Journal of Computational Physics 396 (2019) 427–450

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Discrete adjoint of fractional-step incompressible Navier-Stokes solver in curvilinear coordinates and application to data assimilation

Mengze Wang, Qi Wang, Tamer A. Zaki *

Department of Mechanical Engineering, Johns Hopkins University, Baltimore, MD 21218, USA

ARTICLE INFO

Article history:
Received 25 January 2019
Received in revised form 3 June 2019
Accepted 27 June 2019
Available online 9 July 2019

Keywords:
Navier-Stokes
Discrete adjoint
Fraction-step algorithm
Generalized coordinates
Data assimilation
Taylor-Couette flow

ABSTRACT

The discrete adjoint of an incompressible Navier-Stokes algorithm in generalized coordinates is derived and applied to estimate the states of saturated and turbulent circular Couette flows. The forward Navier-Stokes model is based on the fractional-step algorithm in curvilinear coordinates on a structured grid [1], which has been widely adopted in direct numerical simulations of transitional and turbulent flows. The discrete adjoint equations adopt the same stencil and temporal scheme as the forward discretization, and expressions are derived that relate the discrete adjoint variables to their continuous counterpart. The key ingredients of the forward algorithm can be retained in the adjoint, including the computation of the cell geometry, the approximate factorization method, and the parallelization strategy. The accuracy, efficiency, and stability of the adjoint solver are also commensurate with the forward model. In addition, a novel symmetric projector is proposed to guarantee that the outcome of the adjoint algorithm is divergence free. The implementation of the algorithm in double precision satisfies the forward-adjoint relation up to eight significant figures, and further validation is performed using circular Couette flow. The forward and adjoint growth rates of instability modes from linear theory are accurately reproduced. In addition, an adjoint-variational data-assimilation algorithm (4DVar) is adopted to reconstruct the initial condition of circular Couette flows from limited measurements, obtained from an independent simulation. When the flow is comprised of saturated wavy vortices, wall measurements are sufficient to reconstruct an initial condition that latches onto the target state after a short time. For the more challenging turbulent case, coarse-grained velocity data are used to estimate the initial condition.

1. Introduction

The chaotic nature of transitional and turbulent flows render them sensitive to initial conditions. As a result, attempts to complement experiments measurements using numerical simulations are fraught with difficulty: small changes in the initial state of the simulations will lead to divergence of the numerical and experimental state at long times. This divergence rate can be characterized by a Lyapunov exponent, which was shown to scale with the Reynolds number in turbulent channel flow [2]. In circular Couette flow, both experiments [3] and direct numerical simulations [4] demonstrated that the wave-

* Corresponding author.
E-mail address: t.zaki@jhu.edu (T.A. Zaki).

https://doi.org/10.1016/j.jcp.2019.06.065
0021-9991/

lengths of turbulent Taylor vortices depend on flow history—a difference that leads to up to 20% variation in torque. Even in the transition regime, a small change in free-stream turbulence intensity can qualitatively change turbulence onset in boundary layer flows from orderly breakdown via Tollmien-Schlichting waves to bypass transition [5–7]. Adjoint-variational data assimilation techniques attempt to address this challenge by constructing accurate initial conditions for numerical simulations, by incorporating measurements. In this work, we derive the discrete-adjoint of a widely-adopted algorithm for solution of the Navier-Stokes equations in generalized curvilinear coordinates. We demonstrate the accuracy of the adjoint model and apply it in a data assimilation problem, where the data are obtained from independent simulations as surrogate for experiments.

1.1. Data assimilation techniques

Three classes of data assimilation have been applied to flow problems: adjoint-variational methods (4DVar), Kalman smoother approach, and ensemble-variational techniques. In all three, the main task is to minimize a cost function, which is proportional to the difference between the true measurements and their estimation from the numerical model. The performance of the three approaches was compared in detail by Mons et al. [8], who demonstrated that 4DVar is most accurate; its only shortcoming is the requirement of developing an adjoint to the forward numerical model. Once the adjoint algorithm is available, 4DVar achieves the largest error reduction given the same computational cost. Another advantage is the physical interpretation of the adjoint variable, namely it is first-order sensitivity of the cost function with respect to change in the initial condition.

Two derivations of the adjoint model can be contrasted: the continuous and discrete approaches. The first approach starts with the continuous forward equations, from which the continuous adjoint equations are derived prior to discretization. The first application in fluid mechanics can perhaps be attributed to Pironneau [9], who derived the continuous adjoint of Stokes equations. The extension to inviscid compressible flows and Reynolds-averaged-Navier-Stokes (RANS) equations was proposed by Jameson et al. [10,11]. The utility of adjoint methods has spanned a wide range of applications, including flow control [12–14] and state estimation [15,8]. In the later context, Bewley and Protas derived the continuous adjoint of the full incompressible Navier-Stokes equations and used it to reconstruct the initial condition of turbulent channel flow [15]; Foures et al. [16] used a similar approach to reconstruct the Reynolds stress. It should be noted, however, that the continuous adjoint algorithm only provides an approximate of the cost function, because the discretization of the continuous adjoint equations might not be consistent with that of the forward schemes [17]. Furthermore, the continuous adjoint equations are only well-posed for certain boundary conditions [18]. Recently Vishnampet et al. demonstrated on the aeroacoustic control of a mixing layer that the continuous adjoint suffers from exponential error growth in time [13].

In the second approach, the equations of the forward model are first discretized and then the adjoint of the discrete form is derived, thus providing an exact gradient of the cost function. The boundary conditions for the discrete adjoint equations are straightforward to derive and implement, compared to those of the continuous variant. Since the first application for quasi-one-dimensional compressible Euler equations [19], this approach has been developed for more complex problems. For example, Mani & Mavriplis derived the discrete adjoint of unsteady compressible Euler equations with deforming meshes [20]; Nielsen et al. extended the formulation to three-dimensional unsteady compressible RANS equations on overset grids [21]; de Pando et al. developed adjoint operators for the linearized Navier-Stokes solvers which are generated by an efficient algorithm that they proposed [22]. Most recently, Vishnampet et al. derived the discrete adjoint of the full compressible Navier-Stokes equations, and demonstrated its robustness in a turbulent flow problem [13]. Previous efforts have not, however, derived the discrete adjoint of the full incompressible Navier-Stokes equations in generalized coordinates.

Automatic differentiation (AD) is another approach to generate an adjoint solver [23]. However, the computer algorithm generated by AD is not sufficiently efficient to apply in large-scale simulations. In addition, Gronskis et al. reported unresolved issues associated with message passing and parallel reduction operations when automatic differentiation is applied to parallel programs [24]. By contrast, each term in derived discrete adjoint algorithm has its counterpart in discrete forward model, which means that the efficiency of adjoint simulation should be close to the forward one and that the parallelization strategy and scalability of the adjoint solver remain unchanged.

In §2, we introduce the adjoint-variational data assimilation algorithm. The continuous formulation is briefly summarized, followed by the forward Navier-Stokes model and the detailed derivation of its discrete adjoint and its boundary conditions. The symmetric operator that is proposed to project the output of the discrete adjoint algorithm onto a divergence-free field is presented. At the end of §2, the gradient-based optimization algorithm is introduced, and the procedures of the data assimilation algorithm are summarized. In §3, the adjoint algorithm is validated and applied for data assimilation in saturated and turbulent circular Couette flows. Finally, conclusions from this work are provided in §4.

2. Discrete-adjoint Navier-Stokes algorithm for data assimilation in generalized coordinates

Adjoint-variational data assimilation, or flow estimation, is formulated as a constrained optimization problem. The unknown parameter is the initial condition, $\boldsymbol{u}^0$. The constraint is that the velocity field, $\boldsymbol{u}$, must satisfy the forward numerical model, $\boldsymbol{u}^{n+1} = \boldsymbol{m}(\boldsymbol{u}^n)$, which advances the state from time step n to $n+1$. Under the constraint, the initial condition must

also reproduce the observation data $\{\boldsymbol{y}^n\}_{n=0}^N$. A cost function is thus defined as the difference between the observations and their estimation from the reconstructed initial condition,

$$J(\boldsymbol{u}^0) = \sum_{n=0}^{N} \frac{1}{2} \|\boldsymbol{y}^n - \boldsymbol{h}(\boldsymbol{u}^n, p^n)\|_O^2, \tag{1}$$

where p^n is the pressure field at the n-th time step, $\boldsymbol{h}$ is the observation operator, and $\|\cdot\|_O$ represents evaluation over the observation space. In adjoint-variational data assimilation, the adjoint model is invoked to evaluate the gradient of the cost function. In order to ease the mathematical and physical interpretation of the algorithm, the continuous adjoint is introduced first, followed by derivation of the discrete adjoint algorithm that guarantees accurate evaluation of the gradient of the cost function.

2.1. Continuous form of adjoint variational data assimilation

The evolution of unsteady incompressible flows are governed by the continuity and momentum equations,

$$\nabla \cdot \boldsymbol{u} = 0 \tag{2}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot (\boldsymbol{u}\boldsymbol{u}) = -\nabla p + \frac{1}{Re}\nabla^2 \boldsymbol{u}. \tag{3}$$

These equations are also referred to as the forward model in variational techniques, and $\boldsymbol{u}, p$ as the forward variables. Minimization of the cost function under equality constraints (2)-(3) is equivalent to unconstrained minimization of the Lagrangian function,

$$L = J - \left\langle \boldsymbol{u}^\dagger, \frac{\partial \boldsymbol{u}}{\partial t} + \cdot\nabla(\boldsymbol{u}\boldsymbol{u}) + \nabla p - \frac{1}{Re}\nabla^2 \boldsymbol{u} \right\rangle - \left\langle p^\dagger, \nabla \cdot \boldsymbol{u} \right\rangle. \tag{4}$$

The Lagrangian multipliers $\boldsymbol{u}^\dagger$ and $p^\dagger$ are the adjoint velocity and adjoint pressure, respectively., and the inner product is defined as integration over the whole computational domain and time horizon,

$$\langle \boldsymbol{f}, \boldsymbol{g} \rangle = \int_{t_s}^{t_e} \int_V \boldsymbol{f}^T \boldsymbol{g}\, d^3\boldsymbol{x} dt, \tag{5}$$

where t_s and t_e represent the start and end of the assimilation window. The first-order optimality condition requires the derivatives of L with respect to both adjoint and forward variables to be zero; the first generates the continuous forward equations (2)-(3), and the latter leads to their adjoints,

$$\nabla \cdot \boldsymbol{u}^\dagger = \frac{\partial J}{\partial p} \tag{6}$$

$$\frac{\partial \boldsymbol{u}^\dagger}{\partial t^\dagger} - \boldsymbol{u} \cdot \nabla \boldsymbol{u}^\dagger - \boldsymbol{u} \cdot (\nabla \boldsymbol{u}^\dagger)^T = \nabla p^\dagger + \frac{1}{Re}\nabla^2 \boldsymbol{u}^\dagger + \frac{\partial J}{\partial \boldsymbol{u}}, \tag{7}$$

where $t^\dagger \equiv t_e - t$. A few remarks are in order in relation to the adjoint equations: Firstly, if the pressure field contributes to the observations, the adjoint velocity field is not divergence free. Secondly, the adjoint momentum equations are marched backwards in time, from $t = t_e$ (or $t^\dagger = t_s$) to $t = t_s$ (or $t^\dagger = t_e$), and in backward time adjoint diffusion is well behaved. Thirdly, the adjoint momentum equations include the forward advection velocity, which is reversed in direction, and are forced by the derivatives of the cost function. Lastly, evaluation of the adjoint advection term requires the time history of the forward field $\boldsymbol{u}$. One option is to march forward equations from $t = t_s$ to $t = t_e$ and store $\boldsymbol{u}$ at every time step. The associated storage cost can be restrictive for large problems since it scales with the number of grid points and time steps. An alternative is the checkpointing scheme [25]: In this approach, the forward velocities are stored at large intervals, called checkpoints, and forward simulations are performed between checkpoints to generate the required $\boldsymbol{u}$ fields during adjoint marching.

Note that starting from the non-conservative form of the forward momentum equations (3), and adopting the same definition of the Lagrangian function (4), the adjoint momentum equations (7) take a different form,

$$\frac{\partial \boldsymbol{u}^\dagger}{\partial t^\dagger} - \boldsymbol{u} \cdot \nabla \boldsymbol{u}^\dagger + \boldsymbol{u}^\dagger \cdot (\nabla \boldsymbol{u})^T = \nabla p_{nc}^\dagger + \frac{1}{Re}\nabla^2 \boldsymbol{u}^\dagger + \frac{\partial J}{\partial \boldsymbol{u}}. \tag{8}$$

The third term on the left-hand side now behaves as a source. Equations (8) and (7) are in fact linked by the transformation $p^\dagger = p_{nc}^\dagger - \boldsymbol{u} \cdot \boldsymbol{u}^\dagger$.

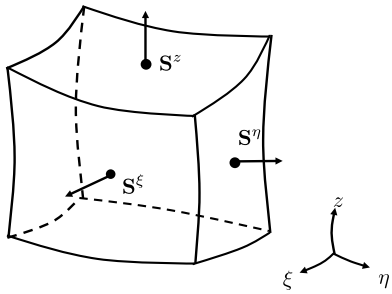

Fig. 1. A computational cell in generalized coordinates.

With the forward and adjoint equations, the general procedures of the adjoint variational approach can be outlined as: (i) Establish a guess of the initial condition $\boldsymbol{u}^0$, advance the forward equations (2)-(3) and store the forward fields. (ii) Compute the adjoint evolution (6)-(7) and evaluate the gradient of the cost function. (iii) Apply a projector to the adjoint field in order to ensure that the search for the initial condition yields a solenoidal state. (iv) Implement a gradient-based algorithm to update the next initial condition. Although there are many ways to discretize the adjoint equations (6) and (7), only the discrete adjoint provides the exact gradient of the cost function. In the next subsections, the discrete forward and adjoint equations are derived, followed by a detailed presentation of steps (iii) and (iv) in the above procedure.

2.2. Discrete Navier-Stokes equations

The incompressible Navier-Stokes equations are solved using a fractional-step algorithm with a local volume flux formulation on a structured and staggered grid [1]. Details of the algorithm are not reproduced here; only the key elements are presented in order to provide the context for the derivation of the discrete adjoint. In the fractional step approach, instead of equations (2)-(3), a momentum equation for an intermediate velocity $\hat{\boldsymbol{u}}$ is solved first,

$$\frac{\hat{\boldsymbol{u}}-\boldsymbol{u}^n}{\Delta t} = -\left(a_1\nabla\cdot(\boldsymbol{u}\boldsymbol{u})^n - a_2\nabla\cdot(\boldsymbol{u}\boldsymbol{u})^{n-1}\right) - \nabla p^n + \frac{1}{Re}\nabla^2\left(b_1\boldsymbol{u}^n + b_2\hat{\boldsymbol{u}}\right). \tag{9}$$

As a result, $\hat{\boldsymbol{u}}$ does not satisfy the divergence-free constraint. Here the weighting coefficients a_1, a_2, b_1, b_2 depend on the numerical scheme adopted ($a_1 = 3/2, a_2 = 1/2$ for Adams-Bashforth, $a_1 = 1, a_2 = 0$ for forward Euler, and $b_1 = b_2 = 1/2$ for Crank-Nicolson). In (9) the pressure gradient is also included, which is referred to as the Δp-form of the equation. Subsequently, an elliptic equation for pressure increment $\phi^{n+1} \equiv p^{n+1} - p^n$ is solved with the source term proportional to the divergence of the intermediate velocity,

$$\nabla^2\phi^{n+1} = -\frac{\nabla\cdot\hat{\boldsymbol{u}}}{\Delta t}. \tag{10}$$

Finally, this pressure increment is used to project the intermediate velocity onto a divergence-free field,

$$\frac{\boldsymbol{u}^{n+1}-\hat{\boldsymbol{u}}}{\Delta t} = -\nabla\phi^{n+1} \tag{11}$$

and update the pressure field $p^{n+1} = p^n + \phi^{n+1}$.

In order to faciliate the application in generalized coordinate, the volume fluxes u^q, $q = \xi, \eta, z$, are adopted as the state variables, and are related to the velocities using the transformation,

$$\boldsymbol{u} = \mathbf{S}_\xi u^\xi + \mathbf{S}_\eta u^\eta + \mathbf{S}_z u^z \tag{12}$$

$$\mathbf{S}^q\cdot\mathbf{S}_m = \delta^q_m \tag{13}$$

where $\mathbf{S}_m$ is the covariant basis, and $\mathbf{S}^q$ is the reciprocal (contravariant) basis. For a computational cell (Fig. 1), the east, west, north, south, front and back faces are denoted e, w, n, s, f and b, respectively. The volume of a primary cell is V_p and the volume of a staggered cell is V_q (the dashed and dashed-dot squares in Fig. 2).

With the transformation (12) and all the implicit components moved to the left-hand side, equation (9) can be written as,

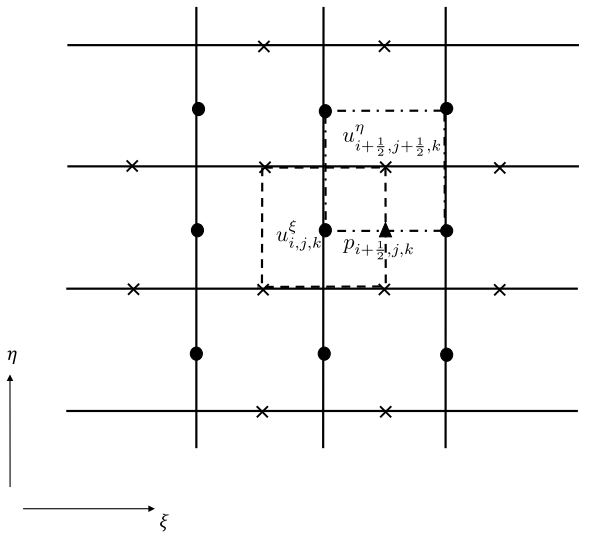

Fig. 2. Staggered grid arrangement. Solid lines are the faces of primary computational cells, and pressure is evaluated at the cell center (the triangular mark). The dashed square and solid circles are the faces and center of a u^{ξ} computational cell. The dashed-dot square and cross marks correspond to a u^{η} cell. All the solid circles and cross marks in this figure form a stencil for advancing $u^{\xi}_{i,j,k}$.

$$\begin{aligned}(I - b_2\frac{\Delta t}{V_\xi}D_\xi)[(\hat{u}^\xi)^n - (u^\xi)^n] = &\frac{\Delta t}{V_\xi}[a_1 H_\xi(u^q)^n - a_2 H_\xi(u^q)^{n-1}] + \frac{\Delta t}{V_\xi}R_\xi(p^n)\\ &+ \frac{\Delta t}{V_\xi}[a_1 D_{\xi,ex}(u^q)^n - a_2 D_{\xi,ex}(u^q)^{n-1}]\\ &+ (b_1+b_2)\frac{\Delta t}{V_\xi}D_\xi(u^\xi)^n\end{aligned} \tag{14}$$

where H, R, D_{ex}, D are the advection, pressure gradient, explicit diffusion and implicit diffusion operators (see equation (18)-(21) and [1] for details). Fig. 2 presents a stencil for computing all these terms. Similar expressions can be written for $\hat{u}^\eta$ and $\hat{u}^z$. The elliptic equation for pressure increment (10) is discretized as,

$$\begin{aligned}&\frac{R_{\xi,e}(\phi)^{n+1}}{V_{\xi,e}} - \frac{R_{\xi,w}(\phi)^{n+1}}{V_{\xi,w}} + \frac{R_{\eta,n}(\phi)^{n+1}}{V_{\eta,n}} - \frac{R_{\eta,s}(\phi)^{n+1}}{V_{\eta,s}}\\ &\quad + \frac{R_{z,f}(\phi)^{n+1}}{V_{z,f}} - \frac{R_{z,b}(\phi)^{n+1}}{V_{z,b}} = -\frac{\hat{u}^\xi_e - \hat{u}^\xi_w + \hat{u}^\eta_n - \hat{u}^\eta_s + \hat{u}^z_f - \hat{u}^z_b}{\Delta t},\end{aligned} \tag{15}$$

and its solution is followed by the projection step and pressure update,

$$(u^\xi)^{n+1} - (\hat{u}^\xi)^n = \frac{\Delta t}{V_\xi}R_\xi(\phi)^{n+1}, \tag{16}$$

$$p^{n+1} = p^n + \phi^{n+1}. \tag{17}$$

In the following expressions (18)-(21), (i, j, k) represents the center of a u^ξ computational cell (as shown in Fig. 2). The advection term is given by,

$$\begin{aligned}H_\xi(u^q) = -\mathbf{S}^\xi_{i,j,k}\cdot\Big[&(u^\xi u^m \mathbf{S}_m)_{i+\frac{1}{2},j,k} - (u^\xi u^m \mathbf{S}_m)_{i-\frac{1}{2},j,k} + (u^\eta u^m \mathbf{S}_m)_{i,j+\frac{1}{2},k}\\ &- (u^\eta u^m \mathbf{S}_m)_{i,j-\frac{1}{2},k} + (u^z u^m \mathbf{S}_m)_{i,j,k+\frac{1}{2}} - (u^z u^m \mathbf{S}_m)_{i,j,k-\frac{1}{2}}\Big]\end{aligned} \tag{18}$$

where m represents summation over ξ, η and z. The diffusion terms are split into explicit and implicit parts,

$$\frac{1}{V_\xi}\left(D_\xi + D_{\xi,ex}\right) = \frac{\mathbf{S}^\xi}{V_\xi Re}\cdot\sum_l\left(\mathbf{S}^l\cdot\left(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T\right)_l\right) \tag{19}$$

where l refers to the center of six faces of the u^ξ-momentum cell, and $\mathbf{S}^l$ is the normal vector of each face. The velocity graident $\nabla\mathbf{u}_{i,j,k}$ is expressed in terms of fluxes according to,

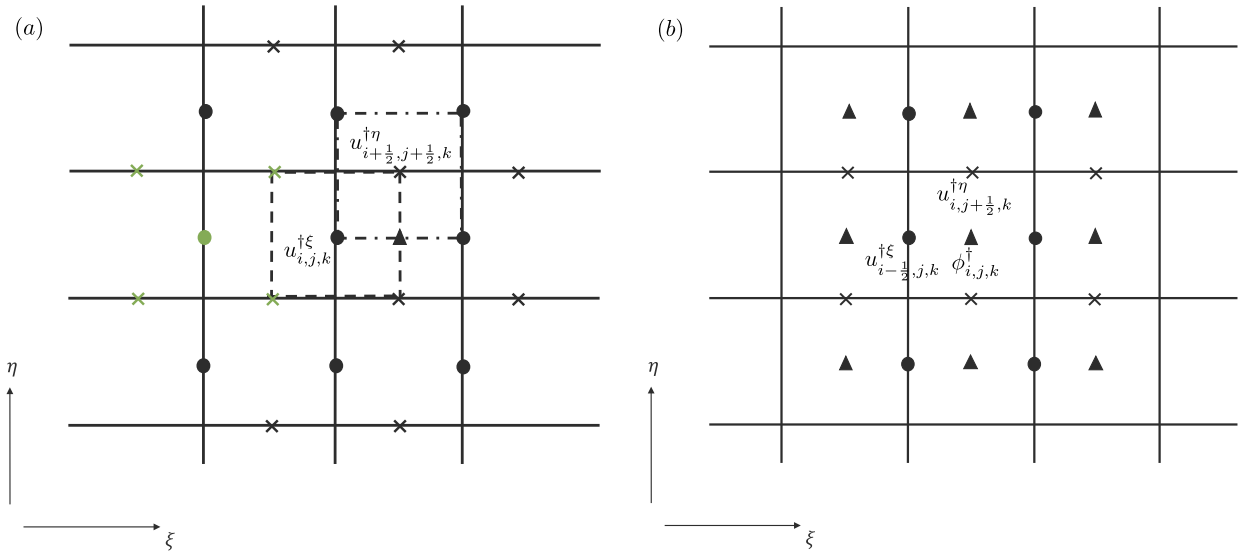

Fig. 3. Grid configuration for the discrete adjoint equations. The definition of lines and symbols are the same as in Fig. 2. (a) Stencil for advancing $u^{\dagger\xi}$, where (i, j, k) mark the center of a $u^{\dagger\xi}$ cell. The green circle is the location to evaluate the sample term in the adjoint diffusion operator (37)-(39). The $u^{\dagger\eta}$ component (38) at this location is computed by averaging the four nearby locations marked by the green cross marks. (b) Stencil for solving the adjoint Poisson equation (30), where (i, j, k) represents the center of a pressure cell. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

$$\begin{aligned}\nabla \boldsymbol{u}_{i,j,k} = \frac{1}{V_{i,j,k}}\Big(&\mathbf{S}^{\xi}_{i+\frac{1}{2}}\mathbf{S}_{m,i+\frac{1}{2}}u^{m}_{i+\frac{1}{2}} - \mathbf{S}^{\xi}_{i-\frac{1}{2}}\mathbf{S}_{m,i-\frac{1}{2}}u^{m}_{i-\frac{1}{2}} + \mathbf{S}^{\eta}_{j+\frac{1}{2}}\mathbf{S}_{m,j+\frac{1}{2}}u^{m}_{j+\frac{1}{2}}\\ &- \mathbf{S}^{\eta}_{j-\frac{1}{2}}\mathbf{S}_{m,j-\frac{1}{2}}u^{m}_{j-\frac{1}{2}} + \mathbf{S}^{z}_{k+\frac{1}{2}}\mathbf{S}_{m,k+\frac{1}{2}}u^{m}_{k+\frac{1}{2}} - \mathbf{S}^{z}_{k-\frac{1}{2}}\mathbf{S}_{m,k-\frac{1}{2}}u^{m}_{k-\frac{1}{2}}\Big).\end{aligned} \tag{20}$$

Detailed expressions for the implicit and explicit diffusion terms were provided by Rosenfeld et al. [26]. Due to the splitting, the momentum equation for $\hat{u}$ (14) can be solved by approximate factorization method [1]. The pressure gradient term is,

$$\begin{aligned}R_{\xi,i,j,k}(p) = -\mathbf{S}^{\xi}_{i,j,k}\cdot\Big[&(\mathbf{S}^{\xi}p)_{i+\frac{1}{2},j,k} - (\mathbf{S}^{\xi}p)_{i-\frac{1}{2},j,k} + (\mathbf{S}^{\eta}p)_{i,j+\frac{1}{2},k}\\ &- (\mathbf{S}^{\eta}p)_{i,j-\frac{1}{2},k} + (\mathbf{S}^{z}p)_{i,j,k+\frac{1}{2}} - (\mathbf{S}^{z}p)_{i,j,k-\frac{1}{2}}\Big].\end{aligned} \tag{21}$$

For circular Couette flow, periodic boundary conditions are enforced in the azimuthal and axial directions, and no-slip boundary conditions are imposed at both walls (see §2.4 for details). The pressure Poisson equation (21) is solved using Fourier Transform in the periodic directions, and a tri-diagonal inversion in the wall-normal coordinate. Other options such as Fourier-cosine transformation and multigrid method for non-periodic configurations are also available in the forward implementation [27–29].

2.3. Discrete adjoint equations

For simplicity, the difference between the left- and right-hand sides in equations (14)-(17) are represented by $F1$ through $F4$ in the following derivation. The discrete Lagrangian is defined as,

$$\begin{aligned}L = J &- \left\langle (\hat{u}^{\dagger q})^{n}, (\mathrm{F1})\right\rangle - \left\langle (\phi^{\dagger})^{n+1}, (\mathrm{F2})\right\rangle - \left\langle (u^{\dagger q})^{n+1}, (\mathrm{F3})\right\rangle\\ &- \left\langle (p^{\dagger})^{n+1}, (\mathrm{F4})\right\rangle,\end{aligned} \tag{22}$$

where $q = \xi, \eta, z$. The definition of the inner product is similar to (5), except that the volume and time integrals are replaced with vector product over the grid cells and instantaneous fields, respectively,

$$\left\langle f^{n}, g^{n}\right\rangle = \sum_{n=0}^{N-1}\sum_{i,j,k} f^{n}_{i,j,k}\, g^{n}_{i,j,k}. \tag{23}$$

In this manner, the adjoint variables directly represent the sensitivity of the Lagrangian function to the constraints and the initial condition.

It is important to note that the continuous and discrete adjoint variables are not identical, but they can be related to one another by comparing their respective definitions in the continuous (4) and discrete Lagrangian (22). For clarity, we introduce subscript c to mark the continuous adjoint variables $\boldsymbol{u}_c^\dagger$ and $p_c^\dagger$.

$$u^{\dagger\xi} = V_\xi \mathbf{S}_\xi \cdot \boldsymbol{u}_c^\dagger - \Delta t(p_{c,e}^\dagger - p_{c,w}^\dagger) \tag{24}$$

$$u^{\dagger\eta} = V_\eta \mathbf{S}_\eta \cdot \boldsymbol{u}_c^\dagger - \Delta t(p_{c,n}^\dagger - p_{c,s}^\dagger) \tag{25}$$

$$u^{\dagger z} = V_z \mathbf{S}_z \cdot \boldsymbol{u}_c^\dagger - \Delta t(p_{c,f}^\dagger - p_{c,b}^\dagger) \tag{26}$$

The continuous adjoint pressure $p_c^\dagger$ appears in (24)-(26) because it is associated with mass conservation, whereas the discrete adjoint pressure is associated with the Poisson equation. These relations are important in instances when an initial condition to the adjoint simulation is provided, for example from adjoint linear-stability theory [30]. In this particular example, the continuous variable must be recast in terms of the discrete adjoint variable prior to marching. This connection will be used in §3 where, for validation of our algorithm, we reproduce the growth rates of forward and adjoint stability modes that are predicted using linear theory of circular Couette flow.

By setting the derivative of L with respect to each of the forward variables to zero, we obtain the following discrete adjoint equations,

$$\begin{aligned} &R_{\xi,w}^\dagger(\phi^\dagger)^{n+1} - R_{\xi,e}^\dagger(\phi^\dagger)^{n+1} + R_{\eta,n}^\dagger(\phi^\dagger)^{n+1} - R_{\eta,s}^\dagger(\phi^\dagger)^{n+1} + R_{z,f}^\dagger(\phi^\dagger)^{n+1} \\ &- R_{z,b}^\dagger(\phi^\dagger)^{n+1} = \Delta t\left(R_\xi^\dagger(u^{\dagger\xi})^{n+1} + R_\eta^\dagger(u^{\dagger\eta})^{n+1} + R_z^\dagger(u^{\dagger z})^{n+1}\right) - (p^\dagger)^{n+1}, \end{aligned} \tag{27}$$

$$(I - b_2 A_\xi^\dagger)(\hat{u}^{\dagger\xi})^n - (u^{\dagger\xi})^{n+1} = \frac{1}{\Delta t}\left((\phi_w^\dagger)^{n+1} - (\phi_e^\dagger)^{n+1}\right), \tag{28}$$

$$\begin{aligned} (u^{\dagger\xi})^n =& (I - b_2\Delta t D_\xi^\dagger)(\hat{u}^{\dagger\xi})^n \\ &+ \Delta t\left(a_1 H_\xi^\dagger\left((\hat{u}^{\dagger q})^n, (u^q)^n\right) - a_2 H_\xi^\dagger\left((\hat{u}^{\dagger q})^{n+1}, (u^q)^{n+1}\right)\right) \\ &+ \Delta t\left(a_1 D_{\xi,ex}^\dagger(\hat{u}^{\dagger q})^n - a_2 D_{\xi,ex}^\dagger(\hat{u}^{\dagger q})^{n+1}\right) \\ &+ (b_1 + b_2)\Delta t D_\xi^\dagger(\hat{u}^{\dagger\xi})^n + \frac{\partial J}{\partial (u^\xi)^n}, \end{aligned} \tag{29}$$

$$(p^\dagger)^n = (p^\dagger)^{n+1} - \Delta t R_\xi^\dagger(\hat{u}^{\dagger\xi})^n - \Delta t R_\eta^\dagger(\hat{u}^{\dagger\eta})^n - \Delta t R_z^\dagger(\hat{u}^{\dagger z})^n - \frac{\partial J}{\partial p^n}. \tag{30}$$

The terms $H^\dagger$, $R^\dagger$, $D_{ex}^\dagger$ and $D^\dagger$ are the adjoints to the forward counterparts (H, R, D_{ex} and D). The time discretization schemes in (29) are the same as in the forward equations. The procedure for deriving the discrete adjoint operators is demonstrated by a simple example, using the inviscid Burgers equation, in Appendix A. Due to the symmetry in the discrete forward operators, only about one ninth of the adjoint terms need to be derived explicitly, and the others terms can be easily deduced by change of indices.

First we consider a cell with indices (i, j, k), centered around a $u^{\dagger\xi}$ node, and provide expressions for $H_\xi^\dagger$ and $D_\xi^\dagger$. The stencil is shown in Fig. 3(a). The adjoint advection term, $H_\xi^\dagger$, has three contributions,

$$H_\xi^\dagger = -H_{\xi,1}^\dagger(u^{\dagger\xi}, u^q) - H_{\xi,2}^\dagger(u^{\dagger\eta}, u^q) - H_{\xi,3}^\dagger(u^{\dagger z}, u^q) \tag{31}$$

and each arises due to one of the three forward advection operators, H_ξ, H_η and H_z. The first contribution is

$$\begin{aligned} H_{\xi,1}^\dagger = &\sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}}\left[\left(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V}\right)_{\alpha-\frac{1}{2},j,k} - \left(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V}\right)_{\alpha+\frac{1}{2},j,k}\right]\cdot\frac{1}{2}\left[(u^\xi\mathbf{S}_\xi)_{\alpha,j,k} + (u^m\mathbf{S}_m)_{\alpha,j,k}\right] \\ &+ \sum_{\beta=j-\frac{1}{2}}^{j+\frac{1}{2}}\left[\left(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V}\right)_{i,\beta-\frac{1}{2},k} - \left(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V}\right)_{i,\beta+\frac{1}{2},k}\right]\cdot\frac{1}{2}(u^\eta\mathbf{S}_\xi)_{i,\beta,k} \\ &+ \sum_{\gamma=k-\frac{1}{2}}^{k+\frac{1}{2}}\left[\left(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V}\right)_{i,j,\gamma-\frac{1}{2}} - \left(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V}\right)_{i,j,\gamma+\frac{1}{2}}\right]\cdot\frac{1}{2}(u^z\mathbf{S}_\xi)_{i,j,\gamma}; \end{aligned} \tag{32}$$

the second is,

$$
\begin{aligned}
H^{\dagger}_{\xi,2} = & \sum_{\beta=j-\frac{1}{2}}^{j+\frac{1}{2}} \Big[(\frac{u^{\dagger\eta}\mathbf{S}^{\eta}}{V})_{i-\frac{1}{2},\beta,k} - (\frac{u^{\dagger\eta}\mathbf{S}^{\eta}}{V})_{i+\frac{1}{2},\beta,k} \Big] \cdot \frac{1}{2} \Big[(u^{\xi}\mathbf{S}_{\xi})_{i,\beta,k} + (u^{m}\mathbf{S}_{m})_{i,\beta,k} \Big] \\
& + \sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}} \Big[(\frac{u^{\dagger\eta}\mathbf{S}^{\eta}}{V})_{\alpha,j-\frac{1}{2},k} - (\frac{u^{\dagger\eta}\mathbf{S}^{\eta}}{V})_{\alpha,j+\frac{1}{2},k} \Big] \cdot \frac{1}{2} (u^{\eta}\mathbf{S}_{\xi})_{\alpha,\beta,k} \\
& + \sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}} \sum_{\beta=j-\frac{1}{2}}^{j+\frac{1}{2}} \sum_{\gamma=k-\frac{1}{2}}^{k+\frac{1}{2}} \Big[(\frac{u^{\dagger\eta}\mathbf{S}^{\eta}}{V})_{\alpha,\beta,\gamma-\frac{1}{2}} - (\frac{u^{\dagger\eta}\mathbf{S}^{\eta}}{V})_{\alpha,\beta,\gamma+\frac{1}{2}} \Big] \cdot \frac{1}{8} \left(u^{z}\mathbf{S}_{\xi}\right)_{\alpha,\beta,\gamma} ;
\end{aligned}
\tag{33}
$$

and the third part $H^{\dagger}_{\xi,3}$ can be derived by switching η with z, j with k, and β with γ in $H^{\dagger}_{\xi,2}$:

$$
\begin{aligned}
H^{\dagger}_{\xi,3} = & \sum_{\gamma=k-\frac{1}{2}}^{k+\frac{1}{2}} \Big[(\frac{u^{\dagger z}\mathbf{S}^{z}}{V})_{i-\frac{1}{2},j,\gamma} - (\frac{u^{\dagger z}\mathbf{S}^{z}}{V})_{i+\frac{1}{2},j,\gamma} \Big] \cdot \frac{1}{2} \Big[(u^{\xi}\mathbf{S}_{\xi})_{i,j,\gamma} + (u^{m}\mathbf{S}_{m})_{i,j,\gamma} \Big] \\
& + \sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}} \Big[(\frac{u^{\dagger z}\mathbf{S}^{z}}{V})_{\alpha,j,k-\frac{1}{2}} - (\frac{u^{\dagger z}\mathbf{S}^{z}}{V})_{\alpha,j,k+\frac{1}{2}} \Big] \cdot \frac{1}{2} (u^{z}\mathbf{S}_{\xi})_{\alpha,j,k} \\
& + \sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}} \sum_{\beta=j-\frac{1}{2}}^{j+\frac{1}{2}} \sum_{\gamma=k-\frac{1}{2}}^{k+\frac{1}{2}} \Big[(\frac{u^{\dagger z}\mathbf{S}^{z}}{V})_{\alpha,\beta-\frac{1}{2},\gamma} - (\frac{u^{\dagger z}\mathbf{S}^{z}}{V})_{\alpha,\beta+\frac{1}{2},\gamma} \Big] \cdot \frac{1}{8} \left(u^{\eta}\mathbf{S}_{\xi}\right)_{\alpha,\beta,\gamma} .
\end{aligned}
\tag{34}
$$

The adjoint of the forward diffusion terms (19) can be expressed as,

$$
D^{\dagger}_{\xi} + D^{\dagger}_{\xi,ex} = -D^{\dagger}_{e} + D^{\dagger}_{w} - D^{\dagger}_{n} + D^{\dagger}_{s} - D^{\dagger}_{f} - D^{\dagger}_{b},
\tag{35}
$$

where the splitting of the terms into explicit and implicit treatments is chosen to mirror that in the forward terms (19). The term $D^{\dagger}_{w}$ is given by,

$$
\begin{aligned}
D^{\dagger}_{w} = & \frac{1}{Re} \sum_{m=\xi,\eta,z} \Bigg\{ \Big[\frac{u^{\dagger m}_{\alpha}\mathbf{S}^{m}_{\alpha}}{V_{\alpha}} \cdot \Big(\frac{\mathbf{S}^{\xi}_{\alpha+\frac{1}{2}}}{V_{\alpha+\frac{1}{2}}} \cdot (\mathbf{S}^{\xi}\mathbf{S}_{\xi} + \mathbf{S}_{\xi}\mathbf{S}^{\xi})_{\alpha+1} \Big) \Big]_{i-1} \\
& - \Big[\frac{u^{\dagger m}_{\alpha}\mathbf{S}^{m}_{\alpha}}{V_{\alpha}} \cdot \Big(\frac{\mathbf{S}^{\xi}_{\alpha+\frac{1}{2}}}{V_{\alpha+\frac{1}{2}}} \cdot (\mathbf{S}^{\xi}\mathbf{S}_{\xi} + \mathbf{S}_{\xi}\mathbf{S}^{\xi})_{\alpha} \Big) \Big]_{i} \\
& + \Big[\frac{u^{\dagger m}_{\alpha,\beta}\mathbf{S}^{m}_{\alpha,\beta}}{V_{\alpha,\beta}} \cdot \Big(\frac{\mathbf{S}^{\xi}_{\alpha+\frac{1}{2},\beta}}{V_{\alpha+\frac{1}{2},\beta}} \cdot (\mathbf{S}^{\eta}\mathbf{S}_{\xi} + \mathbf{S}_{\xi}\mathbf{S}^{\eta})_{\alpha+\frac{1}{2},\beta+\frac{1}{2}} \Big) \Big]_{i-\frac{1}{2},j-\frac{1}{2}} \\
& - \Big[\frac{u^{\dagger m}_{\alpha,\beta}\mathbf{S}^{m}_{\alpha,\beta}}{V_{\alpha,\beta}} \cdot \Big(\frac{\mathbf{S}^{\xi}_{\alpha+\frac{1}{2},\beta}}{V_{\alpha+\frac{1}{2},\beta}} \cdot (\mathbf{S}^{\eta}\mathbf{S}_{\xi} + \mathbf{S}_{\xi}\mathbf{S}^{\eta})_{\alpha+\frac{1}{2},\beta-\frac{1}{2}} \Big) \Big]_{i-\frac{1}{2},j+\frac{1}{2}} \\
& + \Big[\frac{u^{\dagger m}_{\alpha,\gamma}\mathbf{S}^{m}_{\alpha,\gamma}}{V_{\alpha,\gamma}} \cdot \Big(\frac{\mathbf{S}^{\xi}_{\alpha+\frac{1}{2},\gamma}}{V_{\alpha+\frac{1}{2},\gamma}} \cdot (\mathbf{S}^{z}\mathbf{S}_{\xi} + \mathbf{S}_{\xi}\mathbf{S}^{z})_{\alpha+\frac{1}{2},\gamma+\frac{1}{2}} \Big) \Big]_{i-\frac{1}{2},k-\frac{1}{2}} \\
& - \Big[\frac{u^{\dagger m}_{\alpha,\gamma}\mathbf{S}^{m}_{\alpha,\gamma}}{V_{\alpha,\gamma}} \cdot \Big(\frac{\mathbf{S}^{\xi}_{\alpha+\frac{1}{2},\gamma}}{V_{\alpha+\frac{1}{2},\gamma}} \cdot (\mathbf{S}^{z}\mathbf{S}_{\xi} + \mathbf{S}_{\xi}\mathbf{S}^{z})_{\alpha+\frac{1}{2},\gamma-\frac{1}{2}} \Big) \Big]_{i-\frac{1}{2},k+\frac{1}{2}} \Bigg\}.
\end{aligned}
\tag{36}
$$

The square brackets $[...]_{(i,j,k)}$ indicate that the enclosed quantity is evaluated at the location (i, j, k). Due to the control-volume form, the term may need to be evaluated from averaging over faces, and hence (α, β, γ) are introduced to take the appropriate values of (i, j, k), respectively. Consider the term in the first brackets as an example, which is expressed for simplicity using the shorthand $[u^{\dagger m}C_{\alpha}]_{i-1}$. When $m = \xi$, since $u^{\dagger\xi}$ is stored at the $i-1$ node (green circle in Fig. 3(a)), the term $[u^{\dagger\xi}C_{\alpha}]_{i-1}$ is directly computed at that location,

$$
[u^{\dagger\xi}C_{\alpha}]_{i-1} = u^{\dagger\xi}_{i-1}C_{i-1}.
\tag{37}
$$

When $m=\eta$, since $u^{\dagger\eta}$ is not stored at the $i-1$ node, $[u^{\dagger\eta}C_\alpha]_{i-1}$ is evaluated from the nearby $u^{\dagger\eta}$ nodes (four green cross marks in Fig. 3),

$$\begin{aligned}[u^{\dagger\eta}C_\alpha]_{i-1}=&\frac{1}{4}\Big(u^{\dagger\eta}_{i-\frac{3}{2},j+\frac{1}{2}}C_{i-\frac{3}{2},j+\frac{1}{2}}+u^{\dagger\eta}_{i-\frac{3}{2},j-\frac{1}{2}}C_{i-\frac{3}{2},j-\frac{1}{2}}\\&+u^{\dagger\eta}_{i-\frac{1}{2},j+\frac{1}{2}}C_{i-\frac{1}{2},j+\frac{1}{2}}+u^{\dagger\eta}_{i-\frac{1}{2},j-\frac{1}{2}}C_{i-\frac{1}{2},j-\frac{1}{2}}\Big).\end{aligned}\tag{38}$$

Similarly, the $[u^{\dagger z}C_\alpha]_{i-1}$ is also calculated from the nearby $u^{\dagger z}$ nodes,

$$\begin{aligned}[u^{\dagger z}C_\alpha]_{i-1}=&\frac{1}{4}\Big(u^{\dagger z}_{i-\frac{3}{2},k+\frac{1}{2}}C_{i-\frac{3}{2},k+\frac{1}{2}}+u^{\dagger z}_{i-\frac{3}{2},k+\frac{1}{2}}C_{i-\frac{3}{2},k+\frac{1}{2}}\\&+u^{\dagger z}_{i-\frac{1}{2},k+\frac{1}{2}}C_{i-\frac{1}{2},k+\frac{1}{2}}+u^{\dagger z}_{i-\frac{1}{2},k-\frac{1}{2}}C_{i-\frac{1}{2},k-\frac{1}{2}}\Big).\end{aligned}\tag{39}$$

The term $D_e^\dagger$ has a similar form to (36), with i replaced by $i+1$ and α replaced by $\alpha-1$. The term $D_s^\dagger$ is given by,

$$\begin{aligned}D_s^\dagger=&\frac{1}{Re}\sum_{m=\xi,\eta,z}\Bigg\{\Big[\frac{u^{\dagger m}_{\alpha,\beta}\mathbf{S}^m_{\alpha,\beta}}{V_{\alpha,\beta}}\cdot\Big(\frac{\mathbf{S}^\eta_{\alpha,\beta+\frac{1}{2}}}{V_{\alpha,\beta+\frac{1}{2}}}\cdot(\mathbf{S}^\xi\mathbf{S}_\xi+\mathbf{S}_\xi\mathbf{S}^\xi)_{\alpha+\frac{1}{2},\beta+\frac{1}{2}}\Big)\Big]_{i-\frac{1}{2},j-\frac{1}{2}}\\&-\Big[\frac{u^{\dagger m}_{\alpha,\beta}\mathbf{S}^m_{\alpha,\beta}}{V_{\alpha,\beta}}\cdot\Big(\frac{\mathbf{S}^\eta_{\alpha,\beta+\frac{1}{2}}}{V_{\alpha,\beta+\frac{1}{2}}}\cdot(\mathbf{S}^\xi\mathbf{S}_\xi+\mathbf{S}_\xi\mathbf{S}^\xi)_{\alpha-\frac{1}{2},\beta+\frac{1}{2}}\Big)\Big]_{i+\frac{1}{2},j-\frac{1}{2}}\\&+\Big[\frac{u^{\dagger m}_{\beta}\mathbf{S}^m_{\beta}}{V_{\beta}}\cdot\Big(\frac{\mathbf{S}^\eta_{\beta+\frac{1}{2}}}{V_{\beta+\frac{1}{2}}}\cdot(\mathbf{S}^\eta\mathbf{S}_\xi+\mathbf{S}_\xi\mathbf{S}^\eta)_{\beta+1}\Big)\Big]_{j-1}\\&-\Big[\frac{u^{\dagger m}_{\beta}\mathbf{S}^m_{\beta}}{V_{\beta}}\cdot\Big(\frac{\mathbf{S}^\eta_{\beta+\frac{1}{2}}}{V_{\beta+\frac{1}{2}}}\cdot(\mathbf{S}^\eta\mathbf{S}_\xi+\mathbf{S}_\xi\mathbf{S}^\eta)_{\beta}\Big)\Big]_{j}\\&+\Big[\frac{u^{\dagger m}_{\beta,\gamma}\mathbf{S}^m_{\beta,\gamma}}{V_{\beta,\gamma}}\cdot\Big(\frac{\mathbf{S}^\eta_{\beta+\frac{1}{2},\gamma}}{V_{\beta+\frac{1}{2},\gamma}}\cdot(\mathbf{S}^z\mathbf{S}_\xi+\mathbf{S}_\xi\mathbf{S}^z)_{\beta+\frac{1}{2},\gamma+\frac{1}{2}}\Big)\Big]_{j-\frac{1}{2},k-\frac{1}{2}}\\&-\Big[\frac{u^{\dagger m}_{\beta,\gamma}\mathbf{S}^m_{\beta,\gamma}}{V_{\beta,\gamma}}\cdot\Big(\frac{\mathbf{S}^\eta_{\beta+\frac{1}{2},\gamma}}{V_{\beta+\frac{1}{2},\gamma}}\cdot(\mathbf{S}^z\mathbf{S}_\xi+\mathbf{S}_\xi\mathbf{S}^z)_{\beta+\frac{1}{2},\gamma-\frac{1}{2}}\Big)\Big]_{j-\frac{1}{2},k+\frac{1}{2}}\Bigg\}.\end{aligned}\tag{40}$$

And the term $D_n^\dagger$ can be derived by replacing j by $j+1$ and β by $\beta-1$ in (40). The $D_f^\dagger$ term is derived by interchanging j with k and then β with γ in $D_n^\dagger$, similar to the relation between equation (33) and (34). Finally, $D_b^\dagger$ is obtained by the same exchange of indices in $D_s^\dagger$.

Next, we provide expressions for $R_\xi^\dagger$. The stencil for computing these terms is shown in Fig. 3(b). The indices (i,j,k) in (41)-(43) denote the location of a pressure node (triangular in Fig. 3(b)), which is also the cell center for the solution of the adjoint pressure equations (27) and (30).

$$\begin{aligned}R^\dagger_{\xi,i,j,k}(u^{\dagger\xi})=&-\Big[(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V})_{i-\frac{1}{2},j,k}-(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V})_{i+\frac{1}{2},j,k}\Big]\cdot\mathbf{S}^\xi_{i,j,k}\\&-\frac{1}{4}\sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}}\sum_{\beta=j-\frac{1}{2}}^{j+\frac{1}{2}}\Big[(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V})_{\alpha,\beta-\frac{1}{2},k}-(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V})_{\alpha,\beta+\frac{1}{2},k}\Big]\cdot\mathbf{S}^\eta_{\alpha,\beta,k}\\&-\frac{1}{4}\sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}}\sum_{\gamma=k-\frac{1}{2}}^{k+\frac{1}{2}}\Big[(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V})_{\alpha,j,\gamma-\frac{1}{2}}-(\frac{u^{\dagger\xi}\mathbf{S}^\xi}{V})_{\alpha,j,\gamma+\frac{1}{2}}\Big]\cdot\mathbf{S}^z_{\alpha,j,\gamma}.\end{aligned}\tag{41}$$

The term $R_\eta^\dagger(u^{\dagger\eta})$ can be obtained by switching ξ with η, i with j, and α with β in (41). Similarly, $R_z^\dagger(u^{\dagger z})$ can be derived by interchanging ξ with z, i with k, and α with γ in (41). The adjoint Laplacian on the left-hand side of equation (27) involves six adjoint gradient operators of pressure, which are slightly different from the adjoint gradient of fluxes (41). For example,

$$
\begin{aligned}
R^{\dagger}_{\xi,w}(\phi^{\dagger}) = &-\left[\left(\frac{\phi^{\dagger}_{i-1}\mathbf{S}^{\xi}_{i-\frac{1}{2}}}{V_{i-\frac{1}{2}}}\right)_{j,k} - \left(\frac{\phi^{\dagger}_{i}\mathbf{S}^{\xi}_{i+\frac{1}{2}}}{V_{i+\frac{1}{2}}}\right)_{j,k}\right]\cdot\mathbf{S}^{\xi}_{i,j,k} \\
&-\frac{1}{4}\sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}}\sum_{\beta=j-\frac{1}{2}}^{j+\frac{1}{2}}\left[\left(\frac{\phi^{\dagger}_{\alpha-\frac{1}{2}}\mathbf{S}^{\xi}_{\alpha}}{V_{\alpha}}\right)_{\beta-\frac{1}{2},k} - \left(\frac{\phi^{\dagger}_{\alpha-\frac{1}{2}}\mathbf{S}^{\xi}_{\alpha}}{V_{\alpha}}\right)_{\beta+\frac{1}{2},k}\right]\cdot\mathbf{S}^{\eta}_{\alpha,\beta,k} \\
&-\frac{1}{4}\sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}}\sum_{\gamma=k-\frac{1}{2}}^{k+\frac{1}{2}}\left[\left(\frac{\phi^{\dagger}_{\alpha-\frac{1}{2}}\mathbf{S}^{\xi}_{\alpha}}{V_{\alpha}}\right)_{j,\gamma-\frac{1}{2}} - \left(\frac{\phi^{\dagger}_{\alpha-\frac{1}{2}}\mathbf{S}^{\xi}_{\alpha}}{V_{\alpha}}\right)_{j,\gamma+\frac{1}{2}}\right]\cdot\mathbf{S}^{z}_{\alpha,j,\gamma},
\end{aligned}
\tag{42}
$$

which is also the adjoint of the forward pressure gradient $R_{\xi,e}(\phi)$, and

$$
\begin{aligned}
R^{\dagger}_{\xi,e}(\phi^{\dagger}) = &-\left[\left(\frac{\phi^{\dagger}_{i}\mathbf{S}^{\xi}_{i-\frac{1}{2}}}{V_{i-\frac{1}{2}}}\right)_{j,k} - \left(\frac{\phi^{\dagger}_{i+1}\mathbf{S}^{\xi}_{i+\frac{1}{2}}}{V_{i+\frac{1}{2}}}\right)_{j,k}\right]\cdot\mathbf{S}^{\xi}_{i,j,k} \\
&-\frac{1}{4}\sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}}\sum_{\beta=j-\frac{1}{2}}^{j+\frac{1}{2}}\left[\left(\frac{\phi^{\dagger}_{\alpha+\frac{1}{2}}\mathbf{S}^{\xi}_{\alpha}}{V_{\alpha}}\right)_{\beta-\frac{1}{2},k} - \left(\frac{\phi^{\dagger}_{\alpha+\frac{1}{2}}\mathbf{S}^{\xi}_{\alpha}}{V_{\alpha}}\right)_{\beta+\frac{1}{2},k}\right]\cdot\mathbf{S}^{\eta}_{\alpha,\beta,k} \\
&-\frac{1}{4}\sum_{\alpha=i-\frac{1}{2}}^{i+\frac{1}{2}}\sum_{\gamma=k-\frac{1}{2}}^{k+\frac{1}{2}}\left[\left(\frac{\phi^{\dagger}_{\alpha+\frac{1}{2}}\mathbf{S}^{\xi}_{\alpha}}{V_{\alpha}}\right)_{j,\gamma-\frac{1}{2}} - \left(\frac{\phi^{\dagger}_{\alpha+\frac{1}{2}}\mathbf{S}^{\xi}_{\alpha}}{V_{\alpha}}\right)_{j,\gamma+\frac{1}{2}}\right]\cdot\mathbf{S}^{z}_{\alpha,j,\gamma},
\end{aligned}
\tag{43}
$$

which is similarly the adjoint of $R_{\xi,w}(\phi)$. Other operators, $R^{\dagger}_{\eta,n}(\phi^{\dagger})$, $R^{\dagger}_{\eta,s}(\phi^{\dagger})$, $R^{\dagger}_{z,n}(\phi^{\dagger})$ and $R^{\dagger}_{z,n}(\phi^{\dagger})$, are available by applying the exchange of indices that was described after (41) to equations (42) and (43). Note that in Cartesian coordinates, the Laplacian operator is self-adjoint, which in general does not hold in curvilinear coordinates.

From expressions (31)–(43), the discrete adjoint equations use the same stencil as the forward ones (Fig. 2). Furthermore, the coefficients of the adjoint and forward terms have similar forms, which ensures that the floating-point operations are of the same orders of magnitude. The skeleton of the forward solver, including the parallelization strategy, could therefore be easily adapted and adopted for the adjoint. In terms of stability, the Lyapunov exponents are exactly the same for the forward and adjoint systems. In other words, a time step Δt for which the forward solver is stable is also appropriate for the herein derived discrete adjoint model.

2.4. Boundary and initial conditions

At boundaries, some of the cells in the computational stencil are outside the physical domain, or are 'ghost' cells (see Fig. 4). Fluxes at these cells are determined from the boundary conditions. If periodicity is applied in the forward model, the adjoint boundary condition should also be periodic in the same direction, as shown in Appendix A. Here we derive the discrete adjoint of the no-slip boundary condition,

$$
\boldsymbol{u} = \boldsymbol{u}_w \tag{44}
$$

which is commonly adopted in simulations of wall-bounded flows. Note that $\boldsymbol{u}_w$ can have a wall-normal component, which represents blowing or suction. Without loss of generality, the wall-normal direction is assumed to be the η coordinate. Then the discrete boundary conditions can be expressed as,

$$
\begin{aligned}
&|\mathbf{S}_{\xi,0}|u^{\xi}_{0} = 2|\mathbf{S}_{\xi,1/2}|\boldsymbol{u}_w\cdot\mathbf{S}^{\xi}_{1/2} - |\mathbf{S}_{\xi,1}|u^{\xi}_{1}, \quad u^{\eta}_{1} = \boldsymbol{u}_w\cdot\mathbf{S}^{\eta}_{1}, \\
&|\mathbf{S}_{z,0}|u^{z}_{0} = 2|\mathbf{S}_{z,1/2}|\boldsymbol{u}_w\cdot\mathbf{S}^{z}_{1/2} - |\mathbf{S}_{z,1}|u^{z}_{1}, \quad u^{\eta}_{0} = u^{\eta}_{2},
\end{aligned}
\tag{45}
$$

and used for computing the right-hand side of the discrete momentum equation (14). The last condition in (45) is due to continuity. The subscripts correspond to the indices in the wall-normal direction (see Fig. 4).

Since the discrete forward equations are only valid at physical control volumes, the adjoint variables in the Lagrangian function (22) are not defined at ghost cells. However, to evaluate the adjoint advection and diffusion terms in (29), values of $\hat{u}^{\dagger q}$ at ghost cells are required and are derived from the forward boundary conditions. Consider the diffusion term D_{ξ} in (19) as an example. The terms in D_{ξ} that correspond to derivatives in the wall-normal direction are denoted as $A_{\xi}(u^{\xi})$,

$$
A_{\xi,j}(u^{\xi}) = a_j u^{\xi}_{j+1} + b_j u^{\xi}_{j} + c_j u^{\xi}_{j-1}, \tag{46}
$$

and the adjoint counterpart $A^{\dagger}_{\xi}(\hat{u}^{\dagger\xi})$ is

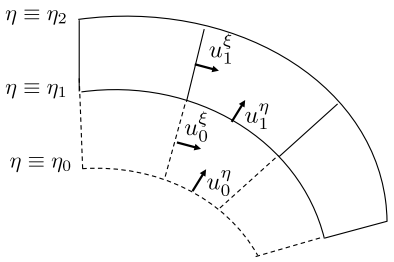

Fig. 4. Interior cells (——) and ghost cells (– – –) near a solid boundary ($\eta \equiv \eta_1$) of wall-bounded flows.

$$A^\dagger_{\xi,j}(\hat{u}^{\dagger\xi}) = a^\dagger_j \hat{u}^{\dagger\xi}_{j+1} + b^\dagger_j \hat{u}^{\dagger\xi}_j + c^\dagger_j \hat{u}^{\dagger\xi}_{j-1}. \tag{47}$$

Based on the forward-adjoint relation, the coefficients in (47) are $a^\dagger_j = c_{j+1}$, $b^\dagger_j = b_j$ and $c^\dagger_j = a_{j-1}$, when none of the terms in (46), (47) are evaluated at ghost cells. However, at $j = 1$ the derivative of $\hat{u}^{\dagger\xi}_1 A_{\xi,1}(u^\xi)$ with respect to u^ξ_1 is,

$$\frac{\partial(\hat{u}^{\dagger\xi}_1 A_{\xi,1}(u^\xi))}{\partial u^\xi_1} = \hat{u}^{\dagger\xi}_1 \left(b_1 - c_1 \frac{|\mathbf{S}_{\xi,1}|}{|\mathbf{S}_{\xi,0}|} \right) \tag{48}$$

due to the forward boundary condition (45). In order to render the expression of $A^\dagger_{\xi,j}(\hat{u}^{\dagger\xi})$ (47) valid at $j = 1$, we can define the $\hat{u}^{\dagger\xi}_0$ at the ghost cell as,

$$\hat{u}^{\dagger\xi}_0 = -\frac{c_1}{c^\dagger_1} \frac{|\mathbf{S}_{\xi,1}|}{|\mathbf{S}_{\xi,0}|} \hat{u}^{\dagger\xi}_1. \tag{49}$$

As a result, the coefficient of $\hat{u}^{\dagger\xi}_1$ in (47) is consistent with (48). The boundary conditions for η and z components can be derived in the same manner. Since the coefficients of fluxes at ghost cells are different in forward advection, implicit diffusion and explicit diffusion terms, the no-slip adjoint boundary conditions for evaluating their adjoints in (29) are distinct.

The starting condition for the discrete adjoint algorithm at $t = t_e$, or $t^\dagger = t_s$, is directly available from setting the derivative of the Lagrangian (22) with respect to u^{qN} and p^N to zero,

$$(u^{\dagger q})^N = \frac{\partial J}{\partial (u^q)^N}, \quad (p^\dagger)^N = \frac{\partial J}{\partial p^N}. \tag{50}$$

In general, J is a quadratic function in the instantaneous field $(u^q)^n$ and p^n. Therefore, the derivatives in (50), and similarly the source terms $\partial J/\partial (u^\xi)^n$, $\partial J/\partial p^n$ in adjoint momentum equations (29)-(30), can be obtained analytically.

Ultimately we seek the gradient of the cost function with respect to initial velocity field, $\boldsymbol{u}$. The gradient is computed by the chain rule,

$$\frac{\partial J}{\partial \boldsymbol{u}^0} = \frac{\partial J}{\partial (u^q)^0} \frac{\partial (u^q)^0}{\partial \boldsymbol{u}^0} = (u^{\dagger q})^0 \frac{\partial (u^q)^0}{\partial \boldsymbol{u}^0} \tag{51}$$

Note that the last equality is because the derivative of cost function with respect to initial flux is equal to the adjoint flux at the initial time. The derivative $\partial u^q/\partial \boldsymbol{u}^0$ can be obtained analytically from the relation between velocity and flux (12).

With the discrete adjoint operators and the associated boundary and initial conditions, at every time step the adjoint Poisson equation (27) is solved first to evaluate the adjoint pseudo pressure $\phi^\dagger$. Its solution adopts the same numerical algorithm as the forward one, but with different coefficients of the adjoint pseudo pressure. Then the "adjoint projection" step is performed (28); note that the projected field is not $u^{\dagger q}$. Finally the adjoint flux and adjoint pressure at the next step, in reverse time, are updated using the adjoint momentum equation (29) and the relation between $p^{\dagger(n+1)}$ and $p^{\dagger n}$ (30), which requires the forward flux field at that time instant. The updated adjoint flux $u^{\dagger q(n+1)}$, as well as the gradient of the cost function (51), are thus not divergence free. In §2.5, we provide a symmetric operator to project the final outcome of the adjoint marching algorithm onto a divergence-free field.

2.5. Symmetric projection onto divergence-free field

As remarked in the previous section, the final output of the adjoint equations (27-30) is not divergence free. As a result, the gradient of the cost function (51) and the updated initial condition based on this gradient are not guaranteed to be solenoidal. Yet it is important to enforce a divergence-free constraint on the initial condition of the forward flow field, i.e. the velocity $\boldsymbol{u}^0$,

$$\frac{u^{0\xi}_{i+\frac{1}{2}} - u^{0\xi}_{i-\frac{1}{2}} + u^{0\eta}_{j+\frac{1}{2}} - u^{0\eta}_{j-\frac{1}{2}} + u^{0z}_{k+\frac{1}{2}} - u^{0z}_{k-\frac{1}{2}}}{V_{i,j,k}} \equiv \mathbf{D}_{iv,f}\mathbf{S}\boldsymbol{u}^0 \equiv \mathbf{D}_{iv}\boldsymbol{u}^0 = 0, \tag{52}$$

where $\mathbf{D}_{iv,f}$ is the divergence operator for the flux, $\mathbf{S}$ is the discrete operator transforming velocity to flux, and $\mathbf{D}_{iv}$ is the divergence operator for the velocity. We therefore introduce a projection operator $\mathbf{P}$ which satisfies the following conditions:

1. $\forall\ \boldsymbol{u}^0$, $\mathbf{D}_{iv}\mathbf{P}\boldsymbol{u}^0 = 0$;
2. If $\mathbf{D}_{iv}\boldsymbol{u}^0 = 0$, $\mathbf{P}\boldsymbol{u}^0_f = \boldsymbol{u}^0$;
3. The projection operator must be symmetric, $\mathbf{P}^T = \mathbf{P}$.

The last criterion is essentially a requirement that the projection is self adjoint. With this condition, it can be introduced at the start of forward marching and at the end of the adjoint evolution, while maintaining the accuracy of the discrete forward-adjoint duality. This property will be beneficial in the gradient-based optimization procedure (see §2.6 below). Under the above three conditions, the projector is unique,

$$\mathbf{P} = \mathbf{I} - \mathbf{D}^T_{iv}(\mathbf{D}_{iv}\mathbf{D}^T_{iv})^{-1}\mathbf{D}_{iv}. \tag{53}$$

Note that this projection operator is different from the one adopted in the fractional-step algorithm (15-16) because in those equations the discrete Laplacian ∇^2 is not $(\mathbf{D}_{iv}\mathbf{D}^T_{iv})$ and the gradient is not $\mathbf{D}^T_{iv}$.

With the projector $\mathbf{P}$, any velocity field $\boldsymbol{u}^{0^-}$ can be projected onto a solenoidal field by computing $\boldsymbol{u}^0 = \mathbf{P}\boldsymbol{u}^{0^-}$. In orthogonal coordinates, $\mathbf{S}$ is a diagonal matrix, which renders the projector (53) as efficient to implement as the projection step in the forward equations (15-16). Therefore, the projector is equivalent to solving the following equations, where (i, j, k) refers to the center of the primary pressure cell,

$$\begin{aligned}
&\left(\frac{\phi_i}{V_i} - \frac{\phi_{i+1}}{V_{i+1}}\right)\left|\mathbf{S}^{\xi}_{i+\frac{1}{2}}\right|^2 - \left(\frac{\phi_{i-1}}{V_{i-1}} - \frac{\phi_i}{V_i}\right)\left|\mathbf{S}^{\xi}_{i-\frac{1}{2}}\right|^2 + \left(\frac{\phi_j}{V_j} - \frac{\phi_{j+1}}{V_{j+1}}\right)\left|\mathbf{S}^{\eta}_{j+\frac{1}{2}}\right|^2 \\
&- \left(\frac{\phi_{j-1}}{V_{j-1}} - \frac{\phi_j}{V_j}\right)\left|\mathbf{S}^{\eta}_{j-\frac{1}{2}}\right|^2 + \left(\frac{\phi_k}{V_k} - \frac{\phi_{k+1}}{V_{k+1}}\right)\left|\mathbf{S}^{z}_{k+\frac{1}{2}}\right|^2 - \left(\frac{\phi_{k-1}}{V_{k-1}} - \frac{\phi_k}{V_k}\right)\left|\mathbf{S}^{z}_{k-\frac{1}{2}}\right|^2 \\
&= \left(\boldsymbol{u}^{0^-}\cdot\mathbf{S}^{\xi}\right)_{i+\frac{1}{2}} - \left(\boldsymbol{u}^{0^-}\cdot\mathbf{S}^{\xi}\right)_{i-\frac{1}{2}} + \left(\boldsymbol{u}^{0^-}\cdot\mathbf{S}^{\eta}\right)_{j+\frac{1}{2}} - \left(\boldsymbol{u}^{0^-}\cdot\mathbf{S}^{\eta}\right)_{j-\frac{1}{2}} \\
&\quad + \left(\boldsymbol{u}^{0^-}\cdot\mathbf{S}^{z}\right)_{k+\frac{1}{2}} - \left(\boldsymbol{u}^{0^-}\cdot\mathbf{S}^{z}\right)_{k-\frac{1}{2}},
\end{aligned} \tag{54}$$

$$u^0_{i+\frac{1}{2}} = u^{0^-}_{i+\frac{1}{2}} - \left(\frac{\phi_{i,j,k}}{V_{i,j,k}} - \frac{\phi_{i+1,j,k}}{V_{i+1,j,k}}\right)\left|\mathbf{S}^{\xi}_{i+\frac{1}{2}}\right|. \tag{55}$$

The equations for updating the other components are similar to (55).

In generalized coordinates, the term $\mathbf{D}_{iv}\mathbf{D}^T_{iv}$ in the projector (53) comprises a $5\times5\times5$ stencil. In lieu of solving the associated elliptic equation and projecting the velocity onto a solenoidal field, $\boldsymbol{u}^0 = \mathbf{P}\boldsymbol{u}^{0^-}$, we can solve the equation,

$$\mathbf{D}^T_{iv}\mathbf{D}_{iv}\boldsymbol{u}^0 = 0 \tag{56}$$

using conjugate gradient method with $\boldsymbol{u}^{0^-}$ as the initial guess. The connection to the projector (53) can be explained by considering that the vector space $\boldsymbol{U}^0$ comprised of all possible initial velocity fields can be written as the direct sum of the kernel and the row space of $\mathbf{D}_{iv}$,

$$\boldsymbol{U}^0 = ker(\mathbf{D}_{iv}) \oplus rowsp(\mathbf{D}_{iv}). \tag{57}$$

For any velocity field $\boldsymbol{u}^{0^-} \in \boldsymbol{U}^0$, the decomposition (57) is unique. The projector $\mathbf{P}$ in (53) removes the components within $rowsp(\mathbf{D}_{iv})$ and retains those within $ker(\mathbf{D}_{iv})$ unchanged. As for the solution of the conjugate gradient procedure, it can be written as,

$$\boldsymbol{u}^0 = \boldsymbol{u}^{0^-} - \text{span}\{(\mathbf{D}^T_{iv}\mathbf{D}_{iv})\boldsymbol{u}^{0^-}, (\mathbf{D}^T_{iv}\mathbf{D}_{iv})^2\boldsymbol{u}^{0^-}, (\mathbf{D}^T_{iv}\mathbf{D}_{iv})^3\boldsymbol{u}^{0^-}, \dots\}. \tag{58}$$

All the search directions are within the column space of $\mathbf{D}^T_{iv}$, or the row space of $\mathbf{D}_{iv}$. Since the search directions are conjugate with respect to $\mathbf{D}^T_{iv}\mathbf{D}_{iv}$, they form an orthogonal basis for $rowsp(\mathbf{D}_{iv})$ as long as sufficient iterations are performed. Therefore, the solution (58) contains only the components of $\boldsymbol{u}^{0^-}$ in $ker(\mathbf{D}_{iv})$, and is equivalent to $\mathbf{P}\boldsymbol{u}^{0^-}$.

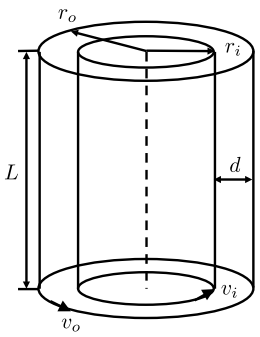

Fig. 5. Schematic of circular Couette flow configuration.

2.6. Gradient-based optimization

With the gradient of the cost function (51) and the projection operator (53), an efficient gradient-based optimization algorithm can be adopted to minimize the cost function. Here the limited-memory formulation of quasi-Newton BFGS method (L-BFGS) is adopted [31] due to its applicability and wide use in large-scale non-linear optimization problems. The L-BFGS method was originally proposed for unconstrained optimization and, therefore, we modify it slightly to enforce the divergence-free constrain. The basic idea is to update the initial condition using,

$$\boldsymbol{u}_{k+1}^0 = \mathbf{P}\left(\boldsymbol{u}_k^0 + \alpha_k \boldsymbol{d}_k\right), \tag{59}$$

where subscript k identifies the k-th iterate of the optimization procedure, and $\boldsymbol{d}_k = -\mathbf{B}_k\mathbf{P}(\nabla J)$. The term $\mathbf{B}_k$ is a rank-two approximation of the inverse of the Hessian matrix of the cost function. The step size α_k is updated through the line search routine CVSRCH [32], which enforces the strong Wolfe condition and uses cubic interpolation to compute α_k. Therefore, it might require more than one forward-adjoint loop to obtain α_k at each L-BFGS iteration. Another possible choice of α_k is to calculate the optimal step size in the linear sense [33,34]. This choice was also tested. The convergence history of the cost function is comparable with line search, but additional computational cost is required to compute the optimal step size.

The symmetric projector features twice in (59), after the adjoint simulation and before the forward simulation. In the first instance, $\mathbf{P}(\nabla J)$ ensures that we are searching in solenoidal space. The second projection, $\mathbf{P}\boldsymbol{d}_k$, is also necessary because multiplication of the search direction by $\mathbf{B}_k$ can become non-solenoidal. The symmetric property, $\mathbf{P}^T = \mathbf{P}$, guarantees that the action of $\mathbf{P}$ at the end of the adjoint evolution is the adjoint of its action at the start of the forward evolution.

Combining the forward and adjoint solvers with the optimization algorithm gives the following minimization procedure for the cost function:

1. Start with a first guess of the initial condition $\boldsymbol{u}^{0^-}$, and compute its projection onto a divergence-free space $\boldsymbol{u}^0 = \mathbf{P}\boldsymbol{u}^{0^-}$.
2. March the forward equations with $\boldsymbol{u}^0$ from $n = 0$ to $n = N$ and store the forward flux fields at every time step.
3. Solve discrete adjoint equations from $n = N$ to $n = 0$ to obtain the adjoint flux at the initial time, which is equal to the gradient of the cost function.
4. Update the next guess of the initial condition using (59), and repeat steps 2-4 until convergence or until a set maximum number of iterations is reached.

3. A case study: data assimilation of circular Couette flow

In order to assess the performance of our algorithm, we applied it to reconstruction of circular Couette flow from limited observations. The circular Couette configuration (Fig. 5) is characterized by the ratio of the inner and outer radii $\chi = r_i/r_o$ and the aspect ratio $\Gamma = L/d$, where d is the gap width $d = r_o - r_i$. We consider a stationary outer cylinder, and therefore the Reynolds number is defined as $Re = v_i d/\nu$, where v_i is the tangential speed of the inner cylinder. The laminar state is only comprised of an azimuthal velocity,

$$\boldsymbol{U}_{lam} = (Ar + B/r)\,\mathbf{e}_\theta, \quad A = -\frac{v_i}{r_i}\frac{\chi^2}{1-\chi^2}, \quad B = r_i v_i \frac{1}{1-\chi^2}. \tag{60}$$

Computational parameters for the two test cases examined herein are summarized in Table 1.

3.1. Validation of forward and adjoint codes

In order to validate the implementation of the forward and adjoint models, we compute the evolution of small-amplitude linear instability waves and their adjoints using the full nonlinear Naviers-Stokes and adjoint operators and compare the

Table 1
Computation setup of two test cases.

Test cases	Re	χ	Γ	Grid	Δy_{min}	Δy_{max}	Δt
Case 1	400	0.714	2π	$128 \times 64 \times 128$	0.007	0.024	0.02
Case 2	3200	0.617	4.58	$256 \times 128 \times 256$	0.004	0.012	0.005

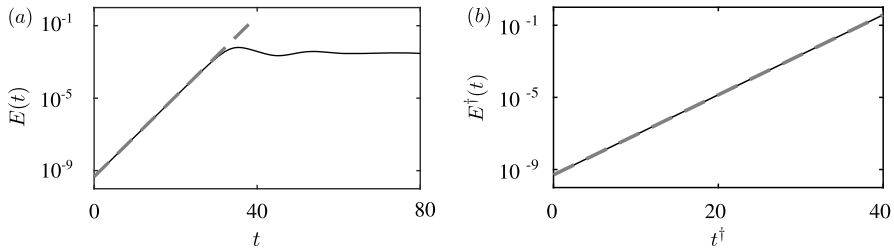

Fig. 6. Time evolution of (a) forward perturbation energy and (b) adjoint energy. (– – –) Prediction of linear theory; (——) numerical simulation.

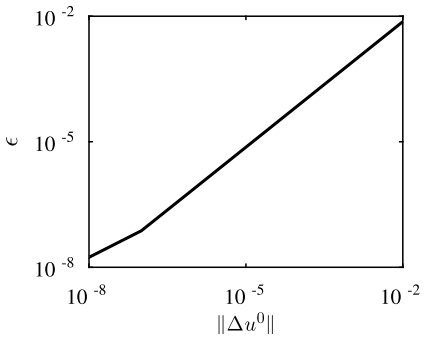

Fig. 7. Validation of the forward-adjoint relation: relative error ϵ between the gradients obtained from the adjoint algorithm and from finite difference approximation plotted versus the perturbation $\|\Delta u^0\|$.

instability growth rates to linear stability theory. The initial condition of the simulations is a superposition of the laminar base state and a small-amplitude perturbation,

$$\boldsymbol{u}^0 = \boldsymbol{U}_{lam} + \boldsymbol{u}'. \tag{61}$$

$\boldsymbol{U}_{lam}$ is given in (60). The perturbation has the form, $v'_j = a\mathrm{Re}\{\hat{v}_j e^{i(k_m\theta + k_z z)}\}$, where $\hat{v}_j$ is the eigenmode of the linear stability equations (given in Appendix B), a is the modal amplitude, and k_m and k_z are the azimuthal and axial wavenumbers. The simulation setup corresponds to the first test case in Table 1, and we select the most unstable mode at wavenumber $(k_z, k_m) = (3, 1)$ for validation. The perturbation energy density,

$$E(t) = \frac{1}{V}\int_V \frac{1}{2}\left(v_\theta'^2 + v_r'^2 + v_z'^2\right) r dr d\theta dz \tag{62}$$

is shown in Fig. 6(a) as a function of time. In the early stage of the modal evolution, E amplifies exponentially at the rate that is predicted by linear theory to within 0.2%, which serves to validate the forward model [35]—additional validations were performed for unstable and stable modes with different values of k_m and k_z with similar agreement. Beyond the linear stage in Fig. 6(a), the perturbation amplitude becomes appreciable, and nonlinear effects lead to energy exchanges with higher harmonics and the mean flow, and a saturated state is ultimately established.

The implementation of the discrete adjoint can be validated in a similar manner, by comparing the exponential growth rate of the adjoint mode from the simulations with the prediction by linear theory (Appendix B). Note that the continuous adjoint eigenmode obtained from the adjoint stability equations (B.14)-(B.17) should be transformed into discrete adjoint fluxes according to the relation between continuous and discrete adjoints (24)–(26). Since the adjoint operator is linear, the adjoint energy density,

$$E^{\dagger}(t) = \frac{1}{V}\int_V \frac{1}{2}\left(v_\theta^{\dagger 2} + v_r^{\dagger 2} + v_z^{\dagger 2}\right) r dr d\theta dz \tag{63}$$

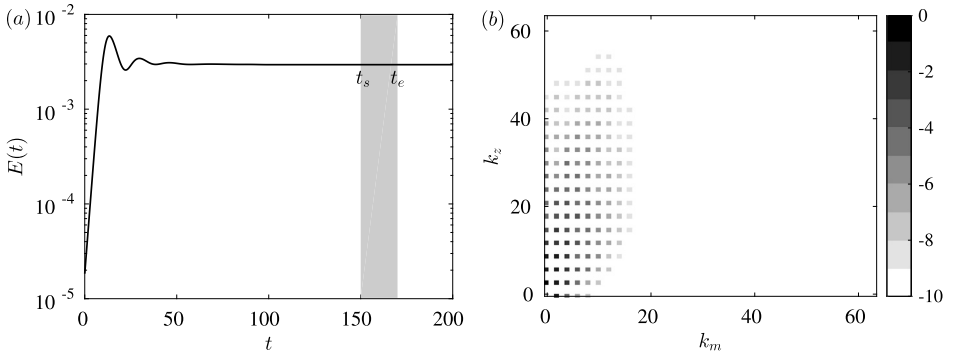

Fig. 8. (a) Time evolution of the total energy in the data-acquisition simulation. The gray region corresponds to the data-acquisition interval. (b) Energy spectra $E(k_z, k_m)$ of the true saturated flow, normalized by its infinity norm.

grows exponentially at all times and does not saturate (Fig. 6(b)). The evolution of $E^\dagger$ agrees well with linear theory, and the error in the growth rate is less than 0.2%.

The forward-adjoint duality relation was validated by comparing the gradient of the cost function obtained from the adjoint algorithm, ∇J_A, and from a finite difference approximation when the forward field is perturbed slightly, ∇J_F. The relative error between these two gradients is defined as ϵ:

$$\nabla J_A = \frac{\partial J}{\partial \boldsymbol{u}^0}; \qquad \nabla J_F = \frac{J(\boldsymbol{u}^0 + \Delta \boldsymbol{u}^0) - J(\boldsymbol{u}^0)}{\Delta \boldsymbol{u}^0}; \qquad \epsilon = \frac{\|\nabla J_A - \nabla J_F\|}{\|\nabla J_A\|} \tag{64}$$

The norm in (64) is L-2 over all the grid points and the three components of the velocity field. As shown in Fig. 7, the relative difference decreases with $\|\Delta u^0\| \to 0$ down to square root of machine precision. This result is independent of the number of grid points and the parameters of the flow configuration. If $\Delta \boldsymbol{u}^0$ is decreased further, the finite difference approximation becomes contaminated by round-off errors and not sufficiently accurate for validating the forward-adjoint relation—see Appendix C. Note that the accuracy of the forward-adjoint relation is unaffected by truncation errors when the discrete adjoint formulation is adopted. In contrast, in the continuous-adjoint approach, the error in the forward-adjoint relation is dominated by the truncation terms of the discretization scheme.

3.2. Test case 1: wavy vortex flow

The first test case for data assimilation corresponds to a saturated wavy vortex flow, that was previously examined using direct numerical simulations by Ostilla et al.[36]. In order to reach the saturated state, the initial condition is a superposition of the laminar base flow and four unstable modes with different wavenumber pairs, $(k_z, k_m) = (3, 0), (3, 1), (4, 0), (4, 1)$. The amplitude of each of the modes is 1% of v_i. After sufficiently long time, the saturated state is reached and the integrated kinetic energy becomes constant (Fig. 8(a)), and the energy spectra of the saturated state are shown in Fig. 8(b). As a result of nonlinear modal interaction, the harmonic $(k_z, k_m) = (3, 0)$ becomes the most dominant, which contains about 81.4% of the perturbation energy. The harmonic $(k_z, k_m) = (3, 2)$ is the second most dominant, containing about 5.4% of the energy. Despite the finite energy in wavy traveling vortices, and the flow being unsteady, the integrated energy is constant in the saturated state. Note that there are other possible states for the same Re and χ, depending on the specific geometry of the circular Couette cell (e.g. axial extent end effects), initial condition and flow history. For example, starting from a lower Reynolds number and gradually increasing the velocity of the cylinder would lead to axisymmetric Taylor vorticies.

Data were acquired from this reference simulation during the saturated state, over the time interval $t \in [t_s, t_e]$, where $t_e - t_s = 20$ convective time units. The observed quantity is the wall shear stress τ_w on both cylinders,

$$\tau_w = \nu \left(\frac{\partial v_\theta}{\partial r} - \frac{v_{\theta,w}}{r_w} \right). \tag{65}$$

The flood contours in Fig. 9 show the observation on the outer cylinder at $t = t_s$ and t_e, which capture the wavy Taylor vortices.

The objective of the data assimilation is to predict the flow state at t_s. The starting guess of the algorithm was a purely laminar flow without any perturbations, which would yield a constant wall stresses. According to Fig. 10, after 100 L-BFGS iterations (105 forward-adjoint loops), the cost function is decreased by three orders of magnitude relative to its initial value. Note that the value of cost function is always decaying from one iteration to the next, due to the accurate evaluation of the gradient from the discrete adjoint. The estimated shear stress on the outer cylinder due to the predicted initial condition is compared to the observation data in Fig. 9 (lines and color contours, respectively). At the initial time, the laminar solution has been replaced with a flow field that captures the shape of wavy Taylor vortices (Fig. 9(a)), and is in close agreement with the observations at $t = t_e$ (Fig. 9(b)). The improvement in the wall stresses with time can be

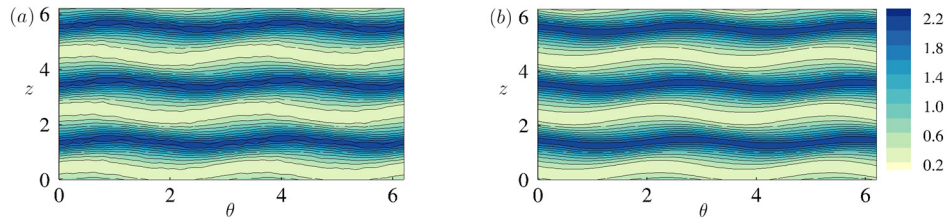

Fig. 9. Wall shear stress τ_w at the outer cylinder (normalized by the averaged value over the outer cylinder). (a) $t = t_s$; (b) $t = t_e$. Color contours are the true wall shear stress and lines contours denote the estimation.

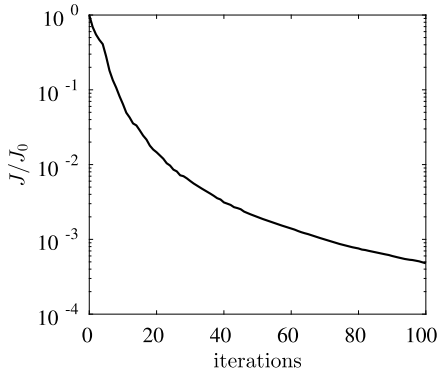

Fig. 10. Value of the cost function (normalized by the value at the first guess) versus number of L-BFGS iterations.

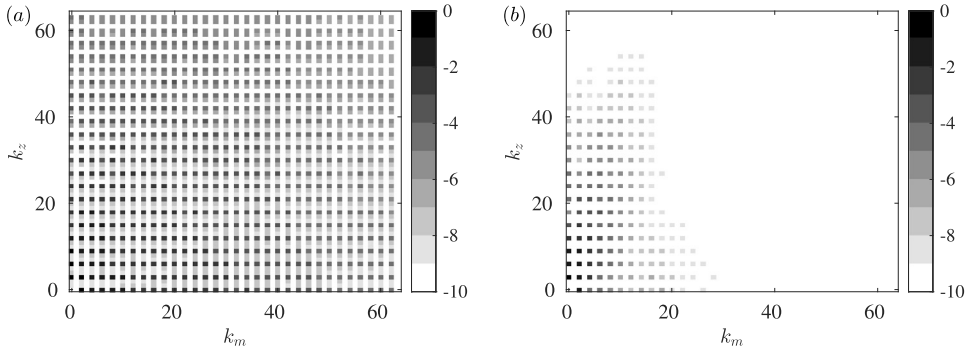

Fig. 11. Energy spectra $E(k_z, k_m)$ of the estimated state at (a) $t = t_s$ and (b) $t = t_e$, normalized by its infinity norm.

explained by considering the energy spectra in Fig. 11. Panel (a) shows that the estimated initial condition has a wider band of wavenumber components. Although the erroneous energy within high-wavenumber modes increases the estimation error of the initial state, the majority of these modes are stable and decay very quickly with time. As a result, they have little impact on the cost function and are not eliminated by the optimization algorithm. The erroneous modes vanish by the end of the assimilation window, the spectra at $t = t_e$ (panel b) converges towards the true state (cf. Fig. 8) and, naturally, the estimation of the wall shear stress also improves.

Comparisons between the true and assimilated mean-velocity profiles and also mode shapes is provided in Fig. 12. At $t = t_s$, the true mean azimuthal velocity is strongly distorted in the bulk region and the near-wall profile is much steeper than the laminar state. The mean of the estimated state is close to the true profile only within a region very close to the wall where the observations are recorded. In the bulk region, however, the estimation is less accurate and lies between the laminar and the true profiles. At $t = t_e$ the mean of the estimated state becomes a more accurate representation of the true state. Fig. 12(b) shows the most energetic non-axisymmetric mode whose wavenumbers are $(k_z, k_m) = (3, 2)$. The estimated mode shape at $t = t_s$ can be regarded as a superposition of different modes with the same wavenumbers, and most of them cannot hold except the one in the true flow field. The mode shape is estimated most accurately at $t = t_e$. Visualizations of the true and estimated instantaneous fields are shown in Fig. 13(a) and (b) respectively. The contours of v_θ and the streamlines are all very similar, which confirms that the data assimilation algorithm has converged onto the true state.

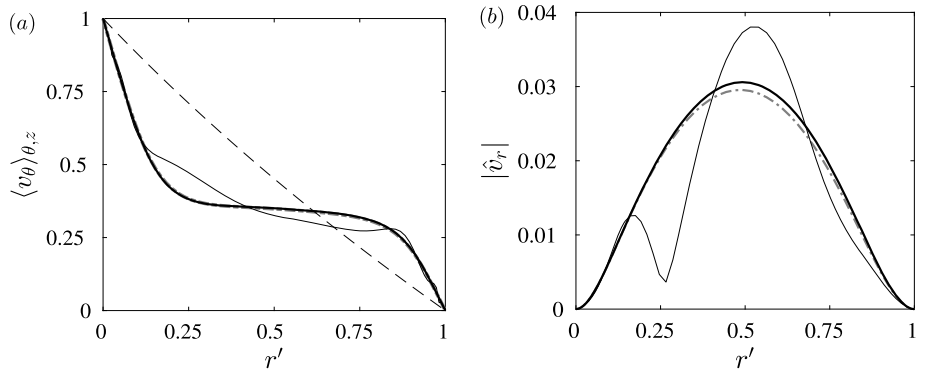

Fig. 12. Wall-normal profiles of the true and estimated states, where $r' = r - r_i$. (a) Mean azimuthal velocity (averaged over azimuthal and axial directions); (b) Radial component of $(k_z, k_m) = (3, 2)$ mode. (– – –) Laminar flow which is also the first guess of flow state; (– · –) true state; (——) estimated state at $t = t_s$; (▬) estimated state at $t = t_e$.

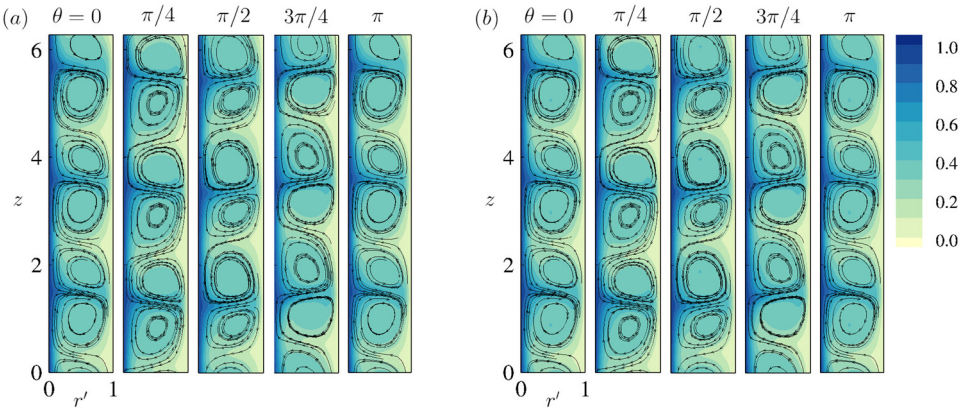

Fig. 13. Visualizations of (a) true and (b) estimated instantaneous velocity fields at $t = t_e$. The flood contour is v_θ, and streamlines are v_r and v_z components.

Although the estimation is accurate by the end of the assimilation window, a more challenging task is to improve the accuracy of the estimated initial condition. One possible approach is to incorporate time-dependent weights in the cost function,

$$J = \frac{1}{2}\sum_{n=0}^{N}\frac{1}{2}\|\boldsymbol{y}^n - \boldsymbol{h}(\boldsymbol{u}^n)\|_0^2 \exp(\lambda(t_n - t_s)), \tag{66}$$

where λ is a parameter that can be chosen to more heavily weight earlier ($\lambda < 0$) or later ($\lambda > 0$) observations. When $t_n = t_s$ in (66) the weight is unity, and when $t_n = t_e$ the weight $\exp(\lambda(t_e - t_s))$ represents the relative importance of the observation at $t = t_e$ compared with $t = t_s$. Five different weights were tested, $\exp(\lambda(t_e - t_s)) = \{0.01, 0.1, 1, 10, 100\}$; note that $\exp(\lambda(t_e - t_s)) = 1$ corresponds to the uniform weighting, or the original definition of the cost function. The errors in the estimated initial velocity fields, after 100 L-BFGS iterations, are reported in Fig. 14. When higher weights are applied to earlier observations (black solid lines), the estimation errors within approximately 20% of the assimilation window are significantly reduced, although the errors at later times are very slightly increased. The improved performance is directly tied to eliminating the erroneous energy in high-wavenumber modes, which affect the early portion of the observations. It can be anticipated that there exists an optimal value of $\lambda < 0$, because an arbitrarily large negative value will not yield an accurate solution. In that extreme limit, only data at $t = t_s$ are important and the estimated initial state is only updated (relative to the initial guess) in the very-near-wall region but not in the bulk, thus leading to high estimation errors at later times.

3.3. Test case 2: turbulent Taylor-Couette flow

In the first test case, although multiple saturated states were possible at the same flow parameters, these states are stable. As such, any initial condition within the basin of attraction of the target state will converge onto it. For turbulent circular Couette flow, multiple statistically stationary states can exist for the same flow parameters, and depend for example

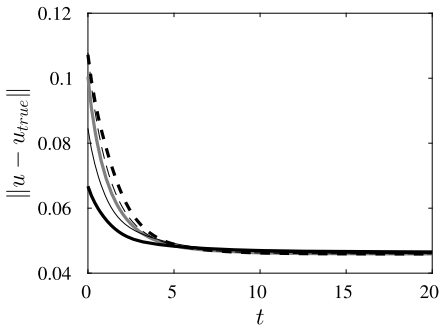

Fig. 14. Root-mean-squared error of instantaneous velocity field estimated using different weights. (▬) Uniform weights for all observations, $\exp(\lambda(t_e - t_s)) = 1$; (——) $\exp(\lambda(t_e - t_s)) = 0.1$; (▬) $\exp(\lambda(t_e - t_s)) = 0.01$; (– – –) $\exp(\lambda(t_e - t_s)) = 10$; (▪ ▪ ▪) $\exp(\lambda(t_e - t_s)) = 100$.

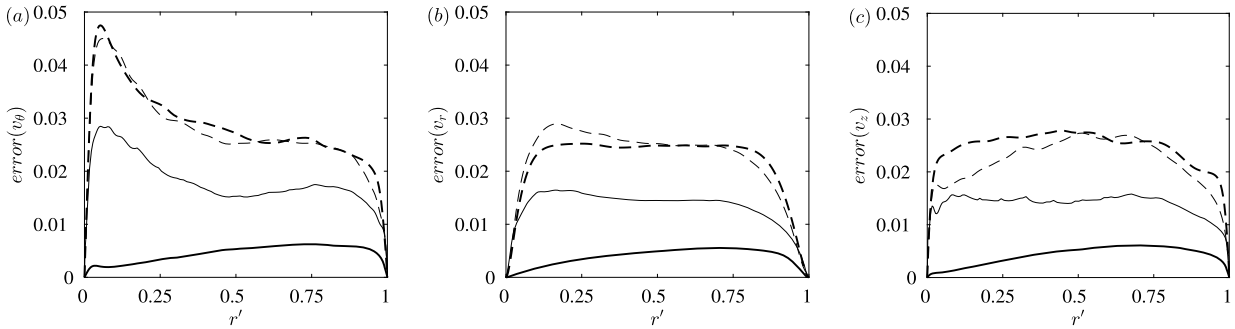

Fig. 15. Wall-normal profiles of root-mean-square error (67) between the true and estimated flow states: (a) Azimuthal, (b) radial and (c) axial components. (– – –) Spline interpolation of observed velocity field at $t = t_s$, also the first guess of the state estimation algorithm; (▪ ▪ ▪) outcome of advancing the interpolated initial condition until $t = t_e$; (——) estimated flow state at $t = t_s$; (▬) estimated flow state at $t = t_e$.

on the flow startup process. Converging onto a particular one is less straightforward in the turbulent regime. And even when there is a unique statistically stationary state, the instantaneous fields are very sensitive to initial conditions due to the chaotic nature of turbulence. In this section we aim to reconstruct these instantaneous turbulent fields from limited measurements.

Previous efforts to reconstruct fully resolved turbulent flow fields using data assimilation are limited and have generally focused on channel flow only, e.g. Bewley et al. [15]. The present test case is the first attempt to reconstruct fully resolved turbulent flow in curvilinear coordinates. Similar to case 1, a data acquisition simulation is first performed where observations are recorded and subsequently used for state estimation using the data-assimilation algorithm.

The flow parameters of this turbulent circular Couette flow configuration are similar to the simulations by Bilson & Bremhorst [4], and are summarized in Table 1. The data acquisition computation was performed starting from an initial state that is a superposition of a constant-angular-momentum profile and white noise with amplitude equal to 30% of local velocity. The simulation was then advanced until a statistically stationary state was established. Observations were stored during a time interval $t_e - t_s = 2.5$, which is approximately one fourth of the period of rotation of the inner cylinder. Since the dynamics are chaotic, the assimilation window should not exceed the time scale associated with the leading Lyapunov exponent of the true state. Otherwise, small errors in the observations or in the adjoint field will grow exponentially with the length of the window and render the optimization algorithm very difficult, even impossible, to converge. When the assimilation window is too long, the cycling scheme can be adopted [37]. Specifically, a long window can be split into several shorter overlapped one, and adjoint-based approach can be applied within each window. In terms of the choice of observations, it was demonstrated by previous studies that wall shear-stress measurements are not sufficient to accurately predict the turbulent state [15,38]. Therefore, in the present test case we use sub-sampled velocities as observations; these are intended to represent limited-resolution experimental measurements that we will augment using data assimilation. The observations are recorded at every eighth grid points, yielding $32 \times 16 \times 32$ observation locations. They are also sub-sampled in time at $\Delta t_{obs} = 10\Delta t_{CFD}$.

The data assimilation algorithm was applied to reconstruct the full-resolution initial state using the low-resolution observations. The first guess of the initial condition was a three-dimensional spline interpolation of the observed velocity field at $t = t_s$. After 100 L-BFGS iterations (103 forward-adjoint loops), the cost function was decreased by three orders of magnitude relative to its initial value. The root-mean-square error between the true and estimate flow states,

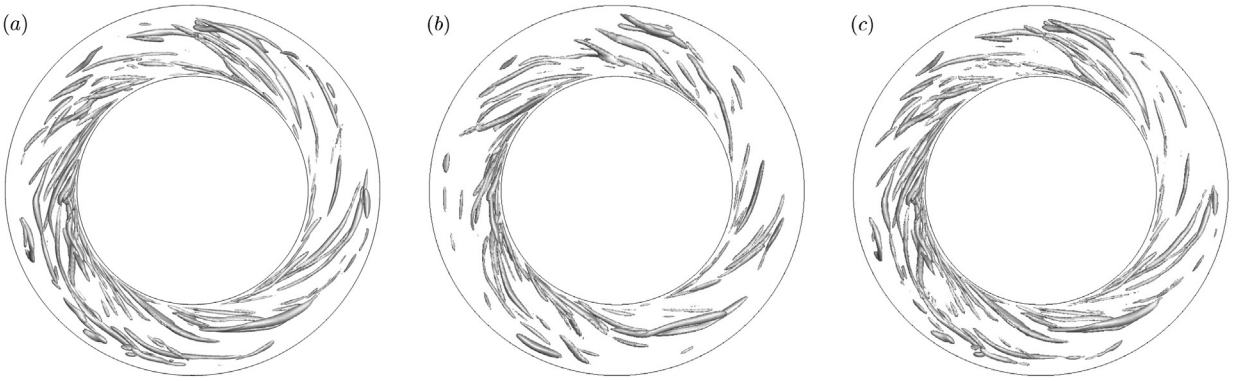

Fig. 16. Top view of vortical structures at $t = t_e$, visualized using the λ_2 vortex-identification criterion with threshold $\lambda_2 = -3.94$. (a) True flow field; (b) flow field generated by advancing the interpolated initial condition with the forward model; (c) estimated vortices by evolving the data-assimilated initial condition.

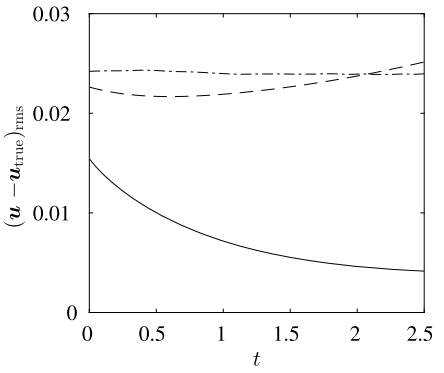

Fig. 17. Root-mean-square error of instantaneous velocity field. (– · –) Interpolation of observation data in space and time; (– – –) initial condition interpolated and advanced forward using the Navier-Stokes equations; (——) estimated initial condition from data assimilation is evolved in time. For simplicity t_s is represented as $t = 0$ here.

$$error(q) = \sqrt{\frac{1}{2\pi L}\int_0^L \int_0^{2\pi} (q - q_{\text{true}})^2 dz d\theta} \tag{67}$$

is plotted in Fig. 15. The estimated initial state achieves approximately 50% error reduction compared to the interpolated initial condition. At $t = t_e$, the data-assimilated state is one order of magnitude more accurate than advancing the interpolated initial condition using the forward model. In addition, the smooth error distribution of the assimilated field at $t = t_e$ demonstrates that the estimated field has improved accuracy throughout the domain, at and away from observation points.

A comparison of the vortical structures at $t = t_e$ is provided in Fig. 16. The structures are visualized using the λ_2 vortex identification criterion, defined as the intermediate eigenvalue of the tensor $S_{ik}S_{kj} + \Omega_{ik}\Omega_{kj}$, where S_{ij} and Ω_{ij} are the symmetric and anti-symmetric parts of the velocity gradient tensor [39]. Panel (a) visualizes the vortices of the true flow field. Most vortices are attached to the inner cylinder, and elongated along the azimuthal direction. Comparatively, appreciably less vortical structures are generated by advancing the interpolated initial state, as shown in panel (b). In panel (c), the estimated vortices from data assimilation are nearly identical to the true state in (a), which demonstrates the accuracy of our data-assimilation algorithm for augmenting under-resolved spatio-temporal measurements of turbulence. From a practical standpoint, these results are relevant to experiments where it is desirable for example to obtain accurate estimates of vorticity from coarse-scale observations of velocity.

Fig. 17 compares the time evolution of error in the flow fields from three simulations: In the first, the state is simply interpolated in space and time from the observations without use of the forward model; the error therefore remains constant throughout time since it is only incurred by the interpolation of the measurements. In the second, only the initial condition is interpolated, projected onto divergence-free space, and advanced using the Navier-Stokes equations; after a transient, the error starts to diverge due to the stochasticity of turbulence. Finally, in the third case, the estimated initial condition using the discrete-adjoint data-assimilation algorithm is evolved; the error decays with time and is appreciably smaller than the error in the first two cases.

4. Conclusions

The discrete adjoint of a fractional-step implementation of the incompressible Navier-Stokes equations in curvilinear coordinates was derived and implemented. The forward model uses volume fluxes for conservation, and hence the discrete adjoint algorithm is in terms of adjoint fluxes. Since each term in the discrete adjoint equations has its counterparts in the forward equations, the adjoint algorithm is as efficient as the forward one and retains the same computational performance and parallel implementation. The relation between the discrete adjoint variables and the well-established continuous adjoint was derived. In addition, a symmetric operator was proposed to project the outcome of the adjoint onto a solenoidal field. The implementations of forward and adjoint equations were validated by comparing their predictions of the growth rates of small-amplitude instability waves to linear stability theory. Furthermore, the forward-adjoint duality relation was demonstrated to be satisfied up to eight significant figures.

A discrete-adjoint data assimilation algorithm was proposed to predict the initial flow state from observations, by minimizing a cost function which is the difference between the observations and their estimates from the forward model. The gradient of the cost function is computed using the forward-adjoint loop. The minimization of the cost function is performed using L-BFGS, which is slightly modified using the projection operator in order to ensure that search directions and the estimated initial flow state are divergence-free. The data-assimilation algorithm was applied to reconstruct the initial state of (a) saturated wavy vortices and (b) turbulent circular Couette flows. In the first case, the measurements were the wall shear stresses on both cylinders. Even though multiple saturated states exist, the observations were sufficient for the algorithm to converge onto the correct state. The reconstruction quality of mean flow, energy spectra, and shapes of dominant modes were all satisfactory.

The turbulent configuration is to date the first application of adjoint-based data assimilation to the reconstruction of fully resolved turbulence in curvilinear coordinates. The observations were low-resolution data at 1/5120 the spatio-temporal resolution of direct numerical simulations, and the data assimilation algorithm achieved more than 50% error reduction within the assimilation window, and its accuracy is uniform throughout the flow field.

The two test cases thus demonstrate the fidelity of the discrete adjoint and the data-assimilation algorithms, their accurate estimation of the gradient of the cost function, and robustness in state estimation of turbulent flows. Future efforts will include applications of data assimilation to transitional flows. In that context, the onset of turbulence is very sensitive to small-amplitude initial disturbances that can be challenging to measure experimentally, and data assimilation can be adopted to infer the environmental state from observations.

Acknowledgements

The authors would like to acknowledge support from the Office of Naval Research (grant no. N00014-16-1-2542).

Appendix A. Derivation of discrete adjoint equations: a simple example

A simple example of deriving discrete adjoint equations is provided in this appendix. Consider the inviscid Burgers equation, discretized by forward Euler and upwind scheme,

$$\frac{u_j^{n+1}-u_j^n}{\Delta t}+\frac{(u_j^n)^2/2-(u_{j-1}^n)^2/2}{\Delta x}=0 \quad 1\le j\le N_x\,,\ 0\le n\le N-1 \tag{A.1}$$

The periodic boundary condition is applied at $j=1$ such that

$$u_0^n=u_{N_x}^n \tag{A.2}$$

In order to derive the adjoint equations, we define the Lagrangian function,

$$L=J-\sum_{n=0}^{N-1}\sum_{j=1}^{N_x}u_j^{\dagger(n+1)}\left(\frac{u_j^{n+1}-u_j^n}{\Delta t}+\frac{(u_j^n)^2/2-(u_{j-1}^n)^2/2}{\Delta x}\right) \tag{A.3}$$

The adjoint velocity $u_j^{\dagger(n+1)}$ is also interpreted as Lagrangian multiplier. Take the derivative of L with respect to u_j^n (when $1\le j\le N_x-1$ and $1\le n\le N-1$),

$$\frac{\partial L}{\partial u_j^n}=\frac{\partial J}{\partial u_j^n}-\left[u_j^{\dagger(n+1)}\left(\frac{-1}{\Delta t}+\frac{u_j^n}{\Delta x}\right)+u_{j+1}^{\dagger(n+1)}\left(\frac{-u_j^n}{\Delta x}\right)+u_j^{\dagger n}\frac{1}{\Delta t}\right]; \tag{A.4}$$

Note that the resulting expression depends on $u_j^{\dagger(n+1)}$, $u_{j+1}^{\dagger(n+1)}$ and $u_j^{\dagger n}$. When $j=N_x$, the $u_{j+1}^{\dagger(n+1)}$ term in (A.4) should be eliminated because $u_{N_x+1}^{\dagger(n+1)}$ is not defined in (A.3). Instead, the $u_1^{\dagger(n+1)}$ term should be included on account of the boundary condition (A.2),

$$\frac{\partial L}{\partial u_{N_x}^n} = \frac{\partial J}{\partial u_{N_x}^n} - \left[u_{N_x}^{\dagger(n+1)} \left(\frac{-1}{\Delta t} + \frac{u_{N_x}^n}{\Delta x} \right) + u_1^{\dagger(n+1)} \left(\frac{-u_{N_x}^n}{\Delta x} \right) + u_{N_x}^{\dagger n} \frac{1}{\Delta t} \right] \tag{A.5}$$

In order to render (A.5) consistent with (A.4), we can define $u_{N_x+1}^{\dagger(n+1)}$ to satisfy the adjoint periodic boundary condition,

$$u_{N_x+1}^{\dagger(n+1)} = u_1^{\dagger(n+1)} \tag{A.6}$$

Now (A.4) is also valid at $j = N_x$. By setting (A.4) to zero, we get the discrete adjoint equations,

$$\frac{u_j^{\dagger n} - u_j^{\dagger(n+1)}}{\Delta t} + \frac{u_j^n u_j^{\dagger(n+1)} - u_j^n u_{j+1}^{\dagger(n+1)}}{\Delta x} = \frac{\partial J}{\partial u_j^n} \quad 1 \le j \le N_x \,,\ 1 \le n \le N-1 \tag{A.7}$$

So far the only ingredient left for solving (A.7) is the "initial" condition at $n = N$. This is resolved by evaluating the derivative of (A.3) at $n = N$,

$$\frac{\partial L}{\partial u_j^N} = \frac{\partial J}{\partial u_j^N} - u_j^{\dagger N} \frac{1}{\Delta t} \tag{A.8}$$

and setting (A.8) to zero. The derived adjoint "initial" condition is,

$$u_j^{\dagger N} = \Delta t \frac{\partial J}{\partial u_j^N} \tag{A.9}$$

Finally, the gradient of the cost function ∇J is equal to the derivative at $n = 0$,

$$\frac{\partial L}{\partial u_j^0} = \frac{\partial J}{\partial u_j^0} - \left[u_j^{\dagger 1} \left(\frac{-1}{\Delta t} + \frac{u_j^n}{\Delta x} \right) + u_{j+1}^{\dagger 1} \left(\frac{-u_j^n}{\Delta x} \right) \right] \equiv \frac{u_j^{\dagger 0}}{\Delta t} \tag{A.10}$$

Note that the definition of $u_j^{\dagger 0}$ in (A.10) is the same as setting $n = 0$ in (A.7). Therefore, starting from (A.9) and marching the adjoint equations (A.7) backward in time to get $u_j^{\dagger 0}$, we can compute the gradient through the expression (A.10).

Appendix B. Forward and adjoint linear stability equations for circular Coutte flow

The linear stability equations for circular Couette flow and their continuous adjoint are provided in this appendix. We start from the incompressible Navier-Stokes equations in cylindrical coordinate,

$$\frac{1}{r}\frac{\partial (r v_r)}{\partial r} + \frac{1}{r}\frac{\partial v_\theta}{\partial \theta} + \frac{\partial v_z}{\partial z} = 0 \tag{B.1}$$

$$\begin{aligned} &\frac{\partial v_r}{\partial t} + v_r \frac{\partial v_r}{\partial r} + \frac{v_\theta}{r}\frac{\partial v_r}{\partial \theta} - \frac{v_\theta^2}{r} + v_z \frac{\partial v_r}{\partial z} = \\ &\quad - \frac{\partial p}{\partial r} + \frac{1}{Re}\left(\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial (r v_r)}{\partial r} \right) + \frac{1}{r^2}\frac{\partial^2 v_r}{\partial \theta^2} - \frac{2}{r^2}\frac{\partial v_\theta}{\partial \theta} + \frac{\partial^2 v_r}{\partial z^2} \right) \end{aligned} \tag{B.2}$$

$$\begin{aligned} &\frac{\partial v_\theta}{\partial t} + v_r \frac{\partial v_\theta}{\partial r} + \frac{v_\theta}{r}\frac{\partial v_\theta}{\partial \theta} + \frac{v_r v_\theta}{r} + v_z \frac{\partial v_\theta}{\partial z} = \\ &\quad - \frac{1}{r}\frac{\partial p}{\partial \theta} + \frac{1}{Re}\left(\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial (r v_\theta)}{\partial r} \right) + \frac{1}{r^2}\frac{\partial^2 v_\theta}{\partial \theta^2} + \frac{2}{r^2}\frac{\partial v_r}{\partial \theta} + \frac{\partial^2 v_\theta}{\partial z^2} \right) \end{aligned} \tag{B.3}$$

$$\begin{aligned} &\frac{\partial v_z}{\partial t} + v_r \frac{\partial v_z}{\partial r} + \frac{v_\theta}{r}\frac{\partial v_z}{\partial \theta} + v_z \frac{\partial v_z}{\partial z} = \\ &\quad - \frac{\partial p}{\partial z} + \frac{1}{Re}\left(\frac{1}{r}\frac{\partial}{\partial r}\left(r \frac{\partial v_z}{\partial r} \right) + \frac{1}{r^2}\frac{\partial^2 v_z}{\partial \theta^2} + \frac{\partial^2 v_z}{\partial z^2} \right) \end{aligned} \tag{B.4}$$

and decompose the velocity field into base flow and perturbation.

$$v_r = v_r' \,, \quad v_\theta = V_\theta(r) + v_\theta' \,, \quad v_z = v_z' \,, \quad p = P(r) + p' \tag{B.5}$$

We assume that the perturbation is infinitesimal, so the non-linear terms in (B.1)-(B.4) can be ignored. Note that the base flow is only a function of r and that the flow is periodic in azimuthal and axial directions, consequently, the perturbation takes the form,

$$v_j' = \hat{v}_j(r) e^{i(k_z z + k_m \theta - \omega t)} \,, \ j = r, \theta, z \tag{B.6}$$

Then the equations for the mode shape $\hat{v}_j(r)$ can be derived,

$$-i\omega\hat{v}_r = \frac{1}{Re}\left(\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial(r\hat{v}_r)}{\partial r}\right) - k^2\hat{v}_r - ik_m\frac{2}{r^2}\hat{v}_\theta\right) + \frac{2V_\theta}{r}\hat{v}_\theta - ik_m\frac{V_\theta}{r}\hat{v}_r - \frac{\partial\hat{p}}{\partial r} \tag{B.7}$$

$$-i\omega\hat{v}_\theta = \frac{1}{Re}\left(\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial(r\hat{v}_\theta)}{\partial r}\right) - k^2\hat{v}_\theta + ik_m\frac{2}{r^2}\hat{v}_r\right) - \left(\frac{dV_\theta}{dr} + \frac{V_\theta}{r}\right)\hat{v}_r - ik_m\frac{V_\theta}{r}\hat{v}_\theta - ik_m\frac{\hat{p}}{r} \tag{B.8}$$

$$-i\omega\hat{v}_z = \frac{1}{Re}\left(\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\hat{v}_z}{\partial r}\right) - k^2\hat{v}_z\right) - ik_m\frac{V_\theta}{r}\hat{v}_z - ik_z\hat{p} \tag{B.9}$$

$$0 = \frac{1}{r}\frac{\partial}{\partial r}(r\hat{v}_r) + ik_m\frac{\hat{v}_\theta}{r} + ik_z\hat{v}_z \tag{B.10}$$

where $k^2 = k_z^2 + k_m^2/r^2$. Solving this system of equations is an eigenvalue problem, where ω is the eigenvalue and $\mathbf{q} = [\hat{v}_r, \hat{v}_\theta, \hat{v}_z, \hat{p}]^T$ is the eigenvector or eigenmode. For clarity, equations (B.7)-(B.10) can be written in a matrix form,

$$\mathbf{Lq} = -i\omega\mathbf{Bq} \tag{B.11}$$

in which $\mathbf{L}$ is the r.h.s. operator of (B.7)-(B.10) and $\mathbf{B} = \mathrm{diag}\{1, 1, 1, 0\}$. For each wavenumber pair and Reynolds number, a spectrum of eigen-pairs $(\omega, \mathbf{q})$ can be obtained, and the mode with largest imaginary part of ω is the most unstable. These eigenmodes are used in §3.1 for validating the implementation of the forward equations and in §3.2 for generating saturated wavy vortices.

The adjoint operator $\mathbf{L}^\dagger$ is defined by:

$$\int_{r_i}^{r_o} (\mathbf{q}^\dagger)^H \mathbf{Lq}\, r dr = \int_{r_i}^{r_o} (\mathbf{L}^\dagger\mathbf{q}^\dagger)^H \mathbf{q}\, r dr \tag{B.12}$$

where H denotes the Hermitian transpose, and $\mathbf{q}^\dagger = [\hat{v}_r^\dagger, \hat{v}_\theta^\dagger, \hat{v}_z^\dagger, \hat{p}^\dagger]^T$ is the adjoint eigenmode. From (B.11) and (B.12), we derive the adjoint problem,

$$\mathbf{L}^\dagger\mathbf{q}^\dagger = i\omega^\dagger\mathbf{B}^\dagger\mathbf{q}^\dagger \tag{B.13}$$

where $\omega^\dagger = \omega^*$ and $\mathbf{B}^\dagger = \mathbf{B}$. A detailed expression of all the terms is provided as follows,

$$i\omega^\dagger\hat{v}_r^\dagger = \frac{1}{Re}\left(\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial(r\hat{v}_r^\dagger)}{\partial r}\right) - k^2\hat{v}_r^\dagger - ik_m\frac{2}{r^2}\hat{v}_\theta^\dagger\right) - \left(\frac{dV_\theta}{dr} + \frac{V_\theta}{r}\right)\hat{v}_\theta^\dagger + ik_m\frac{V_\theta}{r}\hat{v}_r^\dagger - \frac{\partial\hat{p}^\dagger}{\partial r} \tag{B.14}$$

$$i\omega^\dagger\hat{v}_\theta^\dagger = \frac{1}{Re}\left(\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial(r\hat{v}_\theta^\dagger)}{\partial r}\right) - k^2\hat{v}_\theta^\dagger + ik_m\frac{2}{r^2}\hat{v}_r^\dagger\right) + \frac{2V_\theta}{r}\hat{v}_r^\dagger + ik_m\frac{V_\theta}{r}\hat{v}_\theta^\dagger - ik_m\frac{\hat{p}^\dagger}{r} \tag{B.15}$$

$$i\omega^\dagger\hat{v}_z^\dagger = \frac{1}{Re}\left(\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\hat{v}_z^\dagger}{\partial r}\right) - k^2\hat{v}_r^\dagger\right) + ik_m\frac{V_\theta}{r}\hat{v}_z^\dagger - ik_z\hat{p}^\dagger \tag{B.16}$$

$$0 = \frac{1}{r}\frac{\partial}{\partial r}(r\hat{v}_z^\dagger) + ik_m\frac{\hat{v}_\theta^\dagger}{r} + ik_z\hat{v}_z^\dagger \tag{B.17}$$

For a given Reynolds number and wavenumber pair, every forward eigen-pair $(\omega, \mathbf{q})$ has its adjoint counterpart $(\omega^\dagger, \mathbf{q}^\dagger)$. The adjoint modes are used for validating the implementation of discrete adjoint Navier-Stokes equations in §3.1.

Appendix C. Accuracy of the forward-adjoint relation

In §3.1, the forward-adjoint relation was validated by comparing the gradients of the cost function obtained from the adjoint algorithm ∇J_A and a finite-difference approximation ∇J_F, and reporting the relative error ϵ:

$$\nabla J_A = \frac{\partial J}{\partial \boldsymbol{u}^0}; \qquad \nabla J_F = \frac{J\left(\boldsymbol{u}^0 + \Delta \boldsymbol{u}^0\right) - J\left(\boldsymbol{u}^0\right)}{\Delta \boldsymbol{u}^0}; \qquad \epsilon = \frac{\|\nabla J_A - \nabla J_F\|}{\|\nabla J_A\|}. \tag{C.1}$$

This appendix provides a theoretical analysis of the highest accuracy of the forward-adjoint relation that can be validated in this manner.

When ∇J_F is evaluated, round-off errors will be introduced to both the numerator and denominator. The difference between the two gradients can then be written as,

$$\frac{J\left(\boldsymbol{u}^0 + \Delta \boldsymbol{u}^0\right) - J\left(\boldsymbol{u}^0\right) + \delta_1 J(\boldsymbol{u}^0)}{\Delta \boldsymbol{u}^0 (1+\delta_2)} - \frac{\partial J}{\partial \boldsymbol{u}^0}(1+\delta_3) \tag{C.2}$$

where $\delta_1, \delta_2, \delta_3$ are round-off errors. If the forward and adjoint algorithms are implemented in double precision, these errors should be of the order of machine precision. Expanding $J\left(\boldsymbol{u}^0 + \Delta \boldsymbol{u}^0\right)$ using Taylor series, we obtain

$$\frac{J\left(\boldsymbol{u}^0\right) + \frac{\partial J}{\partial \boldsymbol{u}^0}\Delta \boldsymbol{u}^0 + \frac{1}{2}\frac{\partial^2 J}{\partial (\boldsymbol{u}^0)^2}(\Delta \boldsymbol{u}^0)^2 - J\left(\boldsymbol{u}^0\right) + \delta_1 J(\boldsymbol{u}^0)}{\Delta \boldsymbol{u}^0 (1+\delta_2)} - \frac{\partial J}{\partial \boldsymbol{u}^0}(1+\delta_3). \tag{C.3}$$

After some algebra, the leading terms in the above expression are,

$$\frac{1}{2}\frac{\partial^2 J}{\partial (\boldsymbol{u}^0)^2}\Delta \boldsymbol{u}^0 + \frac{\delta_1 J\left(\boldsymbol{u}^0\right)}{\Delta \boldsymbol{u}^0}. \tag{C.4}$$

When the perturbation $\Delta \boldsymbol{u}^0$ is not too small, the first term in (C.4) dominates, so the difference of the two gradients will monotonically decrease with $\Delta \boldsymbol{u}^0$. However, when the perturbation is smaller than a critical value,

$$(\Delta \boldsymbol{u}^0)_{cr} = \sqrt{2\delta_1 J(\boldsymbol{u}^0) \Big/ \frac{\partial^2 J}{\partial (\boldsymbol{u}^0)^2}}, \tag{C.5}$$

the second term in (C.4) becomes more dominant. As a result, the difference of the two gradients will monotonically increase when $\Delta \boldsymbol{u}^0$ is further reduced, and the critical value $(\Delta \boldsymbol{u}^0)_{cr} \propto \sqrt{\delta_1}$ is half of machine precision. Since our implementation is double precision, the critical perturbation $(\Delta \boldsymbol{u}^0)_{cr}$ is near single precision. And if the cost function value, gradient and the second-order derivatives are around $O(1)$, the smallest relative error ϵ in (C.1) should be approximately 10^{-8}. Both the critical perturbation and smallest relative error agree well with our validation results in Fig. 7.

In summary, only accuracy up to the square root of machine precision can be validated using the finite-difference approximation of the gradient. The error of the forward-adjoint relation in our implementation satisfies this criterion.

References

[1] M. Rosenfeld, D. Kwak, M. Vinokur, A fractional step solution method for the unsteady incompressible Navier-Stokes equations in generalized coordinate systems, J. Comput. Phys. 94 (1991) 102–137, https://doi.org/10.1016/0021-9991(91)90139-C.

[2] N. Nikitin, Characteristics of the leading Lyapunov vector in a turbulent channel flow, J. Fluid Mech. 849 (2018) 942–967, https://doi.org/10.1017/jfm.2018.418.

[3] A.R. Lewis, H.L. Swinney, Velocity structure functions, scaling, and transitions in high-Reynolds-number Couette-Taylor flow, Phys. Rev. E 59 (5) (1999) 5457–5466, https://doi.org/10.1103/PhysRevE.59.5457.

[4] M. Bilson, K. Bremhorst, Direct numerical simulation of Taylor-Couette flow, J. Fluid Mech. 579 (2007) 227–270, https://doi.org/10.1017/S0022112007004971.

[5] N.D. Sandham, L. Kleiser, The late stages of transition to turbulence in channel flow, J. Fluid Mech. 245 (1992) 319–348, https://doi.org/10.1017/S002211209200048X.

[6] T.A. Zaki, From streaks to spots and on to turbulence: exploring the dynamics of boundary layer transition, Flow Turbul. Combust. 91 (3) (2013) 451–473, https://doi.org/10.1007/s10494-013-9502-8.

[7] T.A. Zaki, P.A. Durbin, Mode interaction and the bypass route to transition, J. Fluid Mech. 531 (2005) 85–111, https://doi.org/10.1017/S0022112005003800.

[8] V. Mons, J.C. Chassaing, T. Gomez, P. Sagaut, Reconstruction of unsteady viscous flows using data assimilation schemes, J. Comput. Phys. 316 (2016) 255–280, https://doi.org/10.1016/j.jcp.2016.04.022.

[9] O. Pironneau, On optimum profiles in Stokes flow, J. Fluid Mech. 59 (1) (1973) 117–128, https://doi.org/10.1017/S002211207300145X.

[10] A. Jameson, Aerodynamic design via control theory, J. Sci. Comput. 3 (3) (1988) 233–260, https://doi.org/10.1007/BF01061285.

[11] A. Jameson, L. Martinelli, N. Pierce, Optimum aerodynamic design using the Navier-Stokes equations, Theor. Comput. Fluid Dyn. 10 (1998) 213–237, https://doi.org/10.1007/s001620050060.

[12] T.R. Bewley, P. Moin, R. Temam, DNS-based predictive control of turbulence: an optimal benchmark for feedback algorithms, J. Fluid Mech. 447 (2001) 179–225, https://doi.org/10.1017/S0022112001005821.

[13] R. Vishnampet, D.J. Bodony, J.B. Freund, A practical discrete-adjoint method for high-fidelity compressible turbulence simulations, J. Comput. Phys. 285 (2015) 173–192, https://doi.org/10.1016/j.jcp.2015.01.009.

[14] G. Papadakis, L. Lu, P. Ricco, Closed-loop control of boundary layer streaks induced by free-stream turbulence, Phys. Rev. Fluids 1 (2016) 043501, https://doi.org/10.1103/PhysRevFluids.1.043501.
[15] T.R. Bewley, B. Protas, Skin friction and pressure: the “footprints” of turbulence, Physica D 196 (2004) 28–44, https://doi.org/10.1016/j.physd.2004.02.008.
[16] D.P.G. Foures, N. Dovetta, D. Sipp, P.J. Schmid, A data-assimilation method for Reynolds-averaged Navier-Stokes-driven mean flow reconstruction, J. Fluid Mech. 759 (2014) 404–431, https://doi.org/10.1017/jfm.2014.566.
[17] S.K. Nadarajah, A. Jameson, Studies of continuous and discrete adjoint approaches to viscous automatic aerodynamic shape optimization, in: 15th AIAA Computational Fluid Dynamics Conference, Fluid Dynamics and Co-Located Conferences, 2001.
[18] J.E.V. Peter, R.P. Dwight, Numerical sensitivity analysis for aerodynamic optimization: a survey of approaches, Comput. Fluids 39 (2010) 373–391, https://doi.org/10.1016/j.compfluid.2009.09.013.
[19] P. Frank, G. Shubin, A comparison of optimisation-based approaches for a model computational aerodynamics design problem, J. Comput. Phys. 98 (1992) 74–89, https://doi.org/10.1016/0021-9991(92)90174-W.
[20] K. Mani, D.J. Mavriplis, An unsteady discrete adjoint formulation for two-dimensional flow problems with deforming meshes, AIAA J. 46 (6) (2008) 1351–1364, https://doi.org/10.2514/1.29924.
[21] E.J. Nielsen, B. Diskin, Discrete adjoint-based design for unsteady turbulent flows on dynamic overset unstructured grids, AIAA J. 51 (6) (2013) 1355–1373, https://doi.org/10.2514/1.J051859.
[22] M.F. de Pando, D. Sipp, P.J. Schmid, Efficient evaluation of the direct and adjoint linearized dynamics from compressible flow solvers, J. Comput. Phys. 231 (2012) 7739–7755, https://doi.org/10.1016/j.jcp.2012.06.038.
[23] L. Hascoet, R. Greborio, V. Pascual, Computing Adjoints by Automatic Differentiation with TAPENADE, Springer, 2005.
[24] A. Gronskis, D. Heitz, E. Mémin, Inflow and initial conditions for direct numerical simulation based on adjoint data assimilation, J. Comput. Phys. 242 (2013) 480–497, https://doi.org/10.1016/j.jcp.2013.01.051.
[25] A. Griewank, Achieving logarithmic growth of temporal and spatial complexity in reverse automatic differentiation, Optim. Methods Softw. 1 (1) (1992) 35–54.
[26] M. Rosenfeld, D. Kwak, M. Vinokur, Development of a Fractional-Step Method for the Unsteady Incompressible Navier-Stokes Equations in Generalized Coordinate Systems, Tech. rep., NASA-TM-103912 (11), 1992.
[27] T. Jelly, S. Jung, T.A. Zaki, Turbulence and skin friction modification in channel flow with streamwise-aligned superhydrophobic surface texture, Phys. Fluids 26 (2014) 095102, https://doi.org/10.1063/1.4894064.
[28] T.A. Zaki, J.G. Wissink, W. Rodi, P.A. Durbin, Direct numerical simulations of transition in a compressor cascade: the influence of free-stream turbulence, J. Fluid Mech. 665 (2010) 57–98, https://doi.org/10.1017/S0022112010003873.
[29] J. Lee, H.J. Sung, T.A. Zaki, Signature of large-scale motions on turbulent/non-turbulent interface in boundary layers, J. Fluid Mech. 819 (2017) 165–187, https://doi.org/10.1017/jfm.2017.170.
[30] P.J. Schmid, D.S. Henningson, Stability and Transition in Shear Flows, vol. 142, Springer Science & Business Media, 2012.
[31] J. Nocedal, Updating quasi-Newton matrices with limited storage, Math. Comput. 35 (151) (1980) 773–782, https://doi.org/10.2307/2006193.
[32] J.J. Moré, D.J. Thuente, Line search algorithms with guaranteed sufficient decrease, ACM Trans. Math. Softw. 20 (3) (1994) 286–307, https://doi.org/10.1145/192115.192132.
[33] X. Mao, H.M. Blackburn, S.J. Sherwin, Calculation of global optimal initial and boundary perturbations for the linearised incompressible Navier–Stokes equations, J. Comput. Phys. 235 (2013) 258–273.
[34] X. Mao, H. Blackburn, S. Sherwin, Nonlinear optimal suppression of vortex shedding from a circular cylinder, J. Fluid Mech. 775 (2015) 241–265, https://doi.org/10.1017/jfm.2015.304.
[35] S.J. Lee, T.A. Zaki, Simulations of natural transition in viscoelastic channel flow, J. Fluid Mech. 820 (2017) 232–262, https://doi.org/10.1017/jfm.2017.198.
[36] R. Ostilla, R.J.A.M. Stevens, S. Grossmann, R. Verzicco, D. Lohse, Optimal Taylor-Couette flow: direct numerical simulations, J. Fluid Mech. 719 (2013) 14–46, https://doi.org/10.1017/jfm.2012.596.
[37] M. Fisher, H. Auvinen, Long window 4d-var, in: Seminar on Data Assimilation for Atmosphere and Ocean, 6–9 September 2011, ECMWF, ECMWF, Shinfield Park, Reading, 2012, pp. 189–202, https://www.ecmwf.int/node/9410.
[38] T. Suzuki, Y. Hasegawa, Estimation of turbulent channel flow at $Re_\tau = 100$ based on the wall measurement using a simple sequential approach, J. Fluid Mech. 830 (2017) 760–796.
[39] J. Jeong, F. Hussain, On the identification of a vortex, J. Fluid Mech. 285 (1995) 69–94, https://doi.org/10.1017/S0022112095000462.