceforth, we begin by undertaking an analysis that compares various network architectures for a common problem in the following section.

A. Role of network parameters in model performance

We start by presenting the statistical evaluations of models builds using different network architectures for all the testing samples. An effective representation of the distribution of the model performance across a large spectrum of testing samples is done via kernel density estimation (or KDE). Besides highlighting the model confidence, it further allows for the potential identification of outliers. The KDE is evaluated based on the mean absolute error, μ of the model predictions compared to the ground truth data, for the 100th testing frame. The mean absolute error, μ is evaluated as follows:

$$\mu = \frac{1}{n_c}\sum_{i=1}^{n_c}(|u_{\text{ref},i} - u_{\text{baseline},i}|) + (|v_{\text{ref},i} - v_{\text{baseline},i}|). \quad (5)$$

Figure 4 presents the approximate probability distribution via the KDE for the 50 testing samples from different network architectures. It is found that most of the approximate
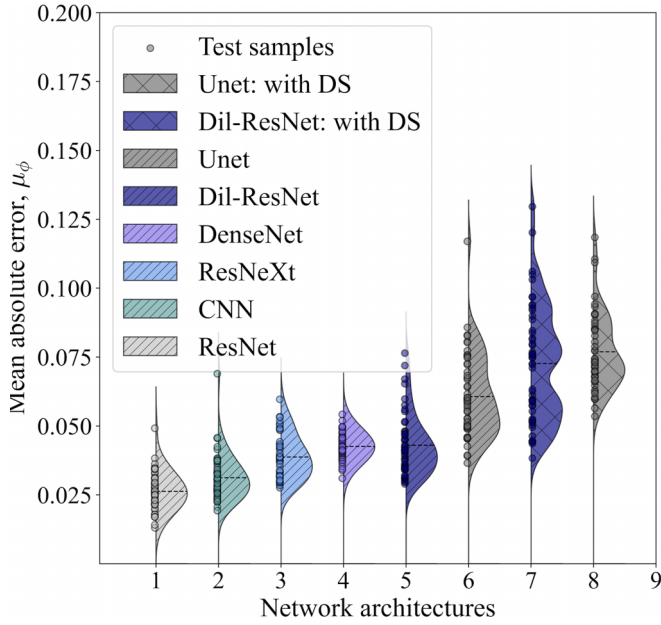
FIG. 4. Comparison of the mean absolute error μ at the end of 100th testing frame for 50 testing samples obtained from different NN architectures.

FIG. 5. Influence of random number seed on the performance of residual network (ResNet) architecture.

probability distribution is indicative of a Gaussian-like distribution. These include the KDE obtained from the convolution neural network (CNN) and the residual network-based models, i.e., ResNet [51], together with the newer variants ResNeXt [52], DenseNet [53], and diluted residual network (Dil-ResNet) from Ref. [30]. On the contrary, however, the Unet [54] exhibits multiple peaks in the distribution while also incurring greater sample mean error μ_N (see Table V). While attributing higher error to the architectures is not straightforward, strikingly, both the Unet with downsample (with DS) and the Dil-ResNet (with DS) architectures employ encoder blocks that compress the input data to a reduced dimension via the `stride` operation. This is not entirely surprising as the model struggles to retain local features of the input upon downsampling. We remark that identical choice of activation function, optimizer, learning rate η, initial weights θ_W, loss function, etc. have been used for all the architectures. Consequently, it can be mentioned that the models without any compression or convolution downsample serve as a better alternative for the present choice of problems.

Neural network-based learning is a stochastic, nonlinear optimization process that aims to iteratively improve the model predictions on training data while managing uncertainty and navigating through a complex, multidimensional solution space. In this context, using a gradient-based optimizer like Adam may result in a locally optimal solution with some sensitivity to the initial guess. To check the robustness of the model, the influence of the random number seed must be evaluated. We investigate the performance of the ResNet-based architecture for multiple seeds. Figure 5 shows the approximate probability density based on the mean absolute error μ obtained using ResNet-based architectures for different values of random number seed for all the testing samples. It can be observed that the Gaussian-like distribution of the error μ is retained for different random seed values, besides showing consistency in the model performance in terms of the sample mean μ_N (see Table VI). In light of these discussions, it can be remarked that the performance is independent of the random number seed. Finally, we also determine the influence of the adaptive learning rate η that determines the convergence and consequently the nature of solutions obtained. This is

TABLE V. Comparative performance among various NN architectures in terms of mean μ_N and standard deviation σ_N (across 50 testing samples, at the end of 100 testing frame).

Architecture	No. parameters	Mean (μ_N)	Standard deviation (σ_N)
CNN	546 478	0.03124	0.0082
ResNet	516 674	**0.02626**	0.00657
Unet (with DS)	520 342	0.07693	0.0138
Unet	520 342	0.06067	0.0152
Dil-ResNet (with DS)	548 066	0.07266	0.0211
Dil-ResNet	548 066	0.04303	0.01101
ResNeXt	524 322	0.03871	0.00787
DenseNet	566 578	0.04258	**0.0047**

TABLE VI. Comparative performance of ResNet architecture for different random seeds.

Random seed no.	Mean (μ)	Standard deviation (σ)
1	0.03029	0.00818
2	0.02643	0.00747
3	**0.02483**	**0.00706**
4	0.02991	0.00896
42	0.02626	0.0082

briefly discussed in Appendix C 1. The remainder of the study is based on the ResNet architecture.

B. Qualitative comparison of predictive models across wake categories

In this section, we compare various benchmark models, assessing their capability to accurately reproduce the dynamic wake characteristics for new, unseen testing samples with different spacing ratios. This analysis is fundamentally important because, intriguingly, different spacing ratios between the cylinders yield diverse wake behaviors, each of which necessitates faithful reproduction. For instance, the deflected gap wake should exhibit a narrow and a wide wake immediately downstream of the two subsequent cylinders, with the gap flow between these cylinders being deflected towards the narrow wake region. This deflection should remain constant with time. In contrast, for the flip-flopping or the chaotic wakes, the gap flow switches direction in an erratic manner. As such, while some wake patterns exhibit a periodic, uniform repetition, others veer into the realm of chaos. It is, therefore, a fascinating problem that tests the predictive capabilities of the models. Five representative testing samples are chosen from the available 50 experiments, each exhibiting a unique spacing ratio and the corresponding wake dynamics. We present vor-

ticity flowfields generated entirely from the network's output. This allows us to highlight the regions of interest, i.e., cylinder wake and near body boundary, besides also indicating the direction of rotation of the vortices. Figure 6 compares the vorticity flowfield obtained from the various benchmarks, including the *Reference* solver at the spacing ratio $L/D = 1.26$. The vorticity flowfield obtained from the *Reference* exhibits a single bluff-body wake with a certain shedding frequency. It is found that while all the baselines can reproduce the correct category of the wake, the *Source* and the supervised learning (SL) approach very quickly deviate from the correct vortex shedding cycle. However, the present DPNN approach compares remarkably well with the *Reference* data even for long time horizons. In addition, the shedding frequency is typically around 0.1 Hz or 10 s. So within the 300 Δt time horizon, the framework is expected to capture close to 30 vortex shedding cycles. However, this is an estimate and actually value would depend upon the L/D value of the cylinder layout.

Figure 7 presents the comparison of the vorticity flowfield obtained at the spacing ratio $L/D = 1.57$ for which the *Reference* solver portrays deflected gap wake. As alluded to earlier, the deflected gap flow shifts toward the narrower wake region (i.e., lower downstream cylinder in this case), with the direction of the gap flow remaining stable over time. Consequently, the predictive models should preserve this phenomenon while accurately predicting the vortex-shedding cycle. It is found from Fig. 7 that the predictions from *Source* quickly transitions to a single bluff-body wake, whereas the vorticity contours obtained from the SL approach transform into a hybrid quasi-in-phase wake. In contrast, however, the current DPNN approach aligns impressively with the *Reference* data across extended time horizons. This observation holds true for the deflected gap flow direction and the overall vortex shedding cycle.

Figure 8 showcases a comparison of the vorticity flowfield at a spacing ratio $L/D = 2.34$, where the *Reference* solver
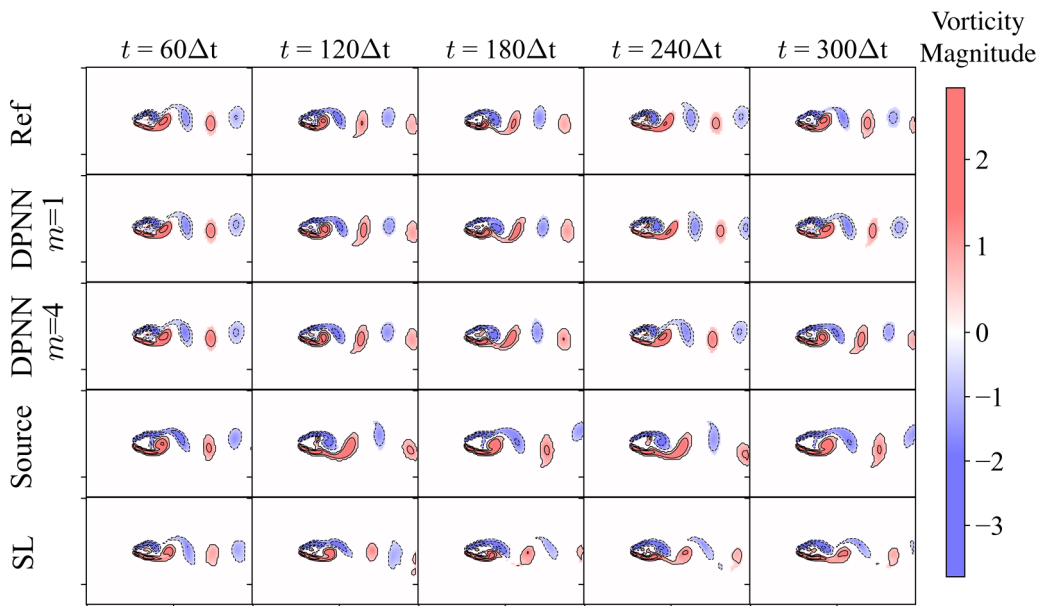

FIG. 6. Comparison of vorticity ω flowfields obtained using various benchmarks for different time instances for the spacing ratio $L/D = 1.26$ (single bluff-body wake, 17th testing sample).
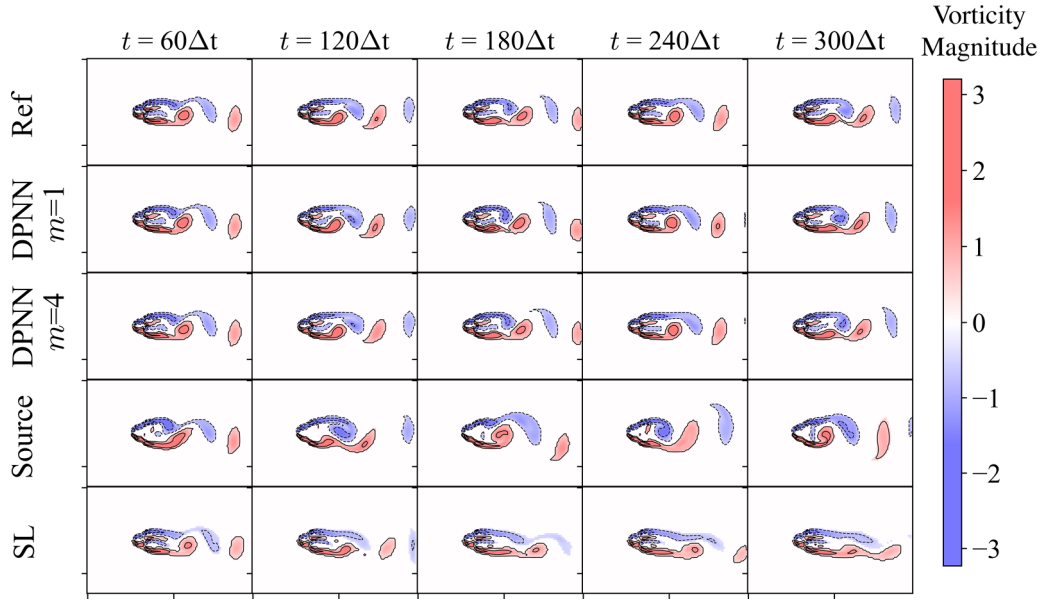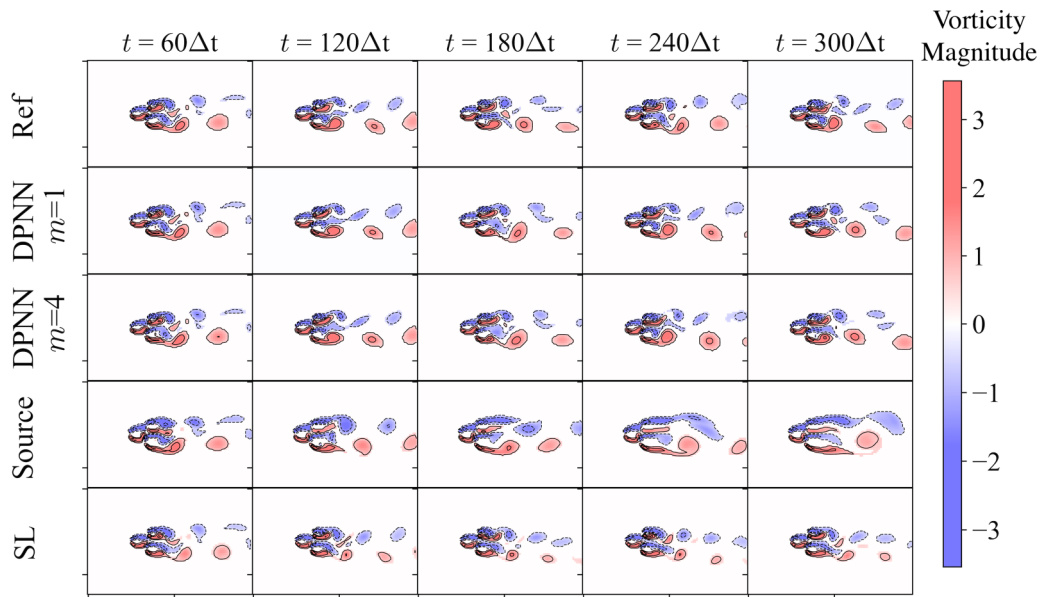
FIG. 7. Comparison of vorticity ω flowfields obtained using various benchmarks for different time instances for the spacing ratio $L/D = 1.57$ (deflected gap wake, 39th testing sample).

displays a flip-flopping (or chaotic) wake, i.e., the gap flow in between the two downstream cylinders chaotically switches direction. This is evident in Fig. 8 wherein the gap flow pointing downwards from the frame at $t = 60\Delta t$ has transitioned to the top at $t = 120\Delta t$. The predictions from DPNN seem to capture this switch correctly up to frame $t = 180\Delta t$, whereas, for long temporal horizons, both the gap flow direction and the vortex shedding cycles have visible differences. These differences further augments when compared to the vorticity flowfields obtained from *Source* and SL approaches.

Figure 9 exhibits a comparative analysis of vorticity flowfields at a spacing ratio of $L/D = 3.37$, which, as per the

Reference solver, corresponds to a fully developed quasi-in-phase wake. The clockwise rotating vortices (highlighted in blue) emitted from the freestream side of the upper cylinder and the gap side of the lower cylinder, appear to be in quasi-in-phase. The same can be mentioned for the anticlockwise rotating vortices (highlighted in red) shed from the freestream side of the lower cylinder and the gap side of the upper cylinder. It is found that among all the baselines, both DPNN and SL approaches perform well, with the former preserving the vortex structure quite remarkably for long-time horizons. However, the *Source* depicts a chaotic wake and fails to reproduce the actual

FIG. 8. Comparison of vorticity ω flowfields obtained using various benchmarks for different time instances for the spacing ratio $L/D = 2.34$ (chaotic wake, 41st testing sample).
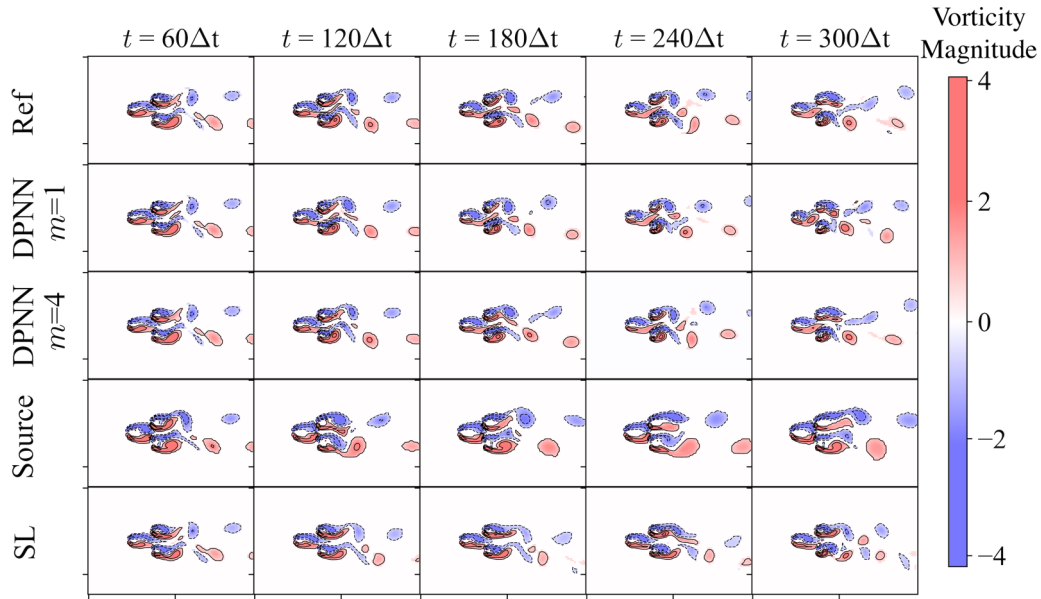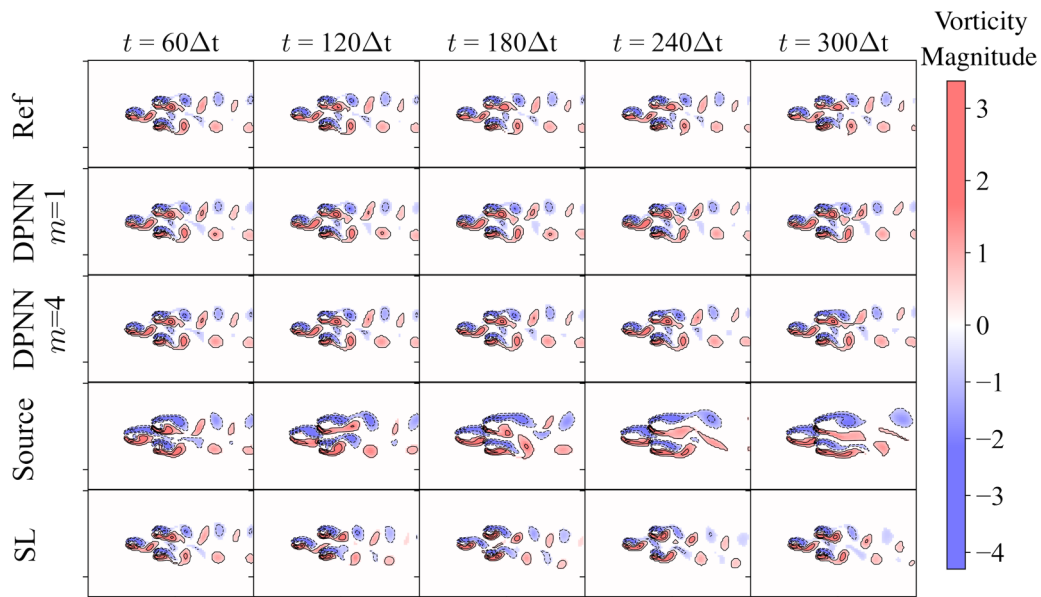
FIG. 9. Comparison of vorticity ω flowfields obtained using various benchmarks for different time instances for the spacing ratio $L/D = 3.37$ (fully developed quasi-in-phase wake, 26th testing sample).

wake dynamics for the given spacing ratio and geometric configuration.

Figure 10 highlights the vorticity flowfields obtained at the spacing ratio of $L/D = 3.89$ from various benchmarks. The flowfield obtained from the *Reference* solver indicates a fully developed quasi-anti-phase wake. It is observed that the vortices emitted from the freestream side of the two downstream cylinders shed vortices in the opposite direction. This dynamic nature of the wake is accurately retained by the present DPNN approach. On the contrary, the SL approach produces a hybrid wake, i.e., wake transitions into a hybrid wake, whereas the *Source* yields a chaotic wake.

The exercise underscored the crucial need for developing robust and precise models. These models should accurately identify the underlying wake category and deliver sustained temporal fidelity over extended periods. While the current section provides a qualitative comparison of individual baselines across multiple wake categories, the subsequent section will present a quantitative analysis of the key variables of interest.

C. Quantitative comparisons in physical and reduced feature space

In the preceding section, we employed velocity and vorticity flowfields as a qualitative measure of performance

FIG. 10. Comparison of vorticity ω flowfields obtained using various benchmarks for different time instances for the spacing ratio $L/D = 3.89$ (fully developed quasi-anti-phase wake, 32nd testing sample).
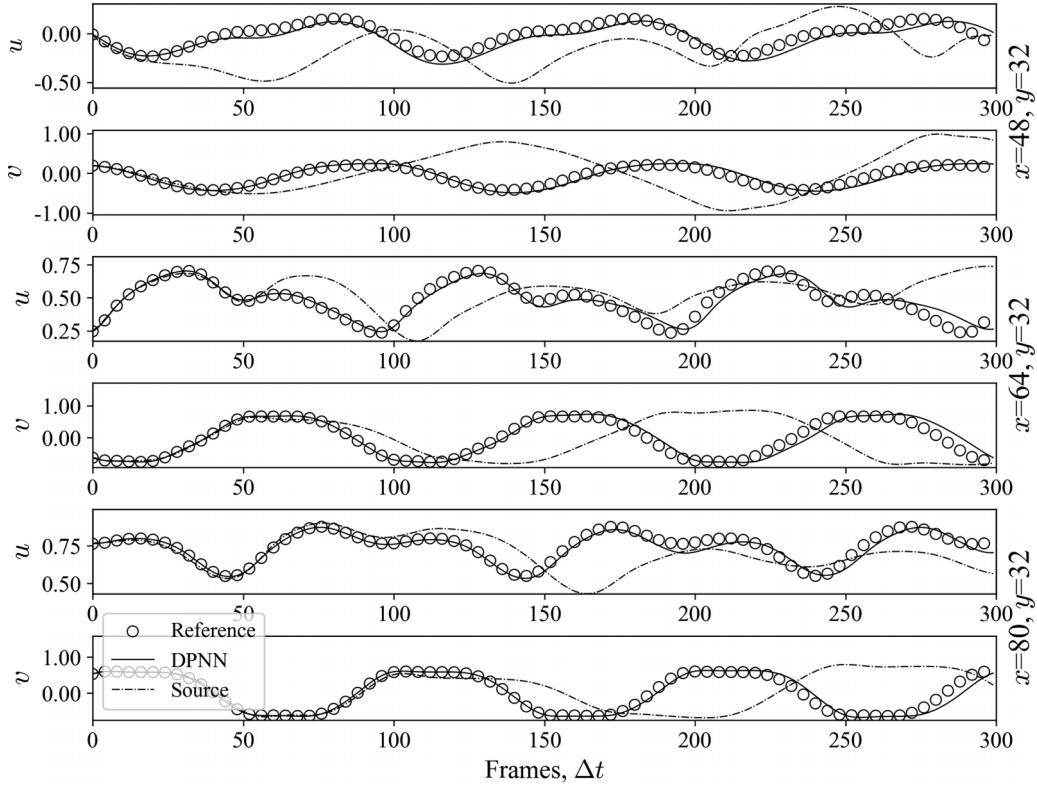
FIG. 11. Comparison of u, v-velocity components downstream of the bodies and its variation with time (39th testing sample).

accuracy. The network used in our study is trained purely on physical space, specifically the velocity flowfields as a collection of control volumes with staggered faces. Through the actions of the kernels, convolutions transform these physical quantities into an equivalent dimensional feature space, achieved through zero-padding. This process poses an engaging question: Does the network retain its performance in Fourier space or the reduced feature space offered by the first three principal components?

We begin our investigation by adopting the 39th testing sample, which depicts a deflected gap wake, as discussed in Sec. IV B. In Fig. 11, we present the temporal variation of the velocity components u, v at multiple spatial probes using different baselines. These probes are placed at the wake of the downstream cylinders along the centerline axis, i.e., $(n_{px}, n_{py}) = (48,32), (64,32), (80,32)$, where n_{px}, n_{py} represents the nodal positioning in the x and y directions, respectively. It must be noted here that the present DPNN approach from hereon refers to $m = 4$ steps of unrolling during training. Thus, an evaluation period up to $t = 300\Delta t$ is a substantial period wherein multiple vortex-shedding cycles are captured. It is found that the present DPNN framework results in very good agreement with the modulated signals obtained for both u, v velocity components, with the agreement seemingly better for probes placed further away from the bodies. The minor offset with the data obtained from the *Reference* solver tends to happen for higher time values t. On the contrary, however, this offset is rather early for the predictions obtained from the *Source* solver. Moreover, for higher values of time t, the predictions significantly differ from the *Reference*.

Figure 12 displays the distributions of u-velocity across several cross-sections aligned in the longitudinal direction (along the rows) at multiple instances of time (along the columns), obtained from multiple benchmarks. Barring some minor differences, we found that the current framework aligns exceptionally well with data obtained from the *Reference* solver across various spatial locations and at different time instances. Similar observations can also be drawn from Fig. 13, where the u-velocity distributions are plotted along multiple horizontal cross-sections at different instances of time. Naturally, these observations are also reflected in Fig. 14 where the mean absolute error, μ [calculated based on Eq. (5)] for the recursive predictions with respect to the ground-truth data are shown. It is found that the present DPNN-based approach yields the lowest cumulative error over time, whereas the data from the *Source* results in the highest growth of error over time. As such, it can be mentioned that the DPNN approach has successfully enabled a hybrid framework that can faithfully reproduce the dynamical behavior of the physical quantity (i.e., velocity flowfields). It must be remarked that the mean absolute error, μ may not always see a monotonic rise. Depending upon the vortex shedding cycle, the error (i.e., difference between velocity flowfield computed by the model and that obtained from the *reference* solver) may dip if the vortices are out of phase.

We now compute the power spectral density (or PSD) based on the velocity flowfields obtained from the predictive frameworks. Figures 15(a)–15(c) presents the PSD obtained from the velocity measurements at the $n_{px}, n_{py} = (64, 32)$ probe location with respect to different time ranges. For reference, node $n_{px}, n_{py} = (0, 0)$ is indexed in the bottom left.
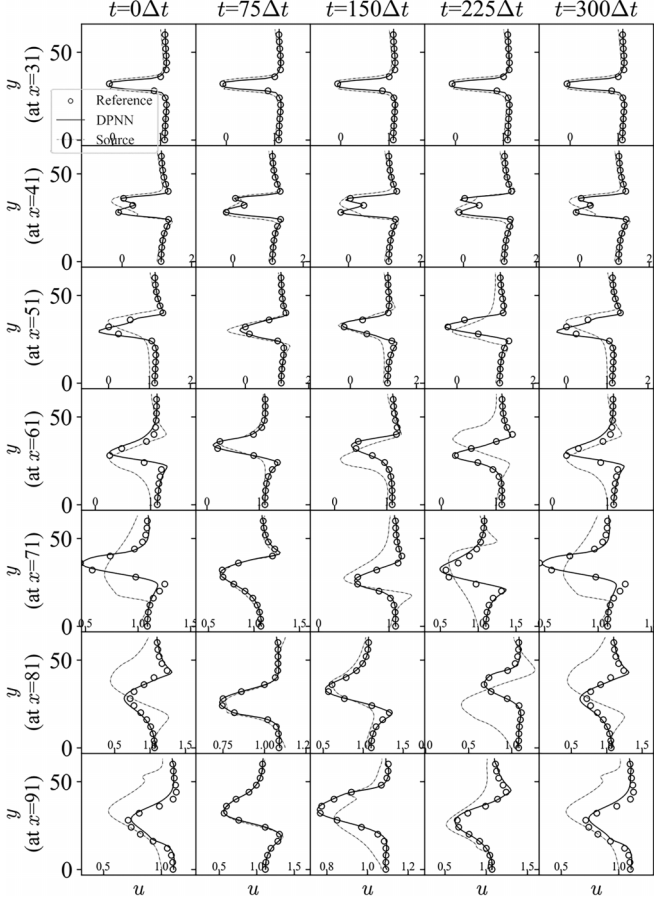
FIG. 12. Comparison of the *u*-velocity at multiple vertical cross-sections (39th testing sample).

The position $n_{px}, n_{py} = (64, 32)$ represents the two-thirds of the distance in streamwise direction and half the distance in the lateral direction from the origin.. It is found that for a relatively smaller time range of up to 100 Δt, the PSD versus frequency obtained from the DPNN aligns exceptionally well with the data obtained from the *Reference* solver signifying that the present framework is able to faithfully reconstruct small as well as large scale structures in the velocity flowfield. However, minor differences in the high-frequency range are found for large recursive predictions (i.e., up to 300 Δt), indicating small-scale differences in the velocity flowfields. This bodes well with the observations in Fig. 11, where observable differences are found for larger recursive predictions.

We perform principal component analysis (or PCA) on the predicted velocity fields obtained from the baselines to evaluate the principal components (or PC) and their variation with respect to time t. Figure 16 presents the first three PC's and their distribution with time. It is found that the temporal variation of PC 1 and PC 2 indicates a periodicity in the data obtained from the *Reference* solver, which seems to be captured correctly in predictions obtained by DPNN. Moreover, the temporal variation in the PC 2, PC 3, and PC 3, PC 1 components are also in good overall agreement with the *Reference* solver. The protracted vortex shedding cycle obtained from *Source* observed in Fig. 11 is also reflected in Fig. 16 in terms of the open-ended arc indicating a longer vortex

shedding time-period. As a result of these investigations, it can be remarked that the performance of the DPNN approach demonstrates consistency in both the physical and reduced feature space.

D. Reconstructing flowfields: Global and local perspectives

As we delve deeper into investigating the present coarse-grained surrogate model, it becomes increasingly clear that both local and global fluid structures are crucial for the overall representation of flow physics. Specifically, precisely reproducing local boundary layer phenomenon and global wake dynamics becomes essential. The intersection of these scales—the local and the global—is where true accuracy in understanding and predicting flow dynamics lies. To that extent, we investigate the ability of the learned models to correctly reconstruct the local mean velocity boundary layer profile (with momentum thickness, θ) as well as the mean gap flow $\bar{U}_G$, while also evaluating the global fluid variables in terms of kinetic energy KE and enstrophy Ω. This step is not trivial given that the *Source* solver employs a masked stair-step representation for the underlying body boundary for computational efficiency.

It is also crucial to remark here that while three level coarsening will naturally influence the local resolution of the boundary layer profile it still retains a parabolic profile for the upstream body, as will be shown later on. Thus at a given level of grid resolution, the goal is to mitigate the difference between the under-resolved boundary layer of the *Source* and the resolved boundary layer of the *Reference* transferred to the *Source* mesh. Hence, our goal is not a super-resolution task, i.e., transferring a low-resolution solution to a higher resolution, but instead to improve the solution given a fixed computational mesh.

The momentum thickness, θ is evaluated as follows:

$$\theta(x) = \left[\int_0^{\delta_{\max}} \frac{\bar{u}(x,y)}{\bar{u}_e(x)} \left(1 - \frac{\bar{u}(x,y)}{\bar{u}_e(x)} \right) dy \right]. \quad (6)$$

The time and space averaged streamwise velocity at the gap [43], U_G/U is evaluated as follows:

$$\frac{U_G}{U_\infty} = \frac{1}{U_\infty} \int_{-0.5}^{0.5} \bar{u} \, d\left(\frac{y}{G} \right), \quad (7)$$

where $\bar{u}$ is the time-mean streamwise velocity at the gap. The kinetic energy KE and enstrophy Ω are calculated as follows:

$$KE(t, U) = \frac{1}{2} \int_{L_{2(A)}} U(t)^2 dA, \quad (8)$$

$$\Omega(t, U) = \frac{1}{2} \int_{L_{2(A)}} \omega(t)^2 dA. \quad (9)$$

While KE is a popular choice for statistical evaluation of fluid flows, enstrophy represents the rotational energy of the flow and corresponds to the dissipative effects of the flow [55].

Figure 17 presents the boundary layer profile in the lateral y direction just above the body boundary along the cross-section drawn at the upstream cylinder for three distinct testing samples. The sharpness in the boundary layer profile is a result of grid resolution, i.e., while the original grid resolution employed to obtain the ground truth solution was done using ($n_c = 768 \times 512$) it was later downsampled

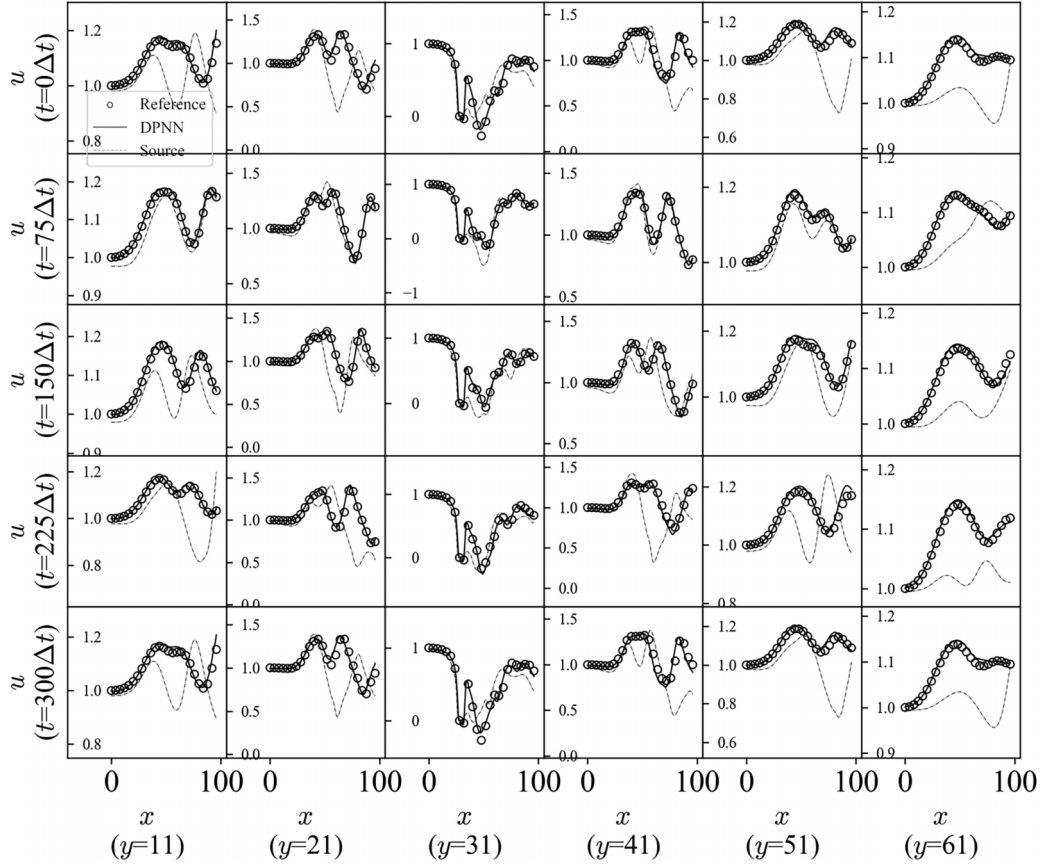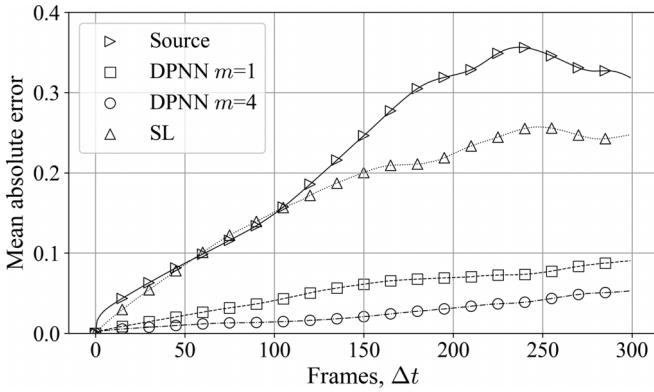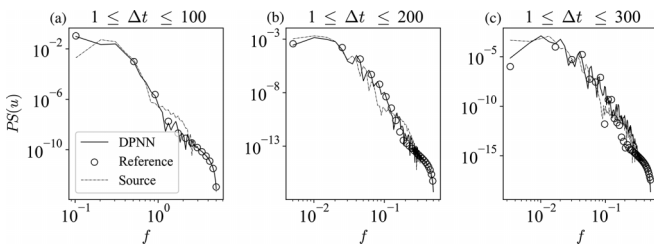FIG. 13. Comparison of the u-velocity at multiple horizontal cross-sections (39th testing sample).

FIG. 14. Variation of the mean absolute error from various frameworks with testing frames for the 39th testing sample.

FIG. 15. Comparison of the power spectral density of the streamwise velocity for multiple time ranges (39th testing sample). For reference, node $n_{px}, n_{py} = (0, 0)$ is indexed in the bottom left.

to ($n_c = 96 \times 64$). Despite the fact that downsampling high-resolution data obtained from the *Reference* solver can distort the actual boundary profile, reconstructing the steep rise in velocity remains a critical element to be preserved. It is evident that the inadequate treatment of the body boundary results in a much thicker boundary layer profile obtained from the *Source*. This results in an enlarged effective diameter of the body, subsequently changing the equivalent effective spacing ratio. Such alteration points to a potential difference in the wake
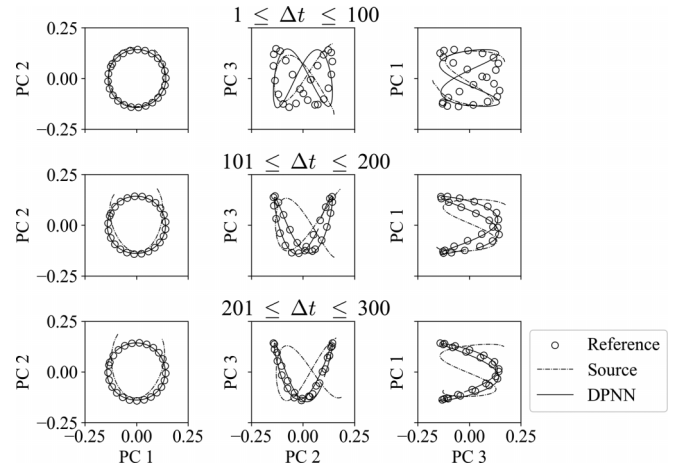
FIG. 16. Comparison of the first three principal components and its variation with time (39th testing sample).
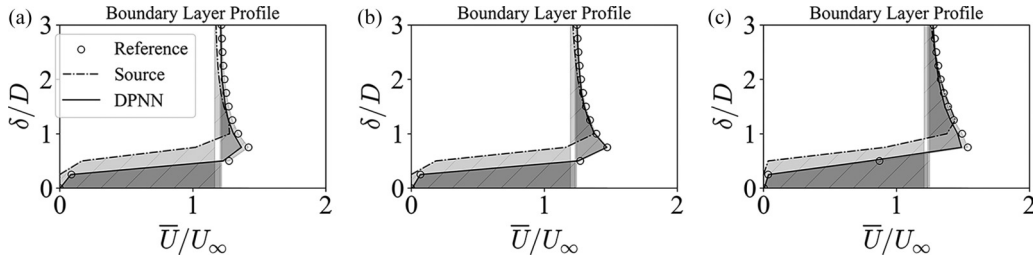
FIG. 17. Comparison of the time-averaged boundary layer profile along the lateral y direction of the upstream body for (a) 28th, (b) 40th, and (c) 46th testing samples.

category (single bluff-body wake instead of deflected gap wake from *Source* in Fig. 7). On the contrary, the boundary layer profile obtained from the DPNN approach agrees quite well with the *Reference* data. This is also observed in Table VII, where the momentum thickness θ/D has been reported for three representative testing samples followed by Fig. 18 where we plot the boundary layer momentum (BLM) thickness θ/D obtained from multiple frameworks for all testing samples arranged in ascending manner. It is clear that the DPNN-based velocity flowfield indeed closely reconstructs the θ/D values for all testing samples, barring very few outliers.

Figure 19 shows the streamwise mean gap velocity profile distribution between the two downstream regions. The prominent peaks adjacent to the parabolic profile near the center are caused by the interaction of the separated shear layer from the upstream cylinder with the gap side shear layers of the downstream cylinders. This interaction penetrates the gap flow. There seems to be a very good agreement of the mean streamwise velocity profile at this gap obtained from the DPNN approach with that of the *Reference*, whereas the thicker boundary layer profile exhibited by the *Source* results in a narrower gap profile. Finally, we also quantify the time and space averaged streamwise velocity at the gap $\frac{U_G}{U_\infty}$ in Table VIII, where a very good agreement between DPNN and *Reference* data has been noted. In addition, we also compare the mean stream velocity (MSV) obtained from multiple frameworks for all testing samples in Fig. 20. It is again found that the DPNN-based velocity flowfields result in excellent agreement of $\frac{U_G}{U_\infty}$ as compared to the *Reference* solution. On the contrary, however, the velocity flowfields obtained from *Source* suffer from severe under predictions. Along the same vein, it is thus clear that the corrections offered by the NN in

the hybrid DPNN-based framework indeed produces solutions that are consistent with actual physics.

To enable a comparison between global variables that represent interesting fluid phenomena, we focus on assessing the enstrophy, symbolized by Ω, and the kinetic energy, represented by KE. Tables IX and X present the time-averaged enstrophy Ω and kinetic energy KE obtained from multiple benchmarks. It is found that the flowfields obtained from *Source* exhibit considerably lower values of Ω and KE, indicating loss of rotational and kinetic energy over time. In other words, the flowfields obtained from *Source* result in significant decay of kinetic and rotational energy due to stronger dissipation effects resulting from the coarse grid simulations. In contrast, however, the time-averaged enstrophy and kinetic energy obtained from DPNN are in excellent agreement with the data obtained from the *Reference* solver. Given the fact that the same numerical schemes are employed in the base solver for both the *Source* and the DPNN (i.e., the solver embedded in the loop), the network thus acts as a forcing function that counters the strong numerical dissipation introduced by the *Source* solver. This exercise, thus, clearly points towards the ability of the DPNN approach to yield predictions that preserve the local boundary layer profile along with the global wake dynamics in a way that results in lower numerical dissipation, as achieved by the *Reference* solver. In this light, it is crucial to remark that the comparisons made in Tables VII-X are sampled at or from different locations, i.e., while momentum thickness θ/D and mean streamwise velocity at the gap $\frac{U_G}{U_\infty}$ are sampled at specific discrete location, the mean enstrophy, Ω/Ω_0 as well as mean kinetic energy, KE/KE_0 are evaluated over the entire domain. Consequently, the relative accuracy of predictions from DPNN for these quantities differ. Hence comparison across of these aforementioned quantities in Tables VII-X should be made with care. So far, we have demonstrated the reliability of the model at an individual pre-

TABLE VII. Comparison of the boundary layer momentum thickness θ/D for multiple testing samples.

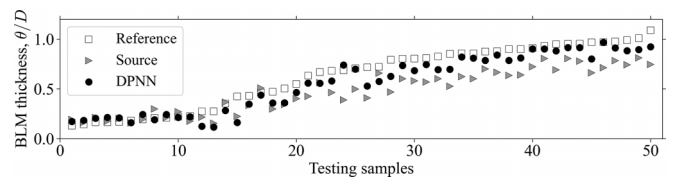| No. testing samples	Boundary layer momentum thickness (θ/D)		
	Reference	*Source*	DPNN
28	0.852	0.527	0.697
40	0.91	0.723	0.902
46	1.01	0.812	0.895

FIG. 18. Boundary layer momentum thickness (arranged in ascending order) obtained from various frameworks for all testing samples.
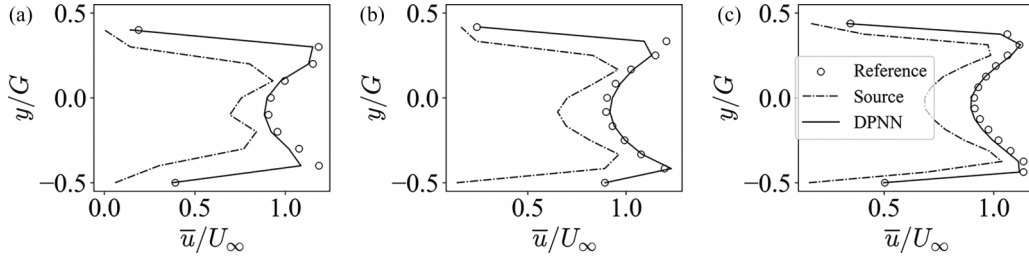
FIG. 19. Comparison of the time-averaged streamwise gap velocity profile in between the two downstream bodies for (a) 28th, (b) 40th, and (c) 46th testing samples.

diction level. Additional evaluations for all the 50 previously unseen test samples have also been performed and are shown in Appendix A, supporting the conclusions drawn above.

E. Insights through model predictions

A key aspect of our research is the model's ability to produce not only robust but also practical and valuable predictions. As we advance, we will undertake a comprehensive evaluation to further substantiate the model's reliability in producing results comparable to established literature. Specifically, we will compare nondimensional parameters such as the Strouhal number (St) and spacing ratio (L/D), which are typically associated with the flow around two or more cylinders. This broadens our evaluation beyond individual testing samples, as was conducted in previous sections. To our knowledge, this marks the first occasion when such an investigation has been carried out for arbitrarily shaped bodies.

Despite the fact that the present dataset corresponds to flowfields obtained for flow past arbitrary configurations, one can still compare the distributions of the Strouhal number St based on the dominant frequency of one of the two downstream cylinders with the spacing ratio L/D. The diameter in the spacing ratio L/D is simply the projected height of the upstream cylinder. This results in the distribution of the nondimensional number (i.e., Strouhal number and spacing ratio), which allows for comparison with Ref. [43], who report the data obtained for flow past equidiameter cylinders computed using an immersed boundary-based flow solver. Figure 21 compares the Strouhal number based on the dominant frequency of the downstream cylinder (upper) as a function of the spacing ratio L/D. Figure 21 offers multiple insights, viz., besides the observable oscillations in the Strouhal number, a good overall fit between the solutions from the two different methodologies is noted. This illustrates that despite the arbi-

trary nature of the embedded bodies, the underlying frequency of vortex shedding (nondimensionalized by the corresponding diameter and freestream velocity) for a certain body at a given spacing ratio exhibits similar patterns. Besides, it is also noted that the transition from low-frequency single bluff-body wake to deflected gap wake and subsequently to high-frequency, chaotic wake happens at nearly the same spacing ratio (as compared to Ref. [43]). This observation is especially important given that the Strouhal number accuracy is crucially dependent on long-term recursive predictions, and deviations cause significant differences in the vortex shedding frequency [28], e.g., from *Source* (not shown here). Consequently, it can be remarked that the present DPNN-based predictive framework allow for useful and reliable predictions enabling long-term temporal accuracy. While multiple factors, such as shear layer growth, instabilities, gap flow, etc., contribute to the frequency of vortex shedding, further investigations are needed to probe the cause of the fluctuations.

In prior sections, we emphasized the necessity of evaluating the performance of the predictive model using test samples representative of different wake categories. In this section, we expand on this by presenting additional metrics of interest. This exercise allows for inferring greater physical insights from the predictions obtained so far, thus offering a unique perspective on solutions obtained from the predictive framework. Figure 22 represents the time-averaged enstrophy Ω evaluated over the spacial domain for 300 testing frames for all testing samples, color marked by the category of the wake. It is found that there exist clear strata or bands of regions belonging to certain wake categories, e.g., single bluff-body, deflected gap, and chaotic wake, whereas these bands overlap for the fully developed quasi-in/-anti-wake categories. This indicates that the mean rotational energy of the flowfields is lowest for the single bluff-body wake and highest for the fully developed quasi-in-phase wake. This is not entirely surprising as the vortices shed from the upstream cylinder in the fully

TABLE VIII. Comparison of the time and space averaged streamwise velocity at the gap $\frac{U_G}{U_\infty}$ for multiple testing samples.

| No. testing samples	Mean streamwise velocity at the gap ($\frac{U_G}{U_\infty}$)		
	Reference	*Source*	DPNN
28	0.896	0.530	0.857
40	0.956	0.657	0.958
46	0.958	0.728	0.946
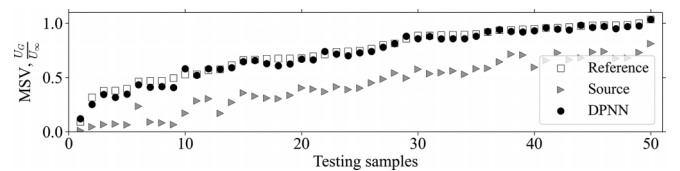
FIG. 20. Mean streamwise velocity at the gap (arranged in ascending order) obtained from various frameworks for all testing samples.

TABLE IX. Comparison of the normalized time-averaged enstrophy Ω/Ω_0 for multiple testing samples.

No. testing samples	Mean enstrophy (Ω/Ω_0)		
	Reference	*Source*	DPNN
28	1.001	0.926	0.985
40	0.998	0.944	0.979
46	0.999	0.988	0.997

TABLE X. Comparison of the normalized time-averaged kinetic energy KE/KE_0 for multiple testing samples.

No. testing samples	Mean kinetic energy (KE/KE_0)		
	Reference	*Source*	DPNN
28	1.000	0.924	0.991
40	1.000	0.929	0.995
46	1.000	0.927	0.996
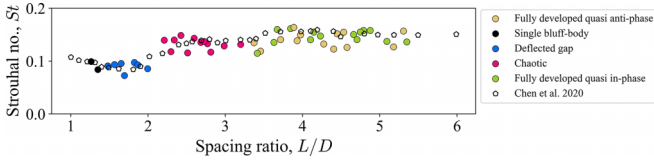
FIG. 21. Strouhal number along with the spacing ratio for the arbitrarily shaped cylinders and its comparison with Chen *et al.* [43]. Strouhal number is evaluated based on the dominant frequency of the downstream (upper) cylinder obtained for 300 frames.
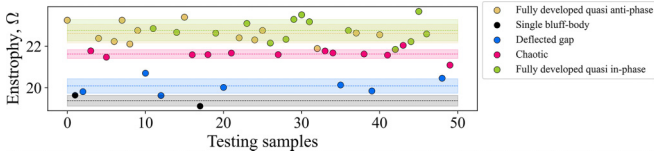
FIG. 22. Comparison of the enstrophy Ω of the predictions obtained from DPNN color-coded by the wake category. The color bars represent the region within the maximum and the minimum Ω for each wake category.
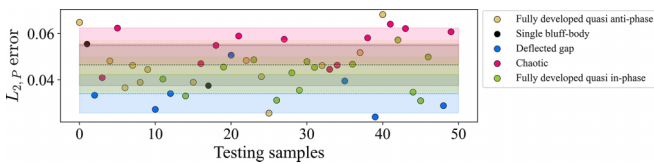
FIG. 23. Comparison of the L_2 norm error of the predictions obtained from DPNN.
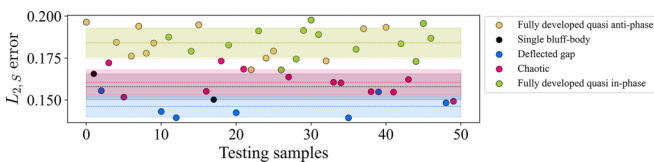
FIG. 24. Comparison of the L_2 norm error of the predictions obtained from *Source*.
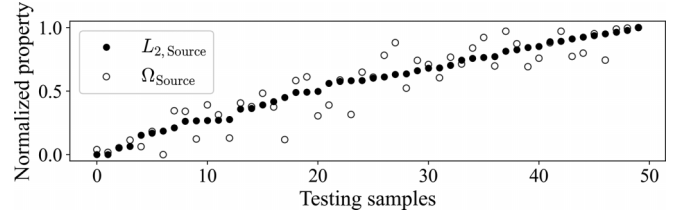
FIG. 25. Comparison of the enstrophy with L_2 norm of the predictions from *Source* solver, sorted in ascending manner.

developed wake category penetrate the gap region while merging with vortices shed from the downstream cylinder, thereby increasing the rotational energy of the flow:

$$L_{2,\phi} = \sqrt{\frac{\sum_{i=1}^{nc} \left(\phi_{\text{Ref},i}^j - \phi_{\text{baseline},i}^j\right)^2}{n_x \times n_y}}. \qquad (10)$$

Figure 23 illustrates the time-averaged L_2 norm error computed using Eq. 10 for each testing sample based on the velocity field up to $t_f = 300$ frames. The variables n_x, n_y are the numbers of computational cells in the streamwise and lateral directions, and ϕ represents the quantity of interest. The samples are color-marked according to the wake category they exhibit. Unsurprisingly, it is found that the testing samples belonging to the chaotic wakes incur the highest error. Additionally, the testing samples corresponding to the deflected gap wake result in the lowest error from both DPNN and Source, which can be attributed to the stable gap flow (see Fig. 24). On the contrary, the testing samples exhibiting fully developed quasi-in-phase wake result in the highest error from the *Source*. To probe further, in Fig. 25, we compare the normalized enstrophy Ω for the flowfields obtained from *Source* and the corresponding L_2 norm error, each in ascending order. It is found that there exists a clear relationship between the dissipation resulting from the higher enstrophy in the flowfields and the corresponding error. This suggests that the error in predictions obtained from *Source* is predominantly influenced by the numerical dissipation.

F. Generalizability towards out-of-distribution cases

A crucial component towards determining the performance of a neural network-based predictive model is its generalizability to new, unseen datasets. As such, an evaluation with test data precludes inherent biases of the training dataset. Evaluations on out-of-distribution (OOD) datasets allows for estimating the robustness of the model while potentially identifying the model's inherent bias towards certain kind of data. In addition, such practices hint towards the applicability of the model to real-world data, which are likely to be OOD. We have carefully designed experiments for OOD datasets which allow us to perform the following:

1. *Three equidiameter cylinders.* This setup mimics the existing one as shown in Fig. 2 with the exception of the geometric symmetry induced by three equidiameter cylinders at spacing ratios of $L/D = [1.5, 3.5, 5.0]$ with each cylinder having unit diameter. This serves as an interesting problem considering that the model was never
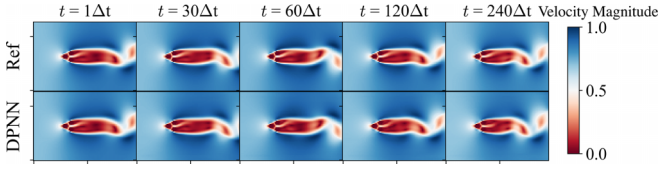
FIG. 26. Comparison of velocity flowfield obtained from DPNN and *Reference* solver at multiple time instances for flow past the three-cylinder case at $L/D = 1.5$.

FIG. 28. Comparison of velocity flowfield obtained from DPNN and *Reference* solver at multiple time instances for flow past the two cylinder case at $L/D = 1.5$.

trained for symmetry (the equidiameter cylinders result in symmetric distribution about the x axis). Additionally, this allows for a direct comparison of the wake category with that obtained by Zheng *et al.* [42]. For the chosen spacing ratio, the flowfield should exhibit deflected gap wake, antiphase wake, and fully developed in-phase wake, respectively.

2. *Two side-by-side cylinders*. This layout chooses two side-by-side equidiameter cylinders which is contrary to the present training setup to assess the performance of the model for an entirely alternative problem of practical relevance. The spacing ratios chosen for this case are $L/D = [1.5, 2.5, 4.0]$, which allow for direct comparison with Bao *et al.* [56] and should yield flip-flopping (or chaotic) wake, in-phase wake, and antiphase wake, respectively.

1. Test case 1: Three cylinders

Figure 26 presents the velocity flowfields obtained from the DPNN approach and its comparisons with the data obtained from the *Reference* solver at multiple time instances, i.e., three equidiameter cylinders arranged in an equilateral triangle layout for the spacing ratio $L/D = 1.5$. The *Reference* (or ground truth data) solver depicts a deflected gap flow wake, with the deflected gap flow being deflected towards the narrow wake region (upper cylinder), which bodes well with the observations by Zheng *et al.* [42]. It is found that the predictions obtained by the present DPNN approach also result in the same wake dynamics. In addition, the vortex shedding cycle result in an excellent agreement with the data obtained from the *Reference* solver, which instills further confidence in the long-term temporal predictions.

Figure 27 presents the velocity flowfields obtained from the present approach and its comparison with the data obtained
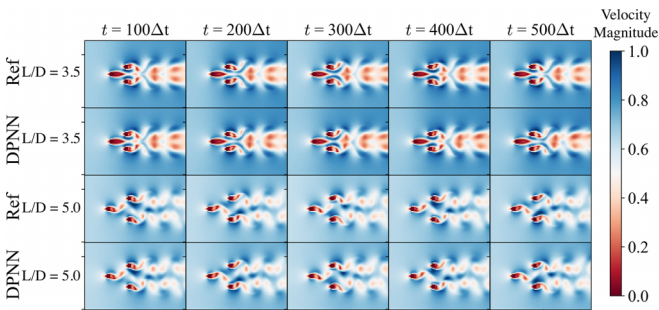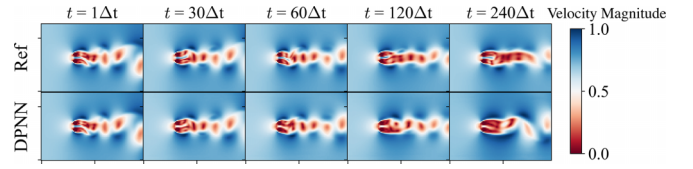
from the *Reference* solver at the spacing ratios $L/D = 3.5$ and 5, up to a total time of $t = 500\Delta t$. The wake exhibited at these spacing ratios clearly depicts antiphase and fully developed in-phase patterns, respectively, as reported by Zheng *et al.* [42]. The comparison shows that DPNN is able to preserve the overall wake structures remarkably well for long time horizons, irrespective of either in-phase or fully developed antiphase wake.

2. Test case 2: Two cylinders

Figure 28 highlights the velocity flowfield obtained from the present approach and its comparison with the corresponding ground truth data at multiple time instances, i.e., two side-by-side cylinders at a spacing ratio $L/D = 1.5$. The data obtained from the *Reference* solver exhibits flip-flopping wake pattern wherein the gap flow switches direction in a chaotic manner and augers well with the observation by Bao *et al.* [56]. This can be clearly seen in Fig. 28, e.g., the gap flow at $t = 1\Delta t$ (directed towards the lower cylinder) switches direction at $t = 30\Delta t$ (directed towards the upper cylinder). Addressing such a tumultuous shift poses a significant challenge for any predictive framework [57,58]. For instance, while DPNN accurately predicts the first two shifts, its effectiveness diminishes when it comes to long-term forecasting, particularly struggling with the accurate reconstruction of chaotic wake dynamics. Nevertheless, considering the fact that the proposed framework yields satisfactory performance for predicting the chaotic dynamics for midtime horizons, this acts as a promising direction for future research.

Figure 29 presents the velocity flowfield obtained from the present DPNN approach and its comparison with the *Reference* data for spacing ratios $L/D = 2.5$ and 4 for multiple time instances. The wake dynamics exhibit in-phase and antiphase wake, respectively, which bodes well with the observations made by Bao *et al.* [56]. It is found that the predictions

FIG. 27. Comparison of velocity flowfield obtained from DPNN and *Reference* solver at multiple time instances for flow past the three-cylinder case at $L/D = 3.5$ and 5.
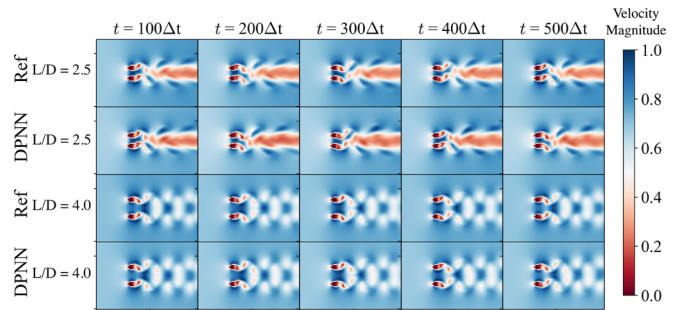

FIG. 29. Comparison of velocity flowfield obtained from DPNN and *Reference* solver at multiple time instances for flow past the two cylinder case at $L/D = 4$ and 2.5.

from the present approach compare really well with the data obtained from the *Reference* solver, especially for the antiphase wake at $L/D = 4$ where it is seen to preserve the symmetry of the flow about the centerline x axis. For the in-phase wake obtained at $L/D = 2.5$, the flowfields diverge at long-time predictions, i.e., $t = 500\Delta t$. Therefore, given these observations, it is encouraging to note that the current DPNN approach is successful in faithfully reconstructing wake dynamics for periodic flows (both in-phase and antiphase) over extended time horizons. Although the accuracy of predictions for chaotic wakes is limited to midtime horizons, the progress made so far offers a solid foundation for further refinement and optimization.

3. Computational complexity

The hybrid DPNN framework incurs a similar computational cost as that of the base Phiflow solver, depending on training or inference phase. This is because Phiflow is embedded into the training itself. For instance, for one training sample within a batch of training dataset, the L_2 norm loss evaluation requires m unrolling steps, necessitating m *Source* solver calls. Consequently, the cost associated for every training sample scales with the number of Phiflow calls. However, and more importantly, during inference we make only 1 Phiflow call, per time step. Consequently, the inference is relatively much quicker than training. In the present work, we use an Intel Xeon E5-1650 CPU and a NVIDIA GTX 1080Ti GPU. The training period using a single GPU on 50 experiments having 100 frames each takes approximately 70 minutes per epoch.

V. CONCLUSIONS

This work implements a differentiable physics-assisted neural network (DPNN) approach as a coarse-grained surrogate for unsteady incompressible flow past arbitrarily shaped bodies for a low Reynolds number of around 100. The bodies under consideration are arranged in a staggered equilateral triangle manner at multiple spacing ratio $1.2 \leqslant L/D \leqslant 5.5$ to produce wakes of varying nature. The main conclusions from the work can be summarized as follows:

(1) The DPNN-based predictive framework outperforms the standard data-driven supervised learning approach and the *Source* as a robust coarse-grained surrogate for unsteady fluid flows past arbitrary bodies. This has been qualitatively and quantitatively verified using multiple performance measuring indices that indicate accurate reconstruction of local (boundary layer) and global (wake dynamics) fluid phenomenon and for both physical quantities and Fourier space or reduced feature space.

(2) The framework has also been evaluated for out-of-distribution test samples for cases that assess the ability of the predictive framework to preserve symmetry (i.e., antiphase wake), stable gap flow (i.e., deflected gap wake), long-term temporal stability, chaotic dynamics (i.e., flip-flopping wake). It is found that the framework is largely wake agnostic, i.e., it renders a good prediction irrespective of the wake category. However, for the chaotic or flip-flopping wake, the framework can only accurately maintain the chaotic switching of the gap flow up to a certain midtime range.

(3) The L_2 errors computed on the predictions obtained from DPNN and *Source* are lowest for the deflected gap wake owing to its stable gap flow as well as long vortex formation length, the latter of which occupies the bulk of the computational space. The DPNN-based predictions seem to suffer from the onset of chaotic dynamics, which later manifest into inaccurate wake dynamics. The flowfields from *Source* seem to suffer from large dissipation due to the underlying coarse computational grid, which results in the continuous decay of kinetic/rotational energy.

(4) The Strouhal number distributions with the spacing ratio for the arbitrary bodies results in good agreement with the literature based on flow past equidiameter bodies. Also, the transition in the wake category from a single bluff-body wake to a deflected gap wake and eventually to a chaotic wake occurs at nearly the same spacing ratios. Moreover, we discover similar categories and patterns in the wake exhibited by the arbitrary bodies, except that in-phase and antiphase wakes yield in quasi-in-/anti-phase wakes. Additionally, the arbitrary nature of the bodies promotes an early vortex shedding by the upstream cylinder and precludes quasi-in-phase wakes from occurring at all.

(5) The enstrophy calculated based on the flowfields obtained from DPNN exhibit clear strata of test samples belonging to unique wake categories. This is particularly noticeable for single bluff-body, deflected gap, and chaotic wakes, whereas these delineations tend to overlap for the fully developed quasi-in-/anti-phase wake categories.

In view of these discussions, we postulate that such a strategy that incorporates both a low-fidelity solver as well as the neural network as a hybrid predictive framework derives merit from each component. Such a strategy paves the way for the low-fidelity solver to incorporate as much of the underlying physics as possible while the network learns the difference between low-fidelity solver and high-fidelity data. This specific attribute is key that makes such a framework more promising than purely data-driven supervised learning.

The present approach also shows potential merits as a generic reconstruction strategy for near-body boundary fluid properties. This is especially true for non body-conformal sharp interface immersed boundary approaches for high Mach and Reynolds numbers flows [59] that are prone to suffer from heat flux reconstruction at the body boundary. Moreover, the present learning framework would also be interesting avenue for flows over rough surfaces [60,61] with practical relevance. In addition, the generalizability of the approach opens up the potential directions for learning the kinematics of dispersed spherical particles [62].

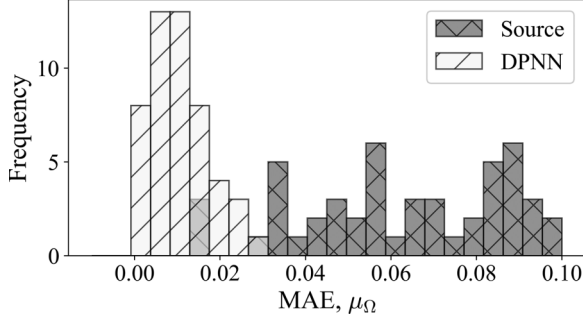The information related to the data and code supporting this study are openly available in Ref. [63].

FIG. 30. Comparison of the mean absolute error based on the enstrophy for 300 frames for all the 50 testing samples via histogram.

APPENDIX A: COMPREHENSIVE EVALUATIONS ACROSS ALL TESTING SAMPLES

Sections IV B and IV C evaluates representative testing samples via quantities of interest such as KE, Ω, etc. To present a comprehensive assessment of the present predictive framework for all testing samples, we compute the mean absolute error μ_ϕ for the predictions based on the kinetic energy KE and enstrophy Ω as follows:

$$\mu_\phi = \frac{1}{n_f} \sum_{j=1}^{n_f} |\phi_{\text{Ref}} - \phi_{\text{baseline}}|. \tag{A1}$$

Figure 30 illustrates the mean absolute error evaluated based on enstrophy μ_Ω for $n_f = 300$ frames for all 50 individual testing samples. Upon observation, it becomes evident that the flowfields generated by our current DPNN methodology consistently result in lower μ_Ω for all the testing samples. This observation also holds true for the mean absolute error based on kinetic energy μ_{KE}, as shown in Fig. 31. This suggests that, without exception, the predictions derived from the current DPNN methodology are consistently accurate and significantly outperform the *Source*.

APPENDIX B: VERIFICATION OF WAKE FLOW REGIMES

In this section, we undertake a few additional verification test cases for flow past three equidiameter cylinders placed in an equilateral position, as adopted in Ref. [43] for multiple spacing ratio L/D. This exercise allows for qualitative reproduction of the expected wake flow regimes for the given spacing ratio, as reported in the literature and further affirms the correctness of the numerical setup employed in the flow
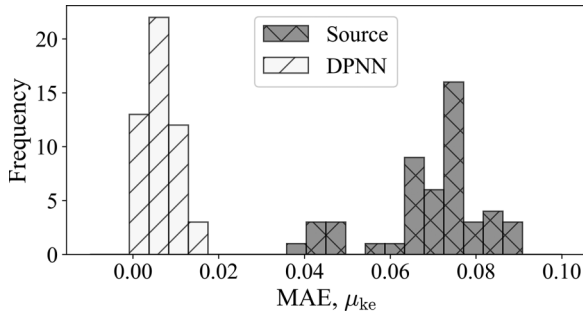

FIG. 31. Comparison of the mean absolute error based on the kinetic energy for 300 frames for all the testing samples via histogram.
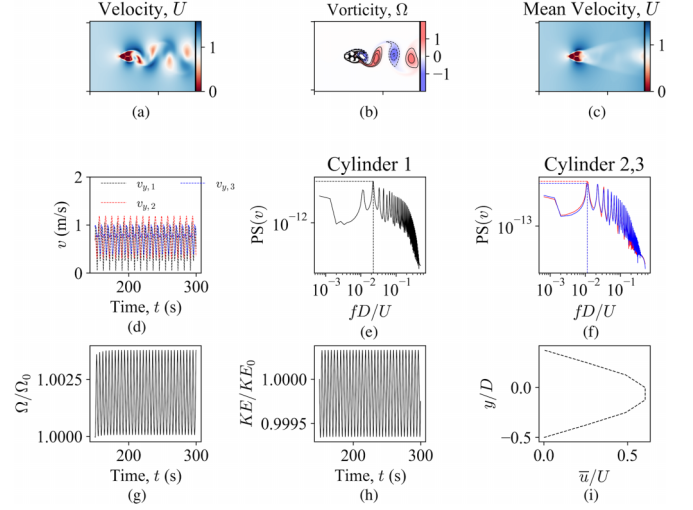

FIG. 32. Single bluff-body wake obtained from the *Reference* solver at $L/D = 1.2$ depicted using (a) instantaneous velocity U flowfield, (b) instantaneous vorticity ω flowfield, (c) time-averaged velocity $\overline{U}$ flowfield, (d) lateral velocities with time based on the probe placed just downstream of the cylinders, (e) PSD based on lateral velocity downstream of upstream cylinder 1, (f) PSD based on lateral velocity in the wake of the downstream cylinders 2,3, (g) scaled enstrophy Ω/Ω_0 with time, (h) scaled kinetic energy KE/KE_0 with time, and (i) mean velocity at the gap $\overline{u}/U$.

solver. In addition, this also sheds light towards the procedure adopted to categorize the flowfield into a distinct wake. Naturally, for the sake of fairness, we adopt the same conditions in terms of Reynolds number Re and spacing ratio, L/D, i.e., Re $= 100$ at L/D of (1.2, 1.5, 2.25, 3.5, 5). For the aforementioned spacing ratios, the expected wake flow should exhibit single bluff-body wake, deflected gap flow, flip-flopping flow, antiphase flow, and fully developed in-phase flow, respectively.

Figure 32 presents the characteristics of the flowfield obtained at the spacing ratio L/D=1.2, which also agrees with Ref. [43]. It can be found from Figs. 32(a) and 32(b) that the velocity and corresponding vorticity flowfields depict a single bluff-body wake, i.e., the cluster of bodies shed vortex as if a single bluff-body. The time-averaged mean velocity flowfield shown in Fig. 32(c) is descriptive of the zone of influence of the downstream wake. The lateral v velocity measured just downstream of the three cylinders exhibits periodicity in the flow with time. The power spectral density (or PSD) calculated based on the lateral v velocities is shown in Figs. 32(e) and 32(f) for upstream cylinder and downstream cylinders, respectively. It is found that the downstream cylinders exhibit identical dominant frequencies of vortex shedding f. In addition, Figs. 32(g) and 32(h) portray the scaled enstrophy Ω/Ω_0 and scaled kinetic energy KE/KE_0, that indicates periodicity of the flowfield. Last, Fig. 32(i) indicates the mean gap flow based on the time-averaged streamwise velocity normalized by freestream velocity. It is found that at such small L/D, the gap flow between the two downstream cylinders is weak and exhibits a parabolic profile. With increasing L/D, the shear layer emitted from the upstream cylinder is captured by the gap.
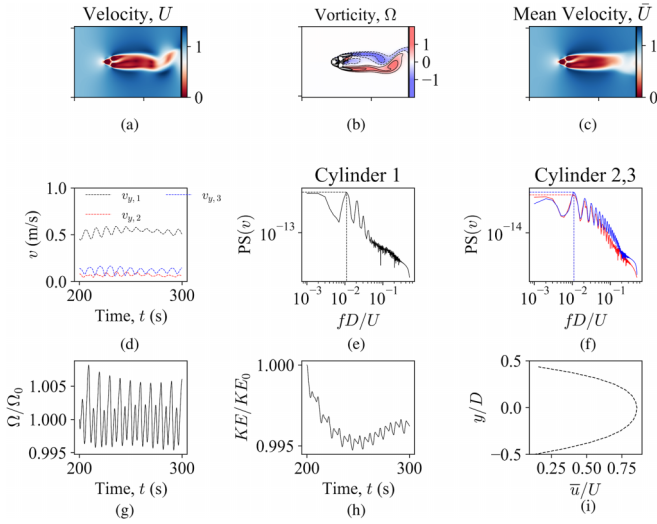
FIG. 33. Deflected gap wake obtained from the *Reference* solver at $L/D = 1.5$ depicted using (a) instantaneous velocity U flowfield, (b) instantaneous vorticity ω flowfield, (c) time-averaged velocity $\overline{U}$ flowfield, (d) lateral velocities with time based on the probe placed just downstream of the three cylinders, (e) PSD based on lateral velocity downstream of the cylinder 1, (f) PSD based on lateral velocity in the wake of the downstream cylinders 2,3, (g) scaled enstrophy Ω/Ω_0 with time, (h) scaled kinetic energy KE/KE_0 with time, and (i) mean velocity at the gap $\overline{u}/U$.

Figure 33 depicts the attributes of the deflected gap wake obtained at a spacing ratio $L/D = 1.5$, as also reported in the independent work by Refs. [42,43]. Figures 33(a) and 33(b) shows the instantaneous velocity U and vorticity ω flowfields, which is typical of the deflected gap wake in terms of the long vortex formation length with the gap flow pointing towards the narrow wake region (i.e., upper downstream cylinder as reported in Ref. [43] in Fig. 3 of their work). The temporal stability of the gap flow is clearly shown in terms of the time-averaged velocity field in Fig. 33(c). The lateral velocities measured just downstream of the three cylinders are plotted in Fig. 33(d), which fluctuate about a mean position. The PSD evaluated based on these lateral velocities in the immediate wake of the three cylinders are shown in Figs. 33(e) and 33(f), where the dominant frequency of the two downstream cylinders are found to be identical. Figures 33(g) and 33(h) present the temporal variation of the scaled enstrophy as well as kinetic energy, with minor dissipation. Last, Fig. 33(i) portrays the parabolic mean velocity at the gap. The spacing ratio $L/D = 1.5$ is not large enough for the detached shear layer from the upstream cylinder to be fully trapped within the gap in this case, and they follow the freestream side of the two downstream cylinders.

Figure 34 presents the features exhibited by a chaotic wake for a spacing ratio $L/D = 2.25$, as also reported by Ref. [43]. This category of wake exhibits irregular switching of the gap flow direction, which is indicated in the temporal variation of the lateral velocities measured just downstream of the three cylinders shown in Fig. 34(d). This transition in the gap is chaotic in nature and also results in an altered frequency of vortex shedding by the two downstream cylinders [as shown in Fig. 34(f)]. The irregularity in the flow is also present in the
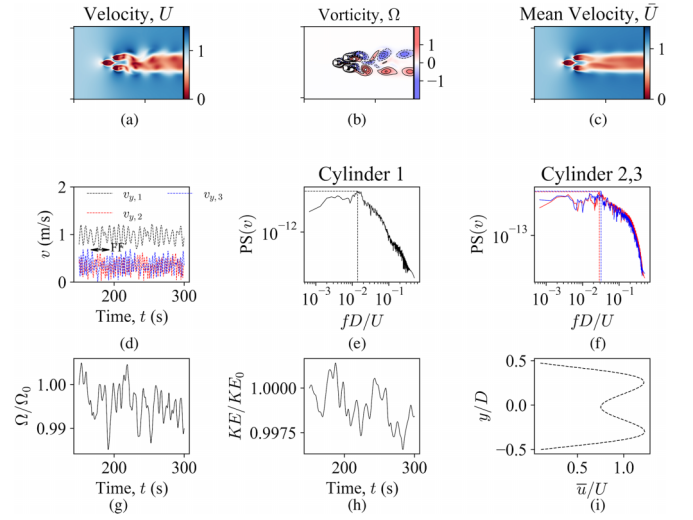

FIG. 34. Flip-flopping chaotic wake obtained from the *Reference* solver at $L/D = 2.25$ depicted using (a) instantaneous velocity U flowfield, (b) instantaneous vorticity ω flowfield, (c) time-averaged velocity $\overline{U}$ flowfield, (d) lateral velocities with time-based on probe placed just downstream of the cylinders, (e) PSD based on lateral velocity downstream of upstream cylinder 1, (f) PSD based on lateral velocity in the wake of the downstream cylinders 2,3, (g) scaled enstrophy Ω/Ω_0 with time, (h) scaled kinetic energy KE/KE_0 with time, and (i) mean velocity at the gap $\overline{u}/U$.

temporal variation of the scaled enstrophy and kinetic energy, as seen in Figs. 34(g) and 34(h). The shear layer shed from the upstream cylinder now enters the gap region along the inner side of the two downstream cylinders and cases two peaks in the time-averaged gap flow [see Fig. 34(i)].
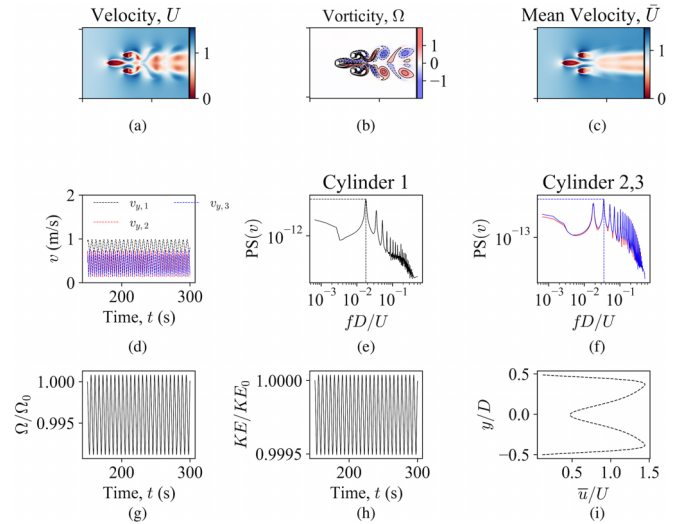

FIG. 35. Antiphase wake obtained from the *Reference* solver at $L/D = 3.5$ depicted using (a) instantaneous velocity U flowfield, (b) instantaneous vorticity ω flowfield, (c) time-averaged velocity $\overline{U}$ flowfield, (d) lateral velocities with time based on the probe placed just downstream of the cylinders, (e) PSD based on lateral velocity downstream of upstream cylinder 1, (f) PSD based on lateral velocity in the wake of the downstream cylinders 2,3, (g) scaled enstrophy Ω/Ω_0 with time, (h) scaled kinetic energy KE/KE_0 with time, and (i) mean velocity at the gap $\overline{u}/U$.
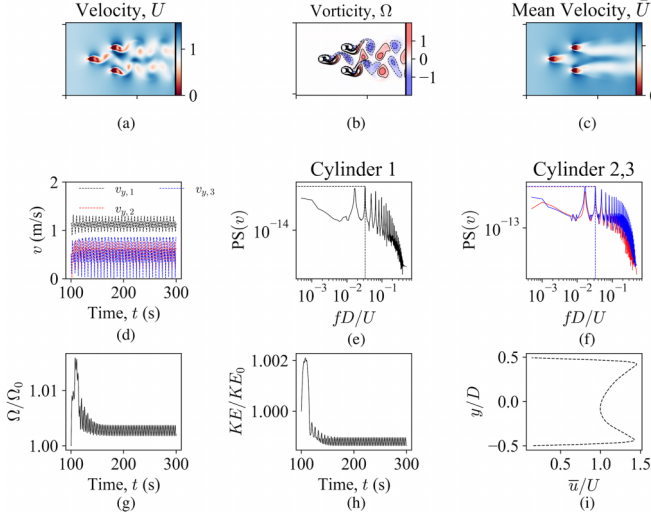
FIG. 36. Fully developed in-phase wake obtained from the *Reference* solver at $L/D = 5$ depicted using (a) instantaneous velocity U flowfield, (b) instantaneous vorticity ω flowfield, (c) time-averaged velocity $\overline{U}$ flowfield, (d) lateral velocities with time based on probe placed just downstream of the cylinders, (e) PSD based on lateral velocity downstream of upstream cylinder 1, (f) PSD based on lateral velocity in the wake of the downstream cylinders 2,3, (g) scaled enstrophy Ω/Ω_0 with time, (h) scaled kinetic energy KE/KE_0 with time, and (i) mean velocity at the gap $\overline{u}/U$.

Figure 35 indicates the characteristics of the antiphase wake obtained at a spacing ratio $L/D = 3.5$, as also reported by Refs. [42,43]. The vorticity flowfield shown in Fig. 35(b)

indicates the counter-rotating vortices while maintaining symmetrical distribution along the centerline x axis, resulting in an identical dominant frequency of vortex shedding by the two downstream cylinders. Besides, this category of wake exhibits a uniform variation of the lateral velocity as well as the scaled enstrophy and kinetic energy, as also seen in the distribution of scaled enstrophy and kinetic energy in Figs. 35(g) and 35(h). Further, Fig. 35(i) indicates greater penetration of the shear layer emitted from the upstream cylinder, as evident from the prominent peaks in the gap flow profile.

Figure 36 presents the features of a fully developed in-phase wake, obtained at a spacing ratio $L/D = 5$, as also confirmed by Refs. [42,43]. For this category of wake, the vortices are also shed by the upstream cylinder and are swallowed by the gap between the two downstream cylinders and later merge with the vortices shed by the two downstream cylinders. Besides, such a category of wake results in a uniform variation of the lateral velocity as well as scaled enstrophy, and kinetic energy, resulting in an identical dominant frequency of vortex shedding by the two downstream cylinders.

Figure 37 presents the time-averaged mean velocity flowfields of all the 100 datasets, arranged in increasing L/D values (left to right, top to bottom), for the time range 150 s $\leqslant t \leqslant 300$ s. It can be found that the first few datasets exhibit a single bluff-body wake. At slightly higher spacing ratios, the wake transitions to *wake* interference as the downstream cylinders are fully submerged in the wake of the upstream cylinder. At moderate spacing ratios, the wake pertains to *proximity + wake* interference due to partial immersion of the downstream cylinders within the wake of the upstream
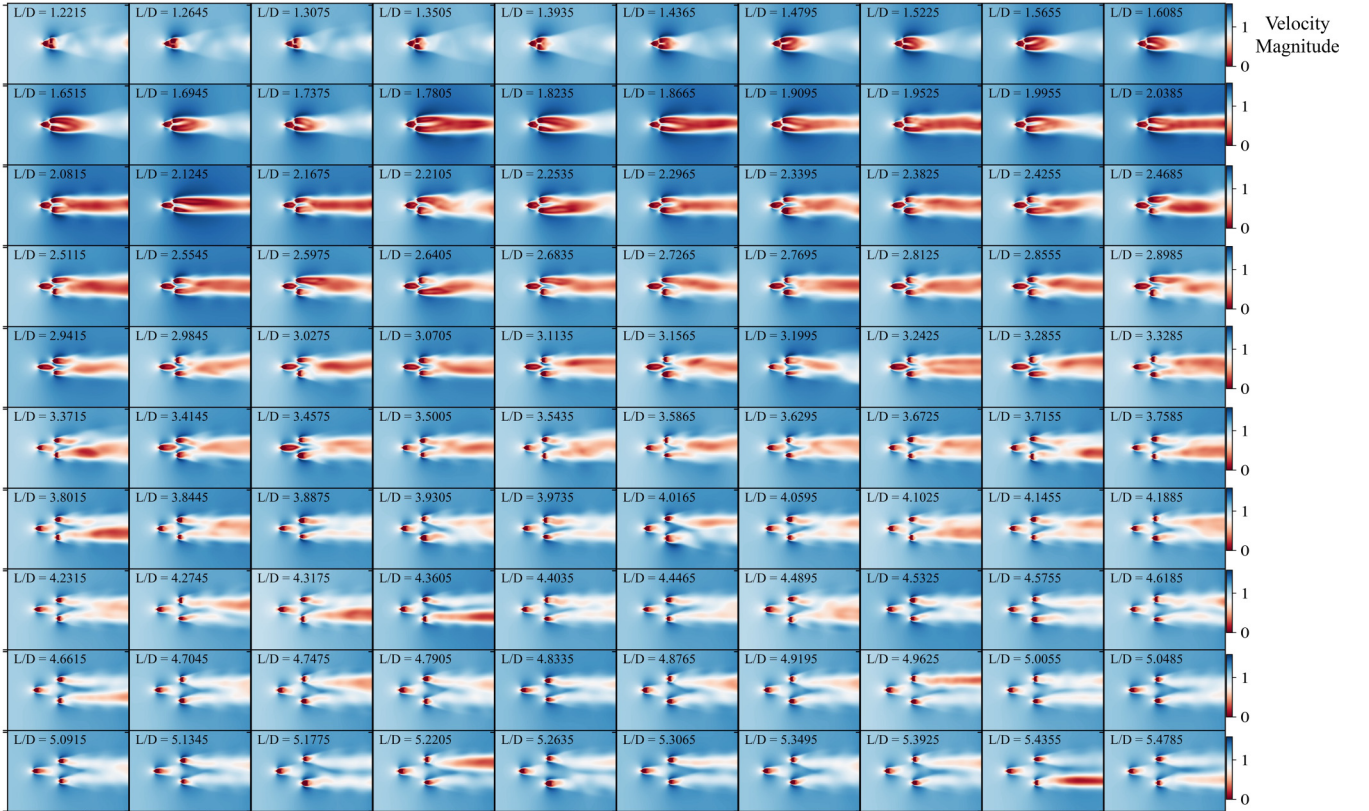

FIG. 37. Time-averaged velocities fields obtained from the *Reference* solver for all datasets arranged in the order of ascending spacing ratio.
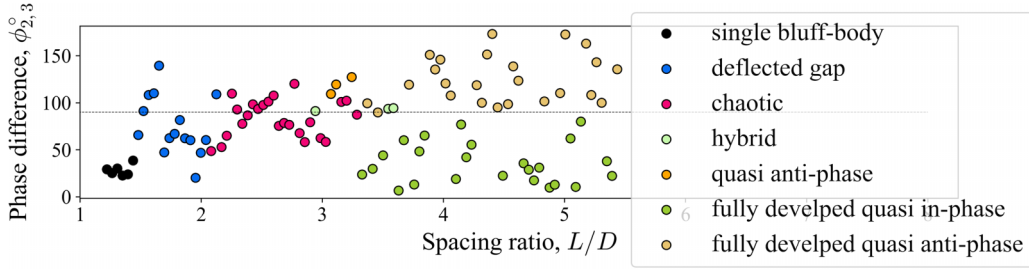
FIG. 38. Comparison of phase difference based on the v velocity probed in the wake of the two downstream cylinders with respect to spacing ratio L/D obtained from the *Reference* solver.

cylinders. At the high spacing ratio values, the wake of the upstream cylinders has minimal interference with the downstream cylinders and hence corresponds to the no-interference wake category.

Besides probing the velocity fields, we analyze the temporal distribution of the cross-stream velocities probed in the immediate wake of the two downstream cylinders. This enables us to further categorize the wake into in-phase (or quasi-in-phase) and antiphase (quasi-anti-phase). Specifically, we use cross-correlation to evaluate the phase difference between the two downstream cylinders using their respective cross-stream v-velocities. Figure 38 presents the phase difference $\phi_{2,3}$ of the 100 training datasets in terms of their spacing ratio L/D. It can be observed that there is a sudden jump in the $\phi_{2,3}$ between the single bluff-body wake and deflected gap wake. This result bodes well with the observations reported in Ref. [43]. In addition, there exists a clear

difference in the $\phi_{2,3}$ values for the quasi-anti-phase wake and fully developed quasi-anti-phase wake flow, which is in line with the expectations. It is also found that unlike the reported observations in Ref. [43], the present study shows a transition from quasi-anti-phase wake to fully developed quasi-in-phase wake due to early shedding of the vortices by the upstream cylinder. Consequently, the arbitrary nature of the bodies promotes such a transition and precludes quasi-in-phase wake from appearing at all. Figures 39 and 40 show the continuous wavelet transformation on the cross-stream velocities for two representative samples that depict quasi-anti-phase and fully developed quasi-in-phase wake flow, respectively. Finally, Fig. 41 presents the effective Reynolds number Re for all the individual datasets calculated based on the height of the upstream cylinder H_{c1}. It is found that the Re values between $75 \leqslant$ Re $\leqslant 100$, among the 100 experiments.
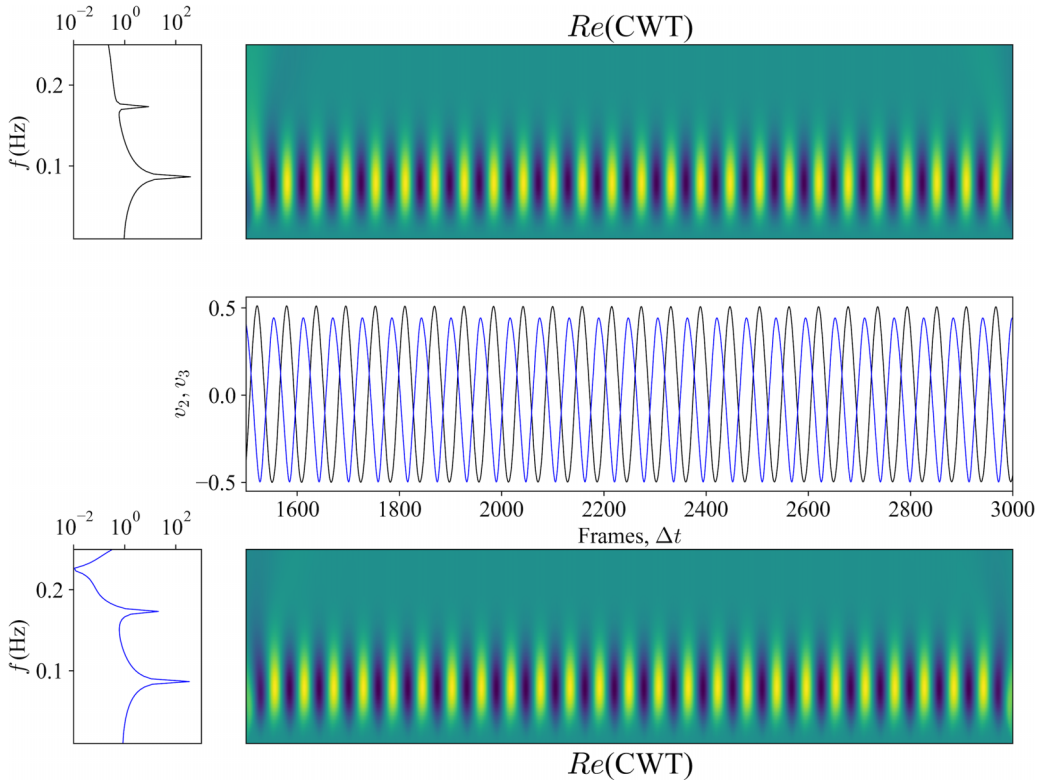

FIG. 39. Continuous wavelet transformation and the associated frequency distribution obtained using the *Reference* solver for the v-velocity distributions of the 83rd dataset. The upper row corresponds to the v velocity obtained from the wake of the upper downstream cylinder (black), whereas the lower row corresponds to the lower downstream cylinder (blue).

FIG. 40. Continuous wavelet transformation and the associated frequency distribution obtained using the *Reference* solver for the v-velocity distributions of the 95th dataset. The upper row corresponds to the v velocity obtained from the wake of the upper downstream cylinder (black), whereas the lower row corresponds to the lower downstream cylinder (blue).
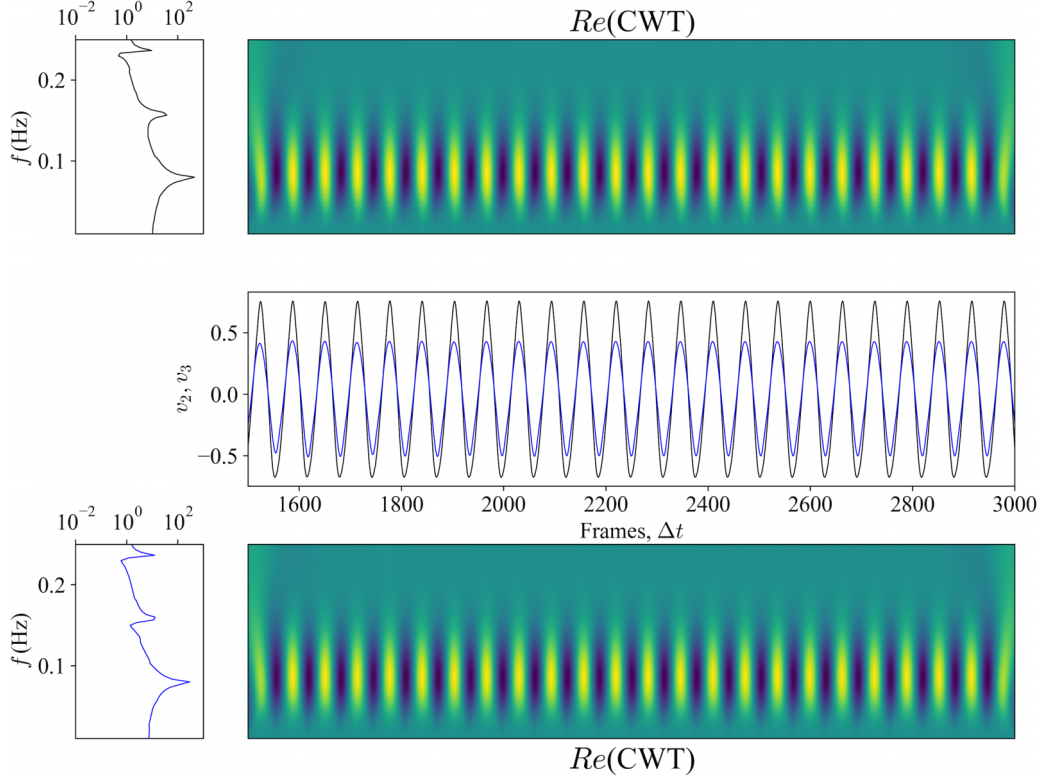
APPENDIX C: FINE TUNING NEURAL NETWORK ARCHITECTURE AND PARAMETERS

1. Adaptive learning rate

While choosing a constant learning rate, η is straightforward, it may heavily influence the decay of the loss function. For instance, an extremely low η may lead to low convergence, whereas for relatively high η may result in divergence or oscillatory convergence. This issue can be remedied using a learning rate schedule, *i.e.*, start with a relatively high η and progressively reduce it with training epochs. The influence of a constant and a variable η is shown in Fig. 42. It is found that while the loss decays with training epochs, the influence of learning rate η on the loss is minimal. We incorporate a variable learning rate for the remainder of the study. We remark that 50 epochs are sufficient for obtaining a convergence in the training loss.

2. Residual network: ResNet

Figure 43 describes a typical residual network (or ResNet) architecture [51] wherein multiple ResNet blocks have been connected in series with skip-connections in between. Each ResNet block comprises two convolutional layers having 32 filters, each with a kernel of size of [5 × 5]. The input is first fed to a convolutional layer whose output is then fed to a ResNet block. There are a total of nine ResNet blocks, the output of which is activated by the rectified linear activation function (or ReLU) activation function. Such a deep network results in a total of 516 674 trainable parameters.

3. Convolutional neural network: CNN

A fully convolutional neural network (or CNN) architecture has been adopted [64]. The CNN architecture comprises
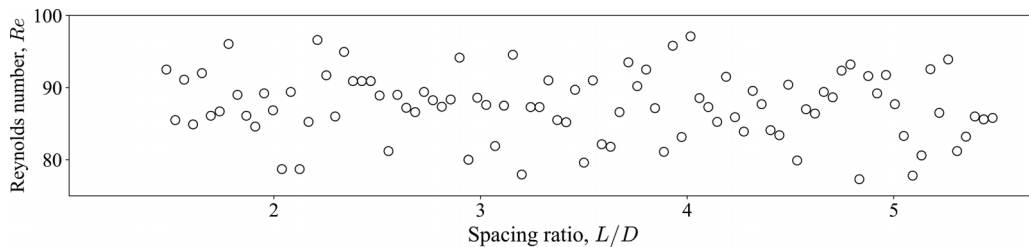

FIG. 41. Comparison of the effective Reynolds number Re for all 100 individual datasets based on the characteristic length scale of the upstream cylinder (cylinder height H_{c1}).
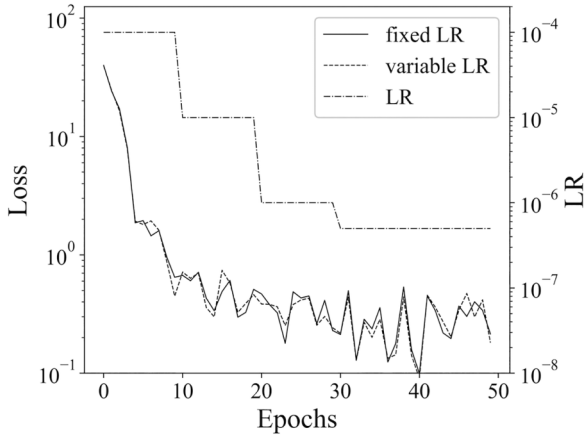
FIG. 42. Influence of learning rate schedule on the convergence. The dash-dotted line indicates the sequential decay of the learning rate value with epochs.

input, hidden, and output layers. For the present case, we have adopted 11 hidden layers of kernel size $= [5 \times 5]$ and filters varying as (4, 8, 16, 32, 64, 128, 64, 32, 16, 8, 4). To preserve the size of the inputs, we employ padding while also using the ReLU activation function. No downsampling or max-pooling was applied throughout the CNN layers. This setting resulted in a total of 546 478 trainable parameters.

4. U-Net

A U-Net-based architecture has been adopted from Ref. [54] that comprises an encoder path followed by a decoder path with skip connections between each level of encoding and decoding. Each encoding level consists of two convolutional layers followed by a $[2 \times 2]$ Maxpooling operator for downsampling. The number of filters for each level is chosen as (4, 8, 16, 32), with the final convolutional block comprising 64 filters, each having a fixed kernel size of $[5 \times 5]$. The output from the $[2 \times 2]$ Upsamping operator is concatenated with the corresponding output from the encoded level. The resulting feature maps are then fed to a convolution layer. The number of filters in each convolution layer in the decoded path is chosen as (32, 16, 8, 4). This architecture resulted in a total of 520 342 trainable parameters.

5. Diluted residual networks: DilResNet

A dilated residual network (or Dil-ResNet) architecture has been adopted from an independent work by Ref. [30], who presented this architecture for 2D as well as 3D turbulent flows with an aim to learn high-resolution turbulent fluid dynamics using low spatial and temporal resolutions. This architecture employs an encoder and decoder (i.e., using MaxPooling and Upsampling, respectively), each containing a single convolutional layer without any activation. In addition to the encoder and decoder layer, a processor is connected in between to enable the encode-process-decode paradigm. The processor consists of four dilated CNN blocks connected in series and residual connections between them. Each CNN blocks comprise 7 dilated CNN layers, each having $n_f = 36$ number of filters, along with dilation rate $D_R \in 1, 2, 4, 8, 4, 2, 1$ and ReLU activation. These values have been kept the same as the original work by Ref. [30] to allow for a fair comparison. The DilResNet architecture results in a total of 548 055 trainable parameters.
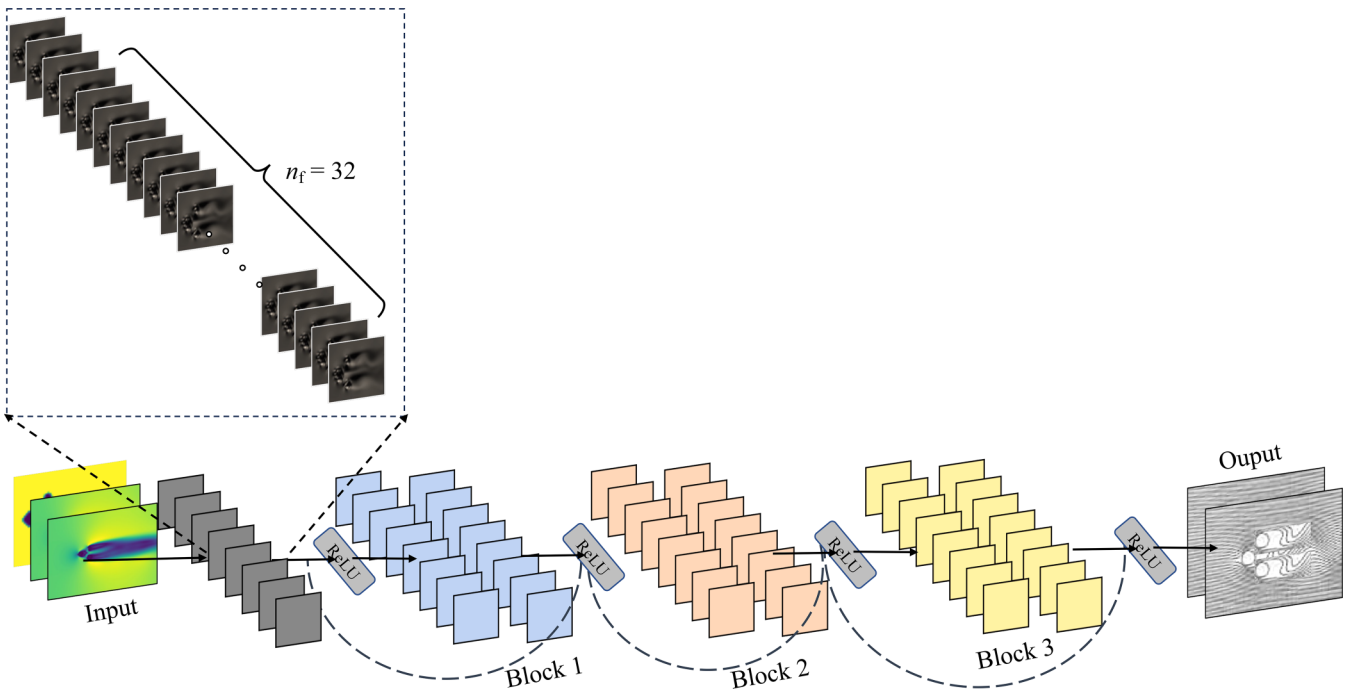

FIG. 43. A residual network architecture with skip connections.

6. ResNeXt

A ResNeXt architecture has been adopted from Ref. [52] that in the original work showed improved performance compared to base ResNet architecture for image classification tasks. The ResNeXt architecture proposed by Ref. [52] comprises a branched design that mimics the *split-transform-merge* strategy on the base ResNet design. The *spitting* is performed once the input is convolved using $n_f = 32$ filters into a number of branches, each having the same topology. The number of branches (or Cardinality) $= 5$ is chosen for the present study. Each branch executes *transformation* on the resulting input by two layers of convolutions followed by Batch Normalisation and ReLU activation. Finally, the outputs from each branch are concatenated to perform *merge* operation. This is followed by another convolution layer which is merged with a skip connection from the input. This operation is repeated 9 times, resulting in a total of 524 322 trainable parameters.

7. Dense convolution network: DenseNet

A dense convolution network (or DenseNet) architecture has been adopted from Ref. [53] that was originally implemented for object recognition tasks. In this architecture, each layer is connected to every other layer in a feed-forward manner, allowing for better feature propagation, supporting feature reuse, and alleviating the vanishing gradient problem. In this architecture, the inputs are initially passed through a convolution layer with $n_f = 64$ filters, followed by a collection of dense layers (or dense blocks) and transition layers, connected in series. In each dense layer, the inputs pass through a bottleneck layer wherein the first convolution layer expands the number of filters to 4×32 and then a second convolution layer containing $n_f = 32$ filters. To further reduce the number of feature maps, the output from the bottleneck layer is passed through the transition layer. There are a total of three Dense layers and 2 transition layers, resulting in a total of 566 578 trainable parameters.

[1] J. N. Kutz, Deep learning in fluid dynamics, J. Fluid Mech. **814**, 1 (2017).

[2] S. L. Brunton, B. R. Noack, and P. Koumoutsakos, Machine learning for fluid mechanics, Annu. Rev. Fluid Mech. **52**, 477 (2020).

[3] J. Ling, A. Kurzawski, and J. Templeton, Reynolds averaged turbulence modelling using deep neural networks with embedded invariance, J. Fluid Mech. **807**, 155 (2016).

[4] J.-L. Wu, H. Xiao, and E. Paterson, Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework, Phys. Rev. Fluids **3**, 074602 (2018).

[5] K. Duraisamy, G. Iaccarino, and H. Xiao, Turbulence modeling in the age of data, Annu. Rev. Fluid Mech. **51**, 357 (2019).

[6] P. A. Srinivasan, L. Guastoni, H. Azizpour, P. Schlatter, and R. Vinuesa, Predictions of turbulent shear flows using deep neural networks, Phys. Rev. Fluids **4**, 054603 (2019).

[7] Y. Xie, E. Franz, M. Chu, and N. Thuerey, tempoGAN: A temporally coherent, volumetric GAN for super-resolution fluid flow, ACM Trans. Graphics (TOG) **37**, 1 (2018).

[8] K. Fukami, K. Fukagata, and K. Taira, Super-resolution reconstruction of turbulent flows with machine learning, J. Fluid Mech. **870**, 106 (2019).

[9] B. Li, Z. Yang, X. Zhang, G. He, B.-Q. Deng, and L. Shen, Using machine learning to detect the turbulent region in flow past a circular cylinder, J. Fluid Mech. **905**, A10 (2020).

[10] F. Pino, L. Schena, J. Rabault, and M. A. Mendez, Comparative analysis of machine learning methods for active flow control, J. Fluid Mech. **958**, A39 (2023).

[11] T. Davydzenka and P. Tahmasebi, High-resolution fluid–particle interactions: A machine learning approach, J. Fluid Mech. **938**, A20 (2022).

[12] D. Xiao, F. Fang, C. Pain, and G. Hu, Non-intrusive reduced-order modelling of the Navier–Stokes equations based on RBF interpolation, Int. J. Numer. Methods Fluids **79**, 580 (2015).

[13] S. Pawar, S. Rahman, H. Vaddireddy, O. San, A. Rasheed, and P. Vedula, A deep learning enabler for nonintrusive reduced order modeling of fluid flows, Phys. Fluids **31**, 085101 (2019).

[14] D. E. Ozan and L. Magri, Hard-constrained neural networks for modeling nonlinear acoustics, Phys. Rev. Fluids **8**, 103201 (2023).

[15] R. Engelken, F. Wolf, and L. F. Abbott, Lyapunov spectra of chaotic recurrent neural networks, Phys. Rev. Res. **5**, 043044 (2023).

[16] C. Xie, X. Xiong, and J. Wang, Artificial neural network approach for turbulence models: A local framework, Phys. Rev. Fluids **6**, 084612 (2021).

[17] L.-W. Chen, B. A. Cakal, X. Hu, and N. Thuerey, Numerical investigation of minimum drag profiles in laminar flow using deep learning surrogates, J. Fluid Mech. **919**, A34 (2021).

[18] M. Raissi, P. Perdikaris, and G. E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys. **378**, 686 (2019).

[19] L. Sun, H. Gao, S. Pan, and J.-X. Wang, Surrogate modeling for fluid flows based on physics-constrained deep learning without simulation data, Comput. Methods Appl. Mech. Eng. **361**, 112732 (2020).

[20] F. D. A. Belbute-Peres, T. Economon, and Z. Kolter, Combining differentiable PDE solvers and graph neural networks for fluid flow prediction, in *Proceedings of the 37th International Conference on Machine Learning*, edited by H. Daumé III and A. Singh (PMLR, 2020), pp. 2402–2411.

[21] J. Brandstetter, D. Worrall, and M. Welling, Message passing neural PDE solvers, arXiv:2202.03376.

[22] M. Cheng, F. Fang, C. C. Pain, and I. Navon, Data-driven modelling of nonlinear spatio-temporal fluid flows using a deep convolutional generative adversarial network, Comput. Methods Appl. Mech. Eng. **365**, 113000 (2020).

[23] S. Lee and D. You, Data-driven prediction of unsteady flow over a circular cylinder using deep learning, J. Fluid Mech. **879**, 217 (2019).

[24] B. Ramsundar, D. Krishnamurthy, and V. Viswanathan, Differentiable physics: A position piece, arXiv:2109.07573.

[25] B. List, L.-W. Chen, and N. Thuerey, Learned turbulence modelling with differentiable fluid solvers: Physics-based loss functions and optimisation horizons, J. Fluid Mech. **949**, A25 (2022).

[26] K. Um, R. Brand, Y. R. Fei, P. Holl, and N. Thuerey, Solver-in-the-loop: Learning from differentiable physics to interact with

iterative PDE-solvers, Adv. Neural Inf. Process. Syst. **33**, 6111 (2020).

[27] D. Kochkov, J. A. Smith, A. Alieva, Q. Wang, M. P. Brenner, and S. Hoyer, Machine learning–accelerated computational fluid dynamics, Proc. Natl. Acad. Sci. USA **118**, e2101784118 (2021).

[28] K. Hasegawa, K. Fukami, T. Murata, and K. Fukagata, Machine-learning-based reduced-order modeling for unsteady flows around bluff bodies of various shapes, Theor. Comput. Fluid Dyn. **34**, 367 (2020).

[29] Z. Y. Wan and T. P. Sapsis, Machine learning the kinematics of spherical particles in fluid flows, J. Fluid Mech. **857**, R2 (2018).

[30] K. Stachenfeld, D. B. Fielding, D. Kochkov, M. Cranmer, T. Pfaff, J. Godwin, C. Cui, S. Ho, P. Battaglia, and A. Sanchez-Gonzalez, Learned coarse models for efficient turbulence simulation, arXiv:2112.15275.

[31] M. Morimoto, K. Fukami, K. Zhang, and K. Fukagata, Generalization techniques of neural networks for fluid flow estimation, Neural Comput. Appl. **34**, 3647 (2022).

[32] G. H. McKinley, R. C. Armstrong, and R. Brown, The wake instability in viscoelastic flow past confined circular cylinders, Philos. Trans. R. Soc. London A **344**, 265 (1993).

[33] C. H. Williamson, Vortex dynamics in the cylinder wake, Annu. Rev. Fluid Mech. **28**, 477 (1996).

[34] P. Bearman, Circular cylinder wakes and vortex-induced vibrations, J. Fluids Struct. **27**, 648 (2011).

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

[36] M. Zdravkovich, The effects of flow interference between two circular cylinders in various arrangements, J. Fluids Struct. **1**, 239 (1987).

[37] G. V. Papaioannou, D. K. Yue, M. S. Triantafyllou, and G. E. Karniadakis, Three-dimensionality effects in flow around two tandem cylinders, J. Fluid Mech. **558**, 387 (2006).

[38] D. Sumner, Two circular cylinders in cross-flow: A review, J. Fluids Struct. **26**, 849 (2010).

[39] K. Lam and W. Cheung, Phenomena of vortex shedding and flow interference of three cylinders in different equilateral arrangements, J. Fluid Mech. **196**, 1 (1988).

[40] D. Guillaume and J. LaRue, Investigation of the flopping regime with two-, three- and four-cylinder arrays, Exp. Fluids **27**, 145 (1999).

[41] Y. Bao, D. Zhou, and C. Huang, Numerical simulation of flow over three circular cylinders in equilateral arrangements at low Reynolds number by a second-order characteristic-based split finite element method, Comput. Fluids **39**, 882 (2010).

[42] S. Zheng, W. Zhang, and X. Lv, Numerical simulation of cross-flow around three equal diameter cylinders in an equilateral-triangular configuration at low Reynolds numbers, Comput. Fluids **130**, 94 (2016).

[43] W. Chen, C. Ji, M. M. Alam, J. Williams, and D. Xu, Numerical simulations of flow past three circular cylinders in equilateral-triangular arrangements, J. Fluid Mech. **891**, A14 (2020).

[44] R. Mittal and G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. **37**, 239 (2005).

[45] S. Brahmachary, Finite volume/immersed boundary solvers for compressible flows: development and applications, Ph.D. thesis, Indian Institute of Technology Guwahati, 2019.

[46] H. Jasak and Z. Tukovic, Immersed boundary method in foam: theory, implementation and use (2015).

[47] R. I. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, J. Comput. Phys. **62**, 40 (1986).

[48] E. Constant, J. Favier, M. Meldi, P. Meliga, and E. Serre, An immersed boundary method in openfoam: Verification and validation, Comput. Fluids **157**, 55 (2017).

[49] A. J. Chorin, Numerical solution of the Navier-Stokes equations, Math. Comput. **22**, 745 (1968).

[50] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980.

[51] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Piscataway, NJ, 2016), pp. 770–778.

[52] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, Aggregated residual transformations for deep neural networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Piscataway, NJ, 2017), pp. 1492–1500.

[53] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, Densely connected convolutional networks, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, Piscataway, NJ, 2017), pp. 4700–4708.

[54] O. Ronneberger, P. Fischer, and T. Brox, U-net: Convolutional networks for biomedical image segmentation, in *Proceedings of the 18th International Conference, on Medical Image Computing and Computer-Assisted Intervention (MICCAI'15)* (Springer, Berlin, 2015), pp. 234–241.

[55] F. Tong, L. Cheng, and M. Zhao, Numerical simulations of steady flow past two cylinders in staggered arrangements, J. Fluid Mech. **765**, 114 (2015).

[56] Y. Bao, D. Zhou, and J. Tu, Flow characteristics of two in-phase oscillating cylinders in side-by-side arrangement, Comput. Fluids **71**, 124 (2013).

[57] P. R. Vlachas, W. Byeon, Z. Y. Wan, T. P. Sapsis, and P. Koumoutsakos, Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks, Proc. R. Soc. A **474**, 20170844 (2018).

[58] N. A. K. Doan, W. Polifke, and L. Magri, Short-and long-term predictions of chaotic flows and extreme events: A physics-constrained reservoir computing approach, Proc. R. Soc. A **477**, 20210135 (2021).

[59] S. Brahmachary, G. Natarajan, V. Kulkarni, N. Sahoo, V. Ashok, and V. Kumar, Role of solution reconstruction in hypersonic viscous computations using a sharp interface immersed boundary method, Phys. Rev. E **103**, 043302 (2021).

[60] S. Lee, J. Yang, P. Forooghi, A. Stroh, and S. Bagheri, Predicting drag on rough surfaces by transfer learning of empirical correlations, J. Fluid Mech. **933**, A18 (2022).

[61] M. A. Jouybari, J. Yuan, G. J. Brereton, and M. S. Murillo, Data-driven prediction of the equivalent sand-grain height in rough-wall turbulent flows, J. Fluid Mech. **912**, A8 (2021).

[62] H. Ozaki and T. Aoyagi, Prediction of steady flows passing fixed cylinders using deep learning, Sci. Rep. **12**, 447 (2022).

[63] https://github.com/tumpbs/DiffPhys-CylinderWakeFlow/.

[64] A. Krizhevsky, I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, Commun. ACM **60**, 84 (2017).