



Craig Thompson · Uttam Cadambi Padmanaban ·
Bharathram Ganapathisubramani · Sean Symon

The effect of variations in experimental and computational fidelity on data assimilation approaches

Received: 24 October 2023 / Accepted: 4 June 2024
© The Author(s) 2024

Abstract We conduct a comprehensive analysis of two data assimilation methods: the first utilizes the discrete adjoint approach with a correction applied to the production term of the turbulence transport equation, preserving the Boussinesq approximation. The second is a state observer method that implements a correction in the momentum equations alongside a turbulence model, both applied to fluid dynamics simulations. We investigate the impact of varying computational mesh resolutions and experimental data resolutions on the performance of these methods within the context of a periodic hill test case. Our findings reveal the distinct strengths and limitations of both methods, which successfully assimilate data to improve the accuracy of a RANS simulation. The performance of the variational model correction method is independent of input data and computational mesh resolutions. The state observer method, on the other hand, is sensitive to the resolution of the input data and CFD mesh.

Keywords Data assimilation · Separated flow · Turbulence modelling · State observer · Variational method · Discrete adjoint

1 Introduction

Data assimilation in fluid dynamics is used to refine models, quantify uncertainty, optimize experiments, and minimize error propagation. It ensures that numerical simulations and predictions align with real-world observations, thereby alleviating experimental shortcomings such as incomplete or noisy data and computational shortcomings such as incorrect boundary conditions or modelling assumptions. With the use of particle image velocimetry (PIV), improved assimilation methods became more applicable to fluid mechanics. Data assimilation (DA) was first introduced within meteorology [1, 2], where real-world observations were used to improve

Communicated by Kilian Oberleithner.

Craig Thompson and Uttam Cadambi Padmanaban have contributed equally to this work.

C. Thompson (✉) · U. C. Padmanaban (✉) · B. Ganapathisubramani · S. Symon
Aero and Astro, University of Southampton, University Rd, Southampton, Hampshire SO17 1BJ, Great Britain
E-mail: ct4g16@soton.ac.uk

U. C. Padmanaban
E-mail: ucp1n22@soton.ac.uk

B. Ganapathisubramani
E-mail: g.bharath@soton.ac.uk

S. Symon
E-mail: sean.symon@soton.ac.uk

the understanding and predictive capabilities of meteorological simulations. There are three groups of DA methods: variational methods [3–11], sequential methods also known as Kalman filtering [12–17] and state observer methods [18–25]. An in-depth comparison of variational methods and sequential methods is shown by Mons et al. [5] and an understanding of the application of sequential and state observer methods is given by Hayase [26].

To overcome the Reynolds closure problem, turbulence models are employed. These turbulence models are inherently incorrect but provide a good estimate of the physics of particular flows. There are three data assimilation methods for improving the predictions of fluid problems to overcome the Reynolds closure problem. The first involves determining the unknown Reynolds stresses directly from measurement data, shown by Kellaris et al. [27]. The second method implements and directly corrects a turbulence model by means of tuning a field or constant within the model, and the final method implements but indirectly corrects a turbulence model by means of an additional term within the governing equations. Work by Franceschini et al. [8] compared the final two approaches utilising a variational data assimilation approach.

The variational method (also known as 3D/4D Var) modifies uncertain parameters in the numerical model by minimizing the discrepancy between the output of such a model and experimental measurements. The discrepancy is formulated as a cost function and gradient-based optimization methods are used to find the minimum. Foures et al. [6] successfully used the variational method to optimise unknown Reynolds stress gradients in the Reynolds-Averaged Navier–Stokes (RANS) equations for a flow past a circular cylinder at Reynolds number $Re = 150$. Direct numerical simulation (DNS) data with varying resolutions were used as reference measurements. Satisfactory reconstruction of mean velocity was achieved. Symon et al. [7] applied a similar methodology to an idealized airfoil at $Re_c = 13, 500$, using planar PIV data as input measurements. Varying the input data resolution affected mean-velocity reconstruction, but its impact on other quantities like skin friction (C_f) and pressure coefficient (C_p) was not investigated.

Franceschini et al. [8] extended the methodology of Foures et al. [6] by assimilating reference data into the RANS equations closed with the Spalart-Allmaras (SA) turbulence model [28]. This allowed the authors to perform DA more efficiently for a backward-facing step at $Re = 28275$ since Symon et al. [7] faced difficulties with the well-posedness of the steady Navier–Stokes equations at high Reynolds numbers. Franceschini et al. [8] compared two approaches that involved tuning a source term either in the momentum equations or the turbulence equation. The momentum source term significantly improved reconstruction when full-field input data was available, while improvement in reconstruction observed with a correction term applied to the turbulence transport equation was less accurate but relatively insensitive to input data resolution. Franceschini et al. [8] also examined skin-friction (C_f) and pressure coefficient (C_p) along the bottom wall, with the momentum correction showing superior performance over the turbulence transport correction. Cato et al. [9] reached a similar conclusion after a comprehensive comparison of six different correction terms across three flow configurations.

The studies by Foures et al. [6], Symon et al. [7], and Franceschini et al. [8] utilized a continuous adjoint method for DA which involves linearizing and discretizing the PDE while reusing the primal solver. In contrast, the discrete approach, as demonstrated by Kenway et al. [29], formulates adjoint equations post-discretization, achieving potential machine precision gradient calculation accuracy. Several studies provide a good comparison between these methods [30–32]. Brenner et al. [10] applied a discrete adjoint method to correct the eddy viscosity field in RANS simulations using a $k - \epsilon$ turbulence model. They used a frozen eddy viscosity approach and optimized a spatially varying scalar multiplier. This approach is constrained by the Boussinesq approximation and requires regularization to promote a smooth parameter field and C_f . However, the gradient accuracy remains a challenge. Recently, Brenner et al. [33] extended the work in Ref. [10] by improving the accuracy of their algorithm and including a momentum source term correction. A promising outcome was that the accuracy of mean-velocity reconstruction was unaffected when coarse input data was considered.

A more recent tool that employs the discrete adjoint method but has been shown to produce high-accuracy gradients is DAfoam [34,35]. In addition to being open source, it seamlessly integrates primal solvers from OpenFOAM with the discrete adjoint method and an optimizer framework, ideal for variational DA. However, it is worth noting that DAfoam lacks a projection and smoothing operation as described in [6,7]. This absence may pose challenges when working with experimental data that has a different resolution compared to the computational mesh.

A DA method that circumvents the complexity of the variational method is the state observer method. Initially developed by Luenberger [36], the state observer method was first implemented into the world of fluids by Hayase [18] and utilises control theory to modify part of a system such that it converges to a known optimal state. The modification to the initial system or equations is generally the addition of a forcing term

that is proportional to the error between the result of the system and the optimal state. This forcing term can be considered as a feedback loop pushing the system towards the optimal state. The implementation of the state observer method by Nisugi, Hayase and Yamagata [19–21] for the flow around a cylinder found that the most improved locations were downstream of the cylinder and that very close to the cylinder surface the error was largest. When modifying the computational domain it was shown that a feedback term reduces the error even when using a coarse computational grid and it is observed that at higher feedback rates and higher experimental spatial resolution, the reduction in error increases. However, even though the results were validated with pressure measurements, there was no indication of the pressure field or surface pressures obtained.

The state observer method can be considered as a proportional-integral-differential control, where only the proportional component is utilized. Imagawa and Hayase [22] use an additional forcing term in the discretized Navier–Stokes (NS) equation, while Zauner et al. [23] incorporate an additional “nudging” term in the momentum part of the unsteady Reynolds-Averaged Navier–Stokes (URANS) and Saredi et al. [24] introduce a proportional-integral forcing term to the momentum part of the RANS equations. In each case, these terms are proportional to the discrepancy in the optimal state and the current computation. Both studies found that increasing the feedback gain improved convergence time up to a limit where the system was then destabilised and the error would increase. Similar to previous studies, higher spatial resolution leads to greater improvements in the assimilated velocity. Nevertheless, these studies were focused on the velocity fields, whereas the pressure of the surface as well as the surrounding field were not evaluated. This research seeks to address this limitation and improve the consideration of “reconstructed” variables, such as pressure, in state observer methods.

From the above discussion, there is a clear need for a robust DA methodology capable of operating on steady-state cases. We introduce a new discrete adjoint DA algorithm that is entirely implemented in OpenFOAM. This new variational algorithm also introduces a way to handle sparse data where grid conformity is ensured either by interpolation or by using a projection operator. We compare the performance of this new DA technique to a simpler state observer approach. There is a particular interest in the sensitivity of the velocity field reconstruction to the input data, which will be averaged to mimic experimental data, and the computational mesh which other studies typically keep fixed. The two methods are also compared with respect to the reconstructed variables, which are less often considered, such as surface pressure, skin friction coefficient and Reynolds stress gradients.

In the following sections, we present a comprehensive examination of the two aforementioned DA methods. Section 2 outlines the mathematical frameworks and implementation details of these methods, shedding light on the core principles that underpin their performance. Section 3 describes the periodic hill test case, input data and baseline computations. Section 4 focuses on resolution effects for the assimilated velocity field while Sect. 5 investigates the reconstructed variables in greater detail. Finally, Sect. 6 concludes our findings, offering practical implications for researchers in the field.

2 Data assimilation methods

Within this section, a comprehensive explanation of the DA techniques employed to improve the periodic hill test case are discussed. In Sect. 2.1 we present the RANS equations for an incompressible fluid, in Sect. 2.2 we describe the variational method where a modification to the production term of the eddy viscosity is made and in Sect. 2.3 we describe the state observer method which utilizes a forcing term in the momentum equations independent of the turbulence model.

2.1 Reynolds-averaged Navier–Stokes

The RANS equations for an incompressible fluid are given by,

$$\frac{\partial U_i}{\partial x_i} = 0, \tag{1}$$

$$U_j \frac{\partial U_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial P^*}{\partial x_i} + \frac{\partial}{\partial x_j} \left(\nu \frac{\partial U_i}{\partial x_j} \right) - \frac{\partial \tau_{ij}}{\partial x_j}, \tag{2}$$

where U_i and P^* are the mean velocity components and pressure respectively, ρ is the density of the fluid, ν is the kinematic viscosity, τ_{ij} is the Reynolds stress tensor and x_i , the spatial co-ordinates. The Reynolds

stress tensor $\tau_{ij} = \overline{u'_i u'_j}$ is the averaged outer product of the fluctuating velocity components that presents the problem of closure. To model this term, the mean flow components are used within the Boussinesq hypothesis alongside the SA turbulence model.

2.2 Variational method

Within this section we introduce a data assimilation algorithm that employs a variational approach to directly modify the production term of the SA turbulence model, which is hereafter referred to as the variational method. The algorithm uses the Field Inversion and Machine Learning (FIML) framework devised by Singh et al. [37]. The production term of the SA turbulence model is augmented with a spatially varying scalar field $\beta(x, y)$ and is given by,

$$\frac{D\tilde{v}}{Dt} = \beta(x, y)P(\tilde{v}, \mathbf{w}) + T(\tilde{v}, \mathbf{w}) - D(\tilde{v}, \mathbf{w}), \quad (3)$$

where \mathbf{w} is the vector of state variables such as mean velocity, pressure and momentum flux and P , T and D are the production, transport and dissipation terms, respectively. The objective function, representing a discrepancy between the velocity fields of the high-fidelity data and RANS simulation using the SA model, is given by

$$f(\mathbf{u}, \beta) = \frac{1}{2} \|\mathcal{Q}(\mathbf{u}, \beta) - \tilde{\mathbf{Q}}\|_Q^2, \quad (4)$$

where $\tilde{\mathbf{Q}}$ is a set of high-accuracy measurements such as experimental data or data extracted from DNS. Operator $\mathcal{Q}(\cdot)$ extracts the computational data in such a way that $\mathcal{Q}(\mathbf{u}) \in Q$ is a projection of the computational mean velocity to the measurement space Q . $\|\cdot\|_Q$ is the generic norm in the measurement space.

Variational DA is now formulated as an optimization problem where the goal is to minimize an objective function subject to some constraints. This is mathematically written as

$$\min_{\mathbf{w} \in \mathbb{R}^{n_w}, \beta \in \mathbb{R}^{n_\beta}} f(\mathbf{w}, \beta), \quad (5)$$

$$\text{s.t. } R(\mathbf{w}, \beta) = 0, \quad (6)$$

$$\beta_L \leq \beta \leq \beta_U, \quad (7)$$

where n_β is the size of the design vector, n_w is the size of the state vector, R is the governing equations that serve as constraints and β_L and β_U denote the lower and upper bounds, respectively, for the design variable. For our test case, R represents the residual function of the NS equations. Equation 5 is a non-linear, constrained minimization problem with equality and bound constraints and can be solved using popular gradient-based techniques.

Gradient-based optimization techniques require the total derivative of the objective function with respect to the design variable (hereafter referred to as sensitivity). An efficient way to compute the sensitivity is by employing an adjoint method, which ensures that the computational cost remains independent of the number of design variables [38]. We use the discrete adjoint method in this study for computing the sensitivities. If f and R are a univariate representation of the objective and residual functions, respectively, the sensitivity can be computed using

$$\frac{df}{d\beta} = \frac{\partial f}{\partial \beta} - \psi^T \frac{\partial \mathbf{R}}{\partial \beta}, \quad (8)$$

where ψ^T is the transpose of the adjoint vector. The detailed derivation can be found in [29]. DAfoam is used to obtain the sensitivity. DAfoam's source code is enriched with AD-forward (ADF) and AD-reverse (ADR) implementations using CoDiPack [39], enabling machine-precision gradient accuracy [29].

Once the sensitivity is obtained, the optimization is carried out by using an interior point (also called a barrier) method with a backtracking line-search filter to solve the constrained minimization problem defined in Eq. 5. The interior-point method solves a sequence of barrier problems [40]. The original problem is reformulated by combining the objective function and the bound constraints along with a barrier parameter into what is called the barrier objective function. We use the interior point method implemented in IPOPT [41]. It

has provisions for second-order correction and feasibility restoration. Convergence is determined based on the satisfaction of the Karush-Kuhn-Tucker (KKT) condition up to a user specified tolerance. Most importantly, it is free and open source.

We use a cell-volume weighted averaging operator \mathcal{Q} to project the computational mean velocity data onto the synthetic PIV grid. This is done to ensure that the discrepancy field is calculated on the synthetic PIV grid. However, the adjoint solution is forced on the computational grid which necessitates the requirement of a smoothing operator $\hat{\mathcal{Q}}$ to transfer the computed discrepancy field to the computational grid. The experimental and computational data are stored on topologically different meshes and cell-cell intersections are taken into consideration during the projection and smoothing operations. We implement the projection and smoothing operations in DAfoam with the help of the OpenFOAM function `interVol()` that obtains the intersection volume between two cells of different meshes. We also implement a custom objective function that works in conjunction with the projection and smoothing operations. This implementation was possible only because of the open-source and modular nature of DAfoam and the details can be found in Appendix A.

2.3 State observer method

Within this section we introduce a data assimilation algorithm that employs a state observer methodology to directly modify an additional term in the RANS momentum equations (independent from turbulence model), which is hereafter referred to as the state observer method. The state observer method introduces a forcing term into Eq. 2 denoted as F_i . For each iteration of the state observer method, the modified RANS equations are solved within OpenFOAM employing the SA turbulence model. The calculation of the forcing term, as expressed in Eq. 9, is determined by summing the product of a proportional gain K_p and the difference between the projected velocity computed in the previous time step $\mathcal{Q}(u_i^{n-1})$ and the target velocity U_i to the forcing term from the previous time step F_i^{n-1}

$$F_i^n = F_i^{n-1} + K_p[\mathcal{Q}(u_i^{n-1}) - U_i]. \quad (9)$$

The method for computing the forcing term draws from a concept in control theory known as proportional control. However, a subtle adjustment is incorporated with the inclusion of the forcing term of the previous iteration F_i^{n-1} to ensure that the calculated forcing term builds from the previous computational result, as shown by Saredi et al. [24]. To ensure the measurement data remains as accurate as possible, the computational flow variables are projected onto the measurement domain given by the operator $\mathcal{Q}(\cdot)$. For all cases when moving from the computational domain to the measurement domain, the data is being down-sampled. Therefore, an interpolant is constructed by triangulating the input data with a Delaunay triangulation, and on each triangle performing linear barycentric interpolation with the use of the function `griddata` from the python library `scipy`.

It has been demonstrated in literature that the proportionality constant plays a crucial role in achieving both computational efficiency and solution accuracy. Increasing the value of K_p leads to faster convergence with lower error in the solution. However, when K_p becomes excessively large, the forcing term modifies the momentum equations too aggressively, causing the solution to become unstable. A preliminary investigation revealed that a gain of $K_p = 10^{-4}$ achieves the highest level of accuracy with the fewest iterations. More details can be found in Appendix B.

The forcing term is computed on the measurement domain, but it needs to be projected back onto the computational domain. From a preliminary investigation into different interpolation methods, it is found that a method which guarantees that the interpolated forcing term is continuously differentiable at all locations provides a more accurate assimilation. A continuous interpolation approach is more beneficial as it provides a regularisation to the forcing not included in the state observer method. Hence, a trivariate Clough-Tocher interpolation method [42] is implemented whereby the interpolant is constructed by triangulating the input data with a Delaunay triangulation and constructing a piecewise cubic interpolating Bezier polynomial on each triangle using the cubic argument in `scipy`'s function `griddata`. To ensure forcing does not occur in regions where data is unavailable, a value of zero is assigned to any points located outside of the measurement domain. To implement the forcing term within the momentum equation the function `vectorCodedSource` is included within OpenFOAM's `fvoptions` dictionary. It must be noted that the implementation of the momentum source term within OpenFOAM is as an absolute variable, hence it is divided by the volume of the cell. Therefore within `vectorCodedSource` the forcing term is pre-multiplied by the cell volume.

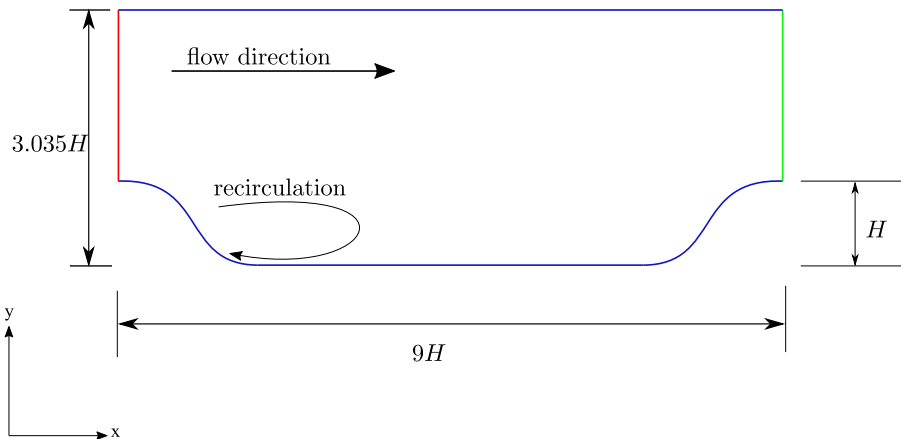


Fig. 1 Graphical description of the periodic hill case parameterized by H with inlet (—), outlet (—) and walls, top and bottom (—) shown. The flow direction and re-circulation zone have also been depicted (color figure online)

A key point that needs emphasizing here is the difference in the nature of corrections applied by the variational and state observer methods. The state observer method applies a forcing to the momentum equations as opposed to the variational method where the forcing is applied in the turbulence transport equation. Being within the confines of the Boussinesq approximation limits the flexibility of the variational method as was also reported in Franceschini et al. [8]. The state observer method escapes the confines of the Boussinesq approximation and is therefore more flexible.

3 Description of test case

This section describes the flow over a periodic hill which serves as a test case for the data assimilation methods. In Sect. 3.1 the details of the periodic hill geometry are explained. In Sect. 3.2 the generation of synthetic input data is described and in Sect. 3.3 the CFD domain and solution procedures are explained.

3.1 Periodic hill flow details

The canonical periodic hill is a good test case since it contains flow physics that most turbulence models struggle to capture accurately. These include flow separation, re-circulation, and re-attachment. As seen in Fig. 1, the geometry consists of a channel with a flat top wall and periodic hill of height H separated by a valley on the bottom wall. The hill normalised length and height of the channel is $L_x/H = 9$ and $L_y/H = 3.035$, respectively.

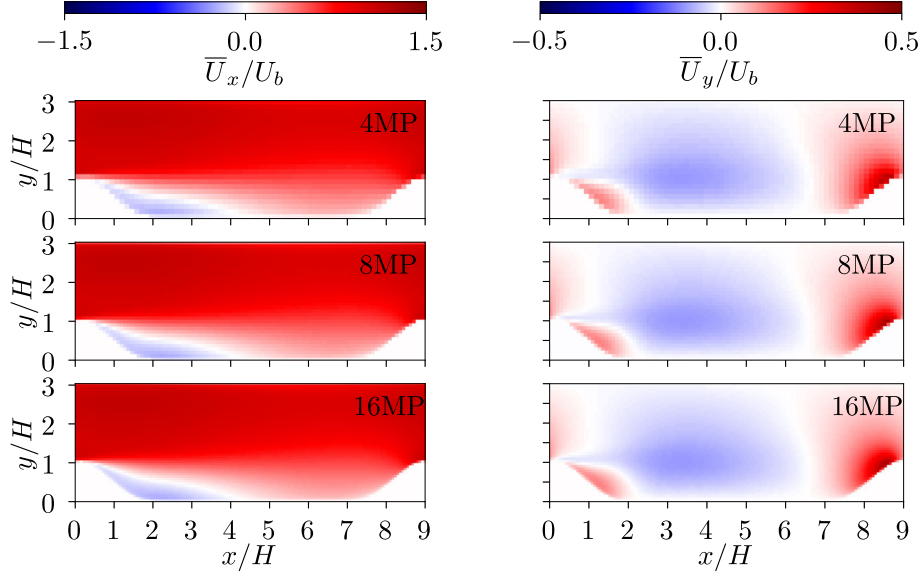
3.2 Synthetic PIV

To test the DA methods on input data that is representative of PIV performed in the water tunnel, we generate synthetic PIV fields from a publicly available DNS database of parameterized periodic hill geometry, found in the work by Xiao et al. [43], available from the following GitHub repository <https://github.com/xiaoh/para-database-for-PIML.git>. The DNS database was generated for the purpose of development and validation of data-driven models. Using this database three PIV experiments are created such that three different vector resolutions of synthetic PIV fields are computed. The hypothetical experimental setup ensures that the Reynolds number is consistent with the DNS data set at $Re = 5600$. The experimental setup was designed to be operated within a closed channel of water at 0.112 ms^{-1} with a test section of length, height and hill height of 0.45 m, 0.15 m and 0.05 m, respectively. Table 1 presents the outcomes of utilizing three different cameras (available on the market) with various lenses and distances, resulting in three distinct image resolutions.

To generate the synthetic PIV vector fields the DNS data set is formatted as a 271, 262 point unstructured mesh, which is naturally interpolated onto the structured 2272×1704 , 3264×2448 and 4608×3456

Table 1 Hypothetical experimental PIV parameters

Name	Unit	4 MP case	8 MP case	16 MP case
Pix density	$\text{pix}/\text{mm}^{-1}$	5.05	7.25	10.24
No. vectors	–	1572	3304	6542
Interrogation window	m	$6.34 \times 10^{-3} H$	$4.41 \times 10^{-3} H$	$3.13 \times 10^{-3} H$

**Fig. 2** Streamwise and wall-normal velocity components for PIV fields with 4MP, 8MP, and 16MP camera resolutions

pixel meshes of each theoretical camera (to represent pseudo-particles). Pixel locations above and below the experimental set-up were padded with zeros, mimicking a particle image. Locations up and downstream of the experimental set-up were replaced by opposite up or downstream points, to represent a cyclic boundary. A moving average with a window size of 32×32 is utilized to simulate a standard cross-correlation window employed in PIV processing. Figure 2 shows the resulting synthetic PIV vector fields. A significant difference in resolution is observed between each generated data set.

3.3 Baseline computations

DA is performed on three distinct computational meshes. Each mesh is characterized by a progressively increasing mesh density achieved by augmenting the number of grid points in both the streamwise and wall-normal directions. The coarsest computational mesh is chosen such that it contains more cells than that of the highest resolution experimental case. Figure 3a shows the initial coarse mesh configuration which consists of a total of 7200 computational cells, distributed with 120 cells in the streamwise direction and 60 cells in the wall-normal direction, respectively. Subsequently, this coarse mesh is refined as shown in Fig. 3b, wherein the number of grid points in the streamwise and wall-normal directions is increased to 160 and 100 cells, respectively. This refinement results in a mesh containing 16,000 computational cells. The highest level of mesh density is achieved in the final computational mesh shown in Fig. 3c, which features 21,600 cells distributed with 180 cells in the streamwise direction and 120 cells in the wall-normal direction. It is noteworthy that this progression entails a doubling and tripling of the mesh density relative to the initial coarse computational mesh. It is ensured that $y^+ < 1$ along the top and bottom walls through stretching applied with decreasing cell volumes as the walls are approached. By employing these three distinct mesh configurations, we were able to systematically investigate and analyze the impact of mesh density on our simulations.

The SA model is used as the baseline which will be improved by utilizing the state-observer and variational methods. The simulations are performed on the open-source finite-volume method (FVM) package OpenFOAM [44]. To solve the RANS equations the Semi-Implicit Method for Pressure Linked Equations (SIMPLE) [45] is

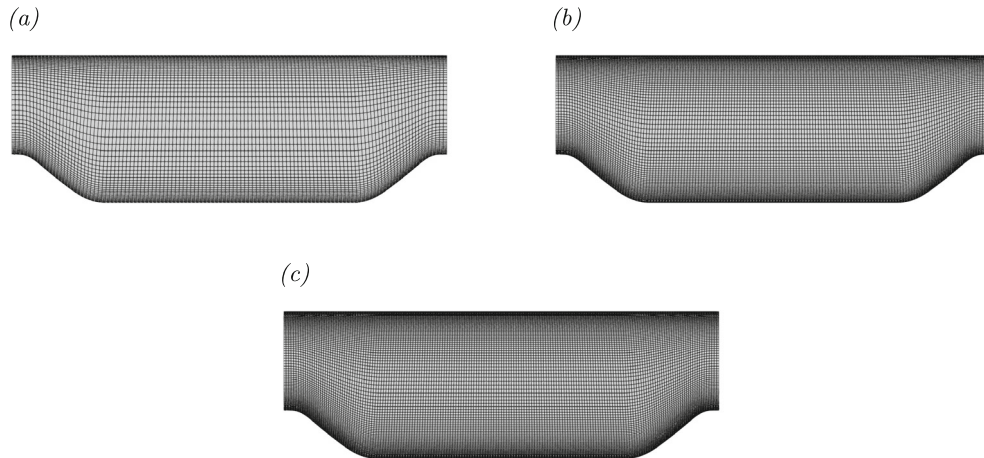


Fig. 3 Computational meshes with **a** 7200, **b** 16,000, and **c** 21,600 cells

employed using OpenFOAM’s inbuilt `simpleFOAM` solver. The gradients are calculated using a second-order accurate central differencing scheme. The velocity term is discretized using a second-order upwind method, while all other flow variables are discretized using a first-order upwind method. Each matrix equation is solved using the Gauss-Seidel method. Convergence is determined based on a tolerance of 10^{-6} for the residual of pressure and velocity components.

The Reynolds number for this investigation is defined based on H and bulk velocity U_B on the inlet face given by,

$$U_B = \frac{1}{2.035H} \int_H^{3.035H} u_x(y) dy, \quad (10)$$

where $u_x(y)$ is the wall-normal velocity profile of the streamwise mean velocity. The bulk velocity is maintained by adding a pressure gradient as a body force to the momentum equation. The natural direction of the flow is designated to be along the positive streamwise direction with the left edge serving as the inlet and the right edge serving as an outlet. The presence of curvature at the inlet results in an adverse pressure gradient that causes flow separation at $\approx 0.17H$ and reattachment on the bottom wall at $x/H = 5.0$ [46]. The simulations are performed at a bulk Reynolds number of $Re_B = 5600$ which is set by fixing $H = 1$ m, $\nu = 5 \times 10^{-6} \text{ m}^2 \text{ s}^{-1}$ and $U_B = 0.028 \text{ ms}^{-1}$. The inlet and outlet boundaries are periodic in nature, with no-slip top and bottom walls. The front and back faces of the domain are designated with a Neumann boundary for all variables and a no flow condition for velocity (designated as symmetry boundary condition in OpenFOAM) essentially treating the case as two-dimensional.

4 Velocity field assimilation

In this section, the variational and state observer methods are applied to a periodic hill test case. Our investigation encompasses a range of computational mesh and experimental resolutions, shedding light on the behavior and performance of these methods under varying conditions. Through an examination of velocity contours shown in Figs. 4 and 5, reattachment locations displayed in Table 2, streamwise velocity contours shown in Fig. 6 and an L_1 error norm, we provide valuable insights into the strengths and limitations of each approach.

Examples of both variational and state observer methods are presented in Fig. 4 along with the initial RANS solution for the periodic hill case utilizing the SA turbulence model. This corresponds to setting the scalar field $\beta = 1$ in the variational method and $F_i = 0$ in the state observer method.

The DNS solution of the same periodic hill case is also presented. Both methods exhibit improvements when compared to the baseline, shown by the contours within the freestream. Notably, the recirculation region aligns more closely with that of the DNS, as indicated by the dividing streamline.

It must be noted that both the variational and state observer methods exhibit a relatively low sensitivity to variations in the experimental resolution when comparing the velocity contours of the freestream. This

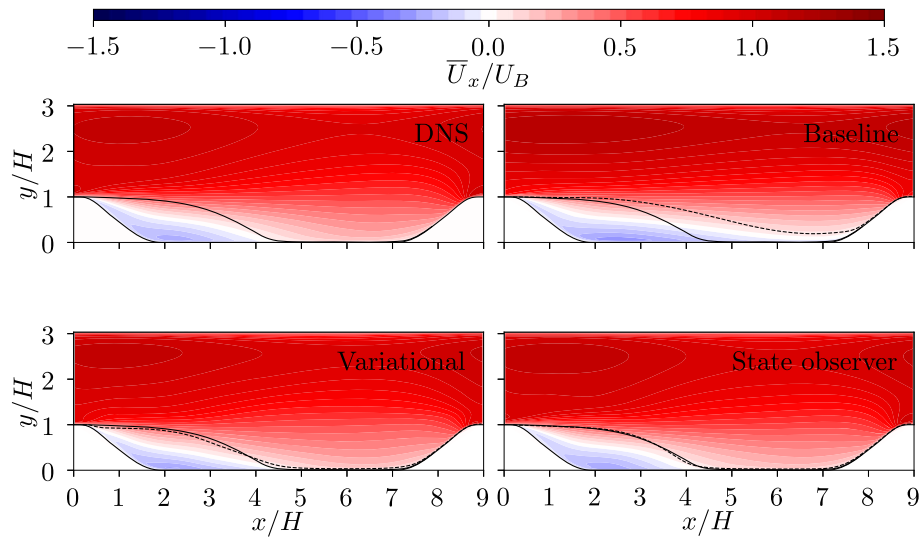


Fig. 4 Comparison of streamwise mean velocity scaled by bulk velocity U_B between DNS, SA baseline, data assimilated variational and data assimilated state observer methods for the highest resolution of computational mesh and input data (21,600 cells and 6542 vectors)

suggests that both approaches are robust to variations in the input data resolution, which can be a critical factor in practical applications. Consequently, Fig. 5 only displays variations in the computational mesh resolution.

The improvement in mean-velocity prediction for the variational method depends on the objective function field. The magnitude of the objective function is higher at the point of separation on the windward hill, along the bottom and top walls being skewed more towards the leeward hill. On the other hand, the state observer method demonstrates superior agreement with experimental data compared to the variational method. Specifically, the state observer method exhibits velocity contours that closely match the experimental values in the freestream region.

The variational method exhibits independence from computational mesh resolution. The variational method falls short at predicting the shear layer and freestream. In contrast, when increasing the mesh resolution, an improvement in the accuracy of the streamwise velocity component for the state observer method is observed. In particular, the recirculation region aligns more closely with the experimental data. However, it must be noted that outside of the recirculation region, the accuracy of the wall-normal velocity component appears to decrease with increasing mesh resolution.

When examining the reattachment point, shown in Table 2, the variational and state observer methods exhibit notable independence from both mesh and experimental resolution. The variational method consistently predicts the reattachment point within a range of $(\pm 0.3)H$. In contrast, the state observer method deviates from the actual reattachment point by $(1 \pm 0.4)H$. This stark difference suggests that the variational method is able to predict the boundary layer physics more accurately.

To clearly understand the discrepancies of the variational method in the freestream and the state observer method at the boundary, the streamwise velocity profiles at nine evenly spaced streamwise locations are investigated. In Fig. 6, we present velocity profiles computed at the finest mesh resolution for both the finest and coarsest experimental resolutions. This analysis aims to explore how changes in experimental data can affect the predicted velocity.

Within the streamwise profiles shown in Fig. 6, similar to the velocity contours, we observe that the variational method presents a notable discrepancy in the recirculation region. This is most noticeable within the shear layer where the velocity is under-predicted and could be a result of an incorrect separation prediction. On the other hand, the state observer method almost perfectly matches with the streamwise velocity profiles of the DNS within the freestream. Both methods produce vast improvements when compared to the baseline prediction within all regions of the fluid domain.

The velocity profile serves as a valuable tool for identifying and amplifying discrepancies in velocity, especially within the near-wall region. The variational method consistently captures boundary layer physics across different experimental resolutions, with very minor improvements as experimental resolution increases. This consistency suggests that the variational method is less influenced by variations in experimental data as

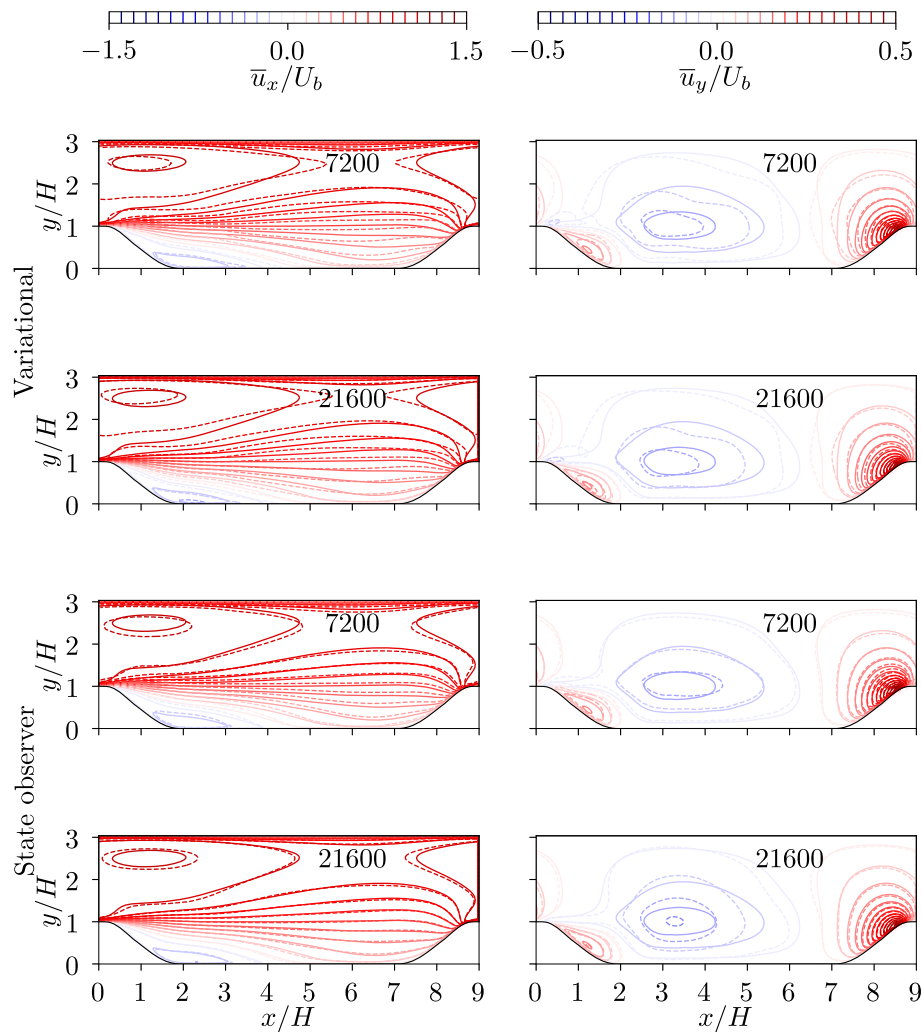


Fig. 5 Contours of streamwise and wall-normal velocity of variational and state observer method (dashed lines) compared with DNS (solid lines) for 7200 and 21,600 computational cells, for the highest resolution of input data (6542 vectors)

a result of a global forcing through the β field. Consequently, the forcing at the wall remains independent of surrounding experimental information.

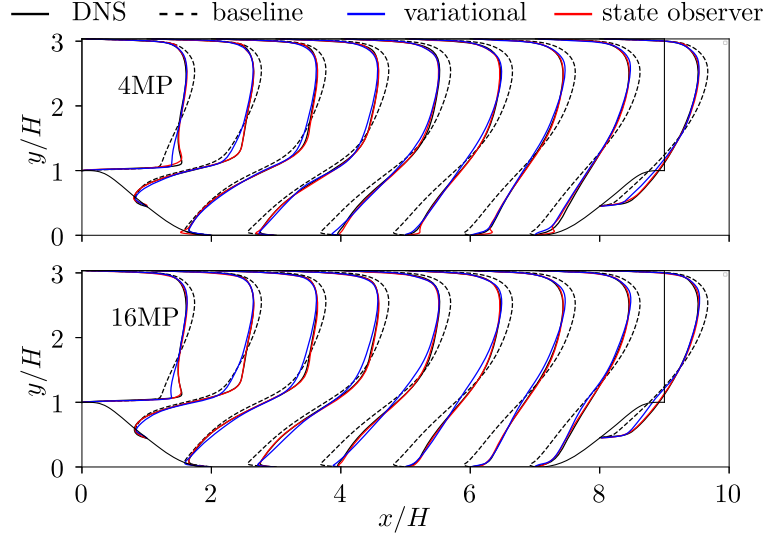
In contrast, the state observer method's limitations near the wall align with the earlier observations with regard to the inaccuracy of predicting the reattachment location, revealing challenges in capturing the boundary layer physics. Contrary to prior assumptions, there is a notable improvement in the state observer method's performance near the wall with increasing experimental resolution. This improvement is attributed to the availability of data closer to the wall.

The state observer method sets extrapolated forcing locations to zero when data is limited, as is the case with a coarse experimental mesh. In such scenarios, a significant number of computational points near the wall receive zero forcing. However, when using a finer experimental mesh that includes more detailed information closer to the wall, the number of computational cells with zero forcing decreases, resulting in improved predictions. As shown in Fig. 2, the coarsest experimental resolution reveals areas without data at the wall of the periodic hill. In these areas, points within the computational mesh are assigned zero forcing.

We present the comparison of the L_1 error of the streamwise and wall-normal velocities scaled by the bulk velocity U_B between the variational and state observer methods for different computational and experimental data resolutions. The L_1 norm, presented in Eq. 11, is selected for comparison due to its resistance to outliers, making it a better indicator of overall error reduction in the domain. The comparisons are made with the DNS data as the reference. All the assimilated fields and the DNS data are interpolated onto the grid of 4MP

Table 2 Reattachment locations for the state observer method, variational method and DNS for variations in computational and experimental resolution

Experimental resolution	Variational [L/h]			State observer [L/h]			DNS [L/h]
	Coarse	Medium	Fine	Coarse	Medium	Fine	
Coarse	5.03	5.10	5.15	4.39	4.35	4.21	5.03
Medium	5.15	5.33	5.23	4.27	4.29	4.39	5.04
Fine	5.01	5.19	5.01	4.37	4.36	4.33	5.04


Fig. 6 Streamwise velocity profile comparison of the lowest and highest input data resolution for a mesh of size 21,600 cells

resolution. This is accomplished using,

$$L_1 = \frac{|\mathcal{Q}(u_x) - \mathcal{Q}(U_x)| + |\mathcal{Q}(u_y) - \mathcal{Q}(U_y)|}{U_B}, \quad (11)$$

where linear interpolation is used as $\mathcal{Q}(\cdot)$ to transfer data from the computational mesh to the experimental grid. This information is visualized in Fig. 7 using a block format for both the variational and state observer methods.

At all levels of experimental and computational resolutions, the state observer method consistently exhibits a more substantial reduction in error between DNS and the final computation when compared to the variational method. Notably, the largest L_1 error value for the state observer method is approximately 30% lower than that of the variational method. This finding aligns with the results reported by Franceschini and Cato [8,9], where they observed that corrections in the beta field led to relatively smaller reductions in the L_1 error compared to corrections in the momentum equation.

There is a decrease in the L_1 error for the variational method with increasing computational mesh resolution for the coarsest experimental data resolution case. This is expected since the variational method has the ability to regularize the input data owing to the global nature of corrections when modifying the forcing term. This inherent regularization allows the effect of computational mesh refinement to manifest a reduction in the error of assimilated quantities. The differences in the error between the computational grids reduce with increasing experimental data resolution. For the 16MP case, the L_1 norms are identical. It can also be observed that for the variational method, the L_1 norms of all the cases lie within 10% of a base value of 0.039 suggesting a less pronounced influence of experimental and computational grid resolution for the assimilated quantities.

On the contrary, the state observer method exhibits an opposing trend: there is a consistent increase in L_1 error as the computational mesh becomes finer, regardless of the experimental data resolution. Finer computational meshes create a greater disparity between the number of input data and mesh points. Dealing with this disparity requires the state observer method to distribute input data among a larger number of computational mesh points. Conversely, reducing this disparity provides a straightforward one-to-one mapping between input

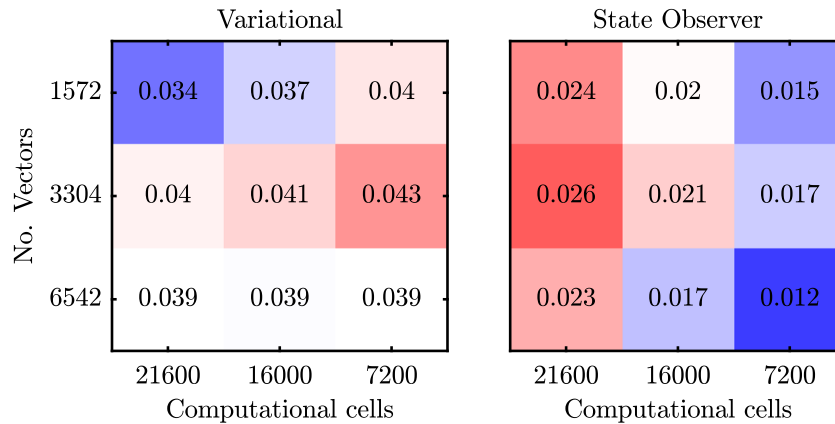


Fig. 7 L_1 error of streamwise and wall-normal velocity scaled by U_B between DNS and (left) variational (right) state observer for each experimental and computational resolution. The colors represent the variation of the error from the mean in each method, such that blue is a reduction and red is an increase in L_1 error (color figure online)

data points and corresponding computational mesh points. This is evident in Fig. 7, where the smallest L_1 error occurs with a 16MP data set (consisting of 6542 points as shown in Table 1) and a computational mesh resolution of 7200. It is clear that having nearly identical numbers of input data and computational mesh points benefits the state observer method.

Since we are using DNS data without the addition of noise or uncertainties, it is theoretically possible to continue improving the L_1 norm at the expense of a large number of primal solver iterations. In a real experiment, there will be sources of uncertainty such as in the recirculation region due to the lack of seeding. Such errors would not permit reducing the discrepancy between the reference and computational fields beyond a certain point. In such cases, a higher value of L_1 norm would be more desirable therefore reducing the computational cost. A description of the computational cost of the two methods is presented with the use of primal solver calls in Appendix C.

5 Reconstructed variables

Upon examination of the velocity, it becomes evident that discrepancies between the two methods, particularly in proximity to the wall, require further analysis. The variational method demonstrates accurate performance on the wall, while the state observer method exhibits shortcomings that warrant further investigation. In this section the skin-friction coefficient and wall pressure gradient shown in Fig. 8 and the curl of the forcing term shown in Fig. 9 are discussed for both the variational and state observer methods. Since both methods utilize the velocity field as a control parameter in different ways, these quantities are labelled “reconstructed” variables that remain indirectly influenced by the DA procedure. The skin-friction coefficient and wall pressure gradient reference data of Krank et al. [46] is used and can be found in <https://mediatum.ub.tum.de/1415670>.

For the variational method, the skin friction coefficient aligns well with the DNS result, with the exception of the separation point and the peak C_f location. In these specific regions, disparities emerge, suggesting that the method faces challenges in accurately predicting skin friction behavior under certain flow conditions. However, an overall improvement over the baseline case is observed when the variational method is used. In contrast, the state observer method consistently overshoots the expected values along the bottom wall, alongside clear discrepancies or “distortions” corresponding to geometric changes within the flow field. Hence it is observed that the state observer method performs just as poorly as the baseline for most of the wall as a result of zero forcing in these locations. These features closely resemble the C_f profile obtained by Brenner et al. [10] in their reconstruction without the use of regularisation.

In Fig. 8 at $x = 8.5H$, the region associated with the peak C_f , the variational method demonstrates an improvement as mesh resolution and experimental resolution increase. For the state observer method, a similar trend is observed, with improvements in predicting the peak C_f corresponding to finer experimental and computational resolutions. The variational method exhibits a peculiar behavior at the separation location where $x = 0.5H$. A sharp dip in the C_f is observed that gets worse with increasing mesh resolution. This dip is observed in literature by Cato et al. [9] and can be explained by a small, yet strong secondary recirculation

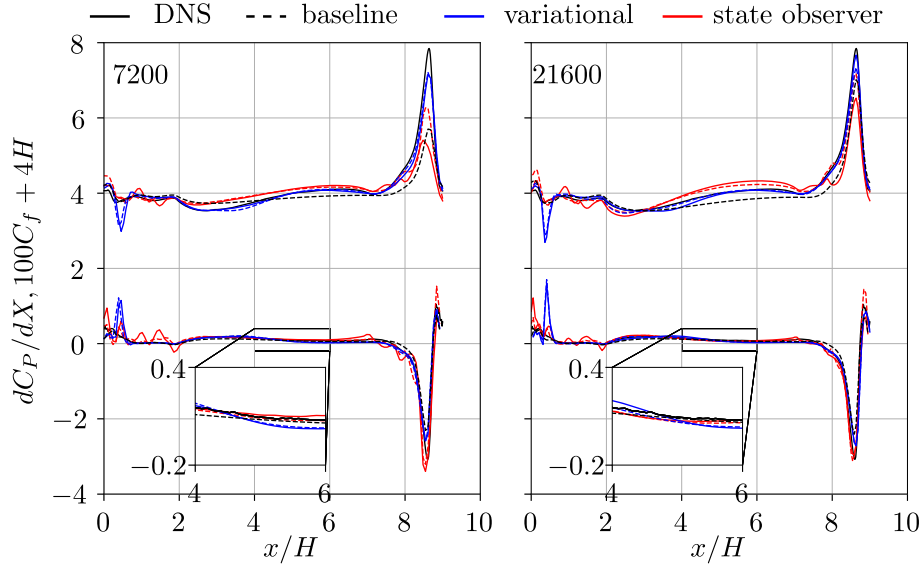


Fig. 8 Skin friction coefficient C_f and pressure gradient dC_P/dX where $X = x/H$ along the bottom wall for an input data resolution of 4MP (solid lines) and 16MP (dashed lines) comparing the variational and state observer methods for 7200 and 21,600 computational cells. Shown in the inset, is a zoomed-in view of the region between $x/H = 4$ and $x/H = 6$. The C_f is scaled and translated

region just downstream of the separation location. The presence of this secondary recirculation region at the separation point is identified as a key contributing factor to the variational method’s discrepancies within shear layer predictions in earlier observations. It is interesting to see a better agreement between DNS reference data (especially in the region $x/H = 4$ to $x/H = 7$) and our C_f prediction obtained using the variational method compared to the one reported in Cato et al. [9].

Notably, the variational method exhibits independence from both mesh and experimental resolutions, aside from deviations in the peak C_f and separation point regions. Conversely, the state observer method displays notable “distortions” in its skin friction coefficient plots when encountering geometry changes, such as transitioning up or down the hill. These “distortions” stem from the influence of experimental resolution on the method’s predictions, which appear to result from forcing effects at the wall. Similar to the explanation of the erroneous velocity profile at the wall, the state observer method enforces zero forcing within regions outside of the experimental domain. Consequently, there are sharp and irregular gradients in the forcing close to the wall hence why “distortions” are observed in the C_f . As expected these “distortions” become less pronounced with greater experimental resolution and in locations where geometric changes are absent.

Similar to the observations regarding C_f , the variational method exhibits an impressive alignment with the DNS results for the pressure gradient, with notable exceptions at the separation point and a slight under-prediction of the gradient at $x = 8.5H$. These deviations suggest that the pressure is influenced by similar challenges as that faced by the velocity in accurately predicting specific flow conditions. Similarly, the state observer method accurately predicts C_f values on the flat region of the hill and correctly predicts the gradient at $x = 8.5H$, though with a notable overprediction at $x = 9H$. When observing the inset of Fig. 8, the state observer method appears to be in better agreement with the results of the DNS. The pressure gradient of the variational method seems to decrease along the wall at a larger rate than that of the DNS and state observer method. Similar to what was observed in the C_f plots, the state observer method shows fluctuations in pressure gradient predictions, mainly in areas with varying geometry.

Aside from the discrepancies at the separation location and peak pressure gradient, Fig. 8 shows the variational method displays remarkable independence from both experimental and mesh resolutions, consistent with earlier observations. In contrast, the state observer method demonstrates an improvement with mesh resolution. Additionally, the presence of “distortions” in the state observer methods prediction of pressure gradient appears to diminish with increasing experimental resolution, echoing the previous discussions. These trends align with the influence of experimental resolution on the state observer method’s predictions, particularly in regions with changing geometry.

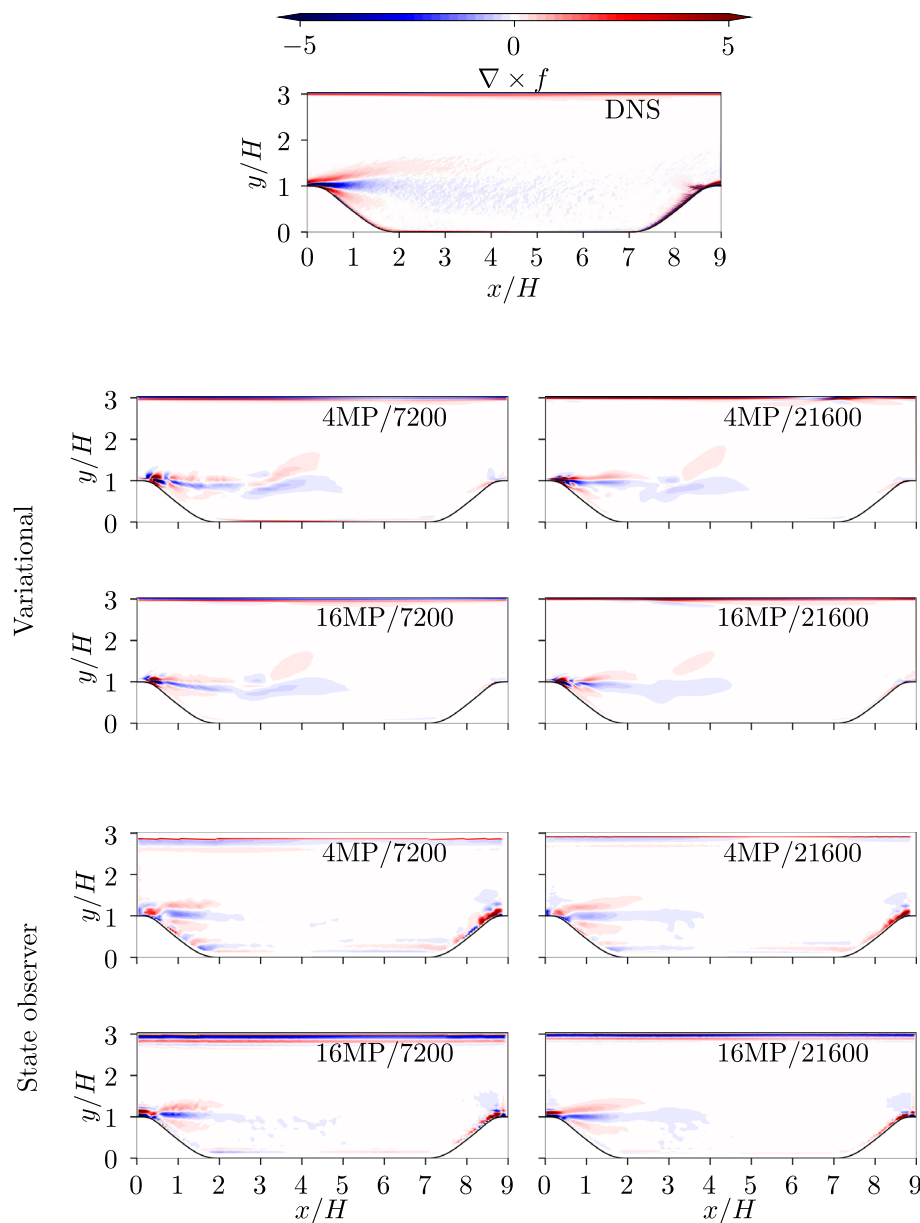


Fig. 9 Contours of $\nabla \times \mathbf{f}$ comparing variational and state observer method with DNS for the lowest and highest computational and input data resolutions (7200 and 21,600 computational cells for 1572 and 6542 input vectors)

To explain why particular methods are producing the results discussed earlier, we focus on the forcing term (which is the divergence of the Reynolds stress tensor), particularly examining the curl of this term, shown in Fig. 9. By taking the curl of the forcing, we remove the contribution of the potential forcing which is absorbed into the pressure and cannot be separated [6]. This allows for a much more accurate comparison with DNS. The magnitude and shape of the curl of the forcing term provide valuable insights into the behavior of both DA methods. It should be noted that the forcing in the case of the variational method is only from the turbulence model. In contrast, the forcing in the case of the state observer method encompasses both the isotropic forcing (obtained from the turbulence model) and the forcing that is added to the momentum equation.

For the variational method, the curl of the forcing term is observed to become smoother as mesh resolution increases. Figure 9 illustrates that the magnitude and shape of the forcing remain consistent across all experimental and mesh resolutions. This consistency aligns with the earlier observations, where the variational method demonstrated independence from both experimental and mesh resolutions within the freestream. In

contrast, for the state observer method, the curl of the forcing term is observed to match more closely with the results of the DNS as mesh resolution increases, extending the influence of the forcing further into the freestream. This observation provides insights into why previous results showed an enhancement in the state observer method’s performance within the freestream region with increasing mesh resolution.

Figure 9, for the lowest input data resolution of the state observer method, shows pockets of substantial forcing close to the wall in regions with geometric changes. These pockets of large forcing appear to reduce as experimental resolution is improved. The presence of pockets of significant forcing offers a crucial link to the previous discussion and signify a localized and pronounced influence on the momentum within the flow. As stated previously, where experimental data is limited, especially for changes in geometry, forcing is set to zero thereby producing large gradients between computational cells. These large gradients are visible by the significant forcing pockets, which gradually reduce with greater experimental resolution. These findings align with the earlier discussions concerning the state observer method’s performance, where improvements in experimental resolution led to reduced “distortions” in skin friction coefficients and pressure gradients, thereby achieving better agreement with the DNS results. Therefore, the presence of these forcing pockets provide a key rationale for the state observer method’s performance and emphasizes the significance of experimental data quality in achieving accurate results.

In contrast, aside from the separation location, the variational method consistently exhibits no forcing on the boundary, regardless of the experimental or computational resolution. Unlike the state observer method, there is no forcing on the leeward hill which might explain the discrepancies in the skin-friction coefficient and pressure gradient peaks observed earlier. This significant difference in forcing behavior at the boundary between the variational and state observer methods plays a crucial role in explaining the robust wall statistics observed in the variational method, which remain consistent across various cases, as opposed to the dependence of the state observer method on experimental resolution. The forcing observed at the separation location for all the variational cases shows small pockets of large forcing. This observation provides an explanation for the discrepancy in the C_f and pressure gradient plots. It is suggested that this discrepancy convects downstream thereby mispredicting the shear layer of the flow, hence why the freestream struggled to improve within this region.

6 Conclusion

We present a new implementation of a variational DA algorithm developed previously [6] that employs a discrete adjoint method with a direct correction to the turbulence transport equation and a state observer method with a correction in the momentum equations independent of the turbulence model. The two methods are applied to a periodic hill test case under varying conditions of computational and experimental mesh resolutions. Our findings reveal that both the variational and state observer methods exhibit distinct strengths and limitations. The variational model correction method, aside from the separation location, is robust and consistent at the wall across various cases, thanks to its minimal boundary-forcing behavior. In contrast, the model-independent state observer method demonstrates a particular sensitivity to experimental resolution in regions with geometric changes, which manifests as poor velocity profiles and localized “distortions” in the skin friction coefficient and pressure gradient.

Furthermore, the study highlights the importance of mesh resolution in shaping the performance of these DA methods. The model-dependant variational method is less accurate in the freestream, especially within the shear layer. It is relatively independent of mesh and experimental resolution, except for specific locations such as the separation point. Within experimental campaigns, ensuring high-resolution velocity data can be expensive and time-consuming. Hence, with limited experimental resolution, the variational method with a correction in the turbulence transport equation will be able to provide improved wall statistics. The state observer method with a correction in the governing equations exhibits improvements in the freestream region with increasing mesh resolution, independent of experimental resolution, and improvements in the near wall region with increasing experimental resolution. Therefore, with limited experimental resolution in the freestream, the state observer method with a correction in the momentum equations will give improved results for those freestream locations.

While we considered input data resolution, it was still synthetically generated to simulate an experimental scenario. The results provide insights into the requirements for an experimental setup that can achieve optimal reconstruction while minimizing costs. As a future expansion of this research, we plan to test our algorithms with real experimental data obtained from PIV. Furthermore, we aim to apply our methods to practical flow scenarios that involve higher Reynolds numbers and more complex flow physics. Utilizing real PIV data and

extending our work to higher Reynolds numbers will undoubtedly present their own set of challenges, which we are eager to explore in a future study.

Acknowledgements The authors acknowledge the use of the IRIDIS High Performance Computing Facility, and associated support services at the University of Southampton, in the completion of this work.

Author contribution CT and UCP both contributed equally in developing DA algorithms, carrying out computations, analysing the data and writing several drafts, BG and SS were responsible for conceptualisation, funding acquisition, editing of drafts and project management.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Funding We gratefully acknowledge funding from EPSRC (Grant Ref: EP/W009935/1) and the School of Engineering at University of Southampton for CT's and UCP's PhD studentships.

Data availability All data supporting this study are openly available from the University of Southampton repository at <https://doi.org/10.5258/SOTON/D3137>.

Code availability All code supporting this study are openly available from the University of Southampton repository at <https://doi.org/10.5258/SOTON/D3137>.

Declarations

Conflict of interest The authors declare no conflict of interest.

Appendix A Projection and smoothing

The details of the projection and smoothing procedure are presented here in a lot more detail. Consider a 3D uniform, structured grid of size N with one cell in the spanwise direction that has the synthetic PIV data stored at the cell centroids. When an unstructured computational mesh (like the ones shown in Fig. 3) of size M is superimposed on it, there are intersections created between the two grids. This can be seen in Fig. 10 where a part of the computational mesh at the hill located close to the inlet is shown superimposed on a structured uniform grid.

Using these intersections, for every synthetic PIV grid cell, we compute the intersection volume \mathcal{V}_{ij} of the i th computational mesh cell associated with the j th synthetic PIV grid cell. Figure 10 has one such grid cell zoomed in displaying the computational mesh cells that intersect it. The state variable (in this case, velocity) stored at the cell centroids of these intersected cells is then weighted by \mathcal{V}_{ij} and scaled by the volume of the synthetic PIV grid cell (which is constant for every cell in the domain),

$$\hat{u}_j = \sum_{i=1}^M \frac{u_i \mathcal{V}_{ij}}{V_{E_j}}, \quad (\text{A1})$$

where \hat{u}_j is the cell-volume weighted averaged velocity at the j th synthetic PIV grid cell and V_{E_j} is the cell volume of the synthetic PIV grid. This velocity is used in the computation of the objective function given by Eq. 4.

Once the objective function is computed, it should be transferred to the computational mesh. Let \hat{f} be the objective function computed on the synthetic PIV grid. We use the operator $\hat{Q}(\hat{f})$ to obtain f that lies on the computational mesh (as denoted by Eq. 4). This involves a smoothing operation of weighting the objective function of a given grid cell with volume fractions and distributing it to those computational cells that intersect with it. The volume fraction is given by

$$\mathcal{W}_{ij} = \frac{\mathcal{V}_{ij}}{V_{C_i}}, \quad (\text{A2})$$

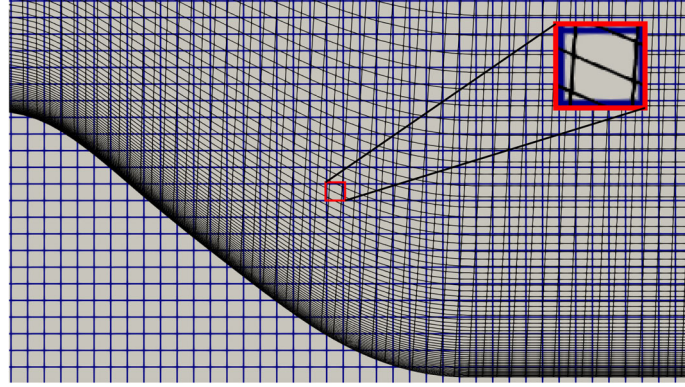


Fig. 10 Overlap of synthetic PIV grid (—) and computational mesh (—). A single grid cell of the synthetic PIV domain is highlighted (—) and zoomed in to show the intersection at a cell level (color figure online)

where \mathcal{W}_{ij} is the volume fraction resulting from the intersection of the i th computational mesh cell with the j th grid cell and V_{C_i} is the volume of the i th computational mesh cell. The smoothed objective function is now given by

$$f_i = \sum_{j=1}^N \mathcal{W}_{ij} \hat{f}_j. \quad (\text{A3})$$

The smooth objective function can now be used to calculate the sensitivity.

The intersections are created at the start of the computation, which is performed only once since this is geometry-dependent (which does not change throughout the course of the optimization process). The calculation of cell-volume weighted averaged velocity and smoothed objective function is done using a newly implemented objective function model. This allows the usage of two separate and independent meshes—one for storing computational data and the other for experimental data.

Appendix B Determining K_p for the SO method

A preliminary investigation into the proportionality constant K_p , as depicted in Fig. 11, compared the state observer method using the coarsest computational mesh and measurement data, utilizing the L_1 norm for evaluation.

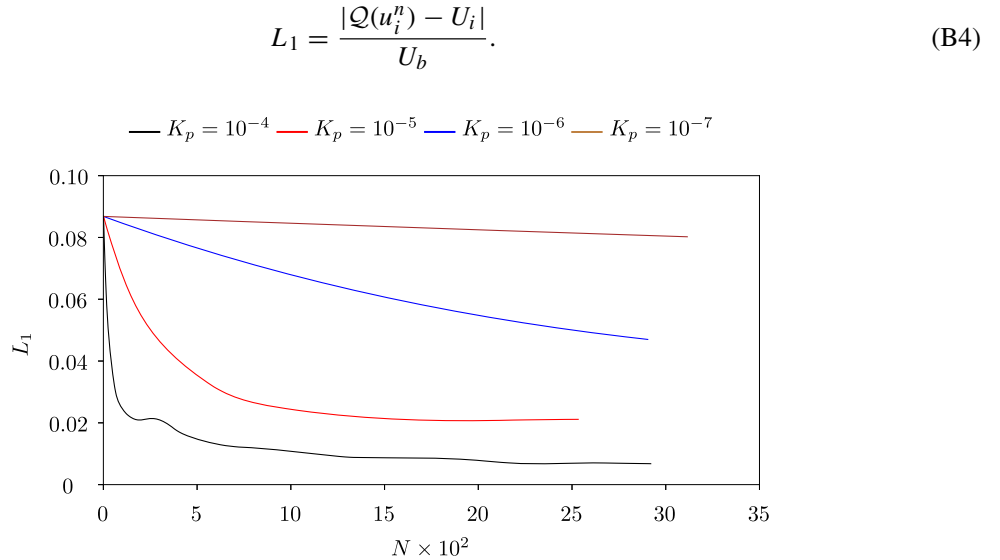


Fig. 11 The L_1 error achieved against number of iterations for varying K_p

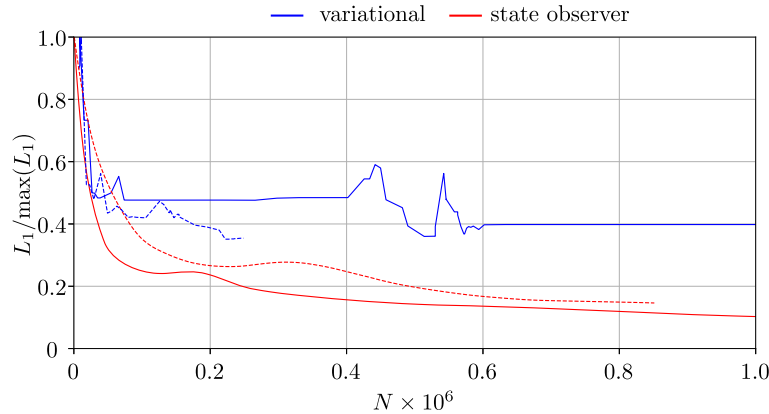


Fig. 12 Variation of the L_1 norm scaled by $\max(L_1)$ with the number of primal solver (OpenFOAM) calls for the variational and state observer methods. Solid lines are for a mesh size of 7200 and the lowest input data resolution (4MP). Dashed lines are for a mesh size of 21,600 and the highest input data resolution (16MP)

The L_1 norm, presented in Eq. B4, is selected for comparison due to its resistance to outliers, making it a better indicator of overall error reduction in the domain. The preliminary investigation revealed that a gain of $K_p = 10^{-4}$ achieves the highest level of accuracy with the fewest iterations of changes in F_i . It must be noted that as K_p increases, the number of SIMPLE iterations necessary for convergence increases between each iterative change in F_i . For example, between $K_p = 10^{-7}$ and $K_p = 10^{-4}$ the number of SIMPLE iterations doubles. For $K_p > 10^{-4}$, the state observer method diverges, aligning with findings in existing literature. Consequently, a value of $K_p = 10^{-4}$ is employed for all computations throughout the investigation.

Appendix C Cost and primal solver calls

The computational cost of the two methods is presented as the number of iterations of the primal solver (OpenFOAM). We choose this metric since it allows a fair comparison between the two DA methods given that they differ in the way corrections are applied. The baseline cases converged in an average of 800 primal solver iterations. Figure 12 shows the dependence of the L_1 norm on the number of primal solver iterations. Here we are interested in exploring the accuracy of reconstruction (measured using the L_1 norm) as a function of the computational cost (measured using the number of primal solver iterations). The results are shown for the two extreme cases—lowest input data resolution (4MP) with smallest mesh size (7200 cells) and highest input data resolution (16MP) with largest mesh size (21,600 cells). The L_1 norm is seen to drop to almost 50% in $O(10^5)$ primal solver iterations. This corresponds to the methods correcting the large-scale features in the flow that contribute the most towards the L_1 norm. Beyond this point, the variational method is unable to make any significant improvements to the L_1 norm which can be attributed to the choice of the model correction that is employed. On the other hand, the state observer method can continue improving the reconstruction, albeit more gradually.

References

1. Pandya, D., Vachharajani, B., Srivastava, R.: A review of data assimilation techniques: applications in engineering and agriculture. *Mater. Today Proc.* **62**, 7048–7052 (2022)
2. Le Dimet, F.-X., Talagrand, O.: Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus A Dyn. Meteorol. Oceanogr.* **38**(2), 97–110 (1986)
3. Gronskis, A., Heitz, D., Mémin, E.: Inflow and initial conditions for direct numerical simulation based on adjoint data assimilation. *J. Comput. Phys.* **242**, 480–497 (2013)
4. Mons, V., Chassaing, J.-C., Sagaut, P.: Optimal sensor placement for variational data assimilation of unsteady flows past a rotationally oscillating cylinder. *J. Fluid Mech.* **823**, 230–277 (2017)
5. Mons, V., Chassaing, J.-C., Gomez, T., Sagaut, P.: Reconstruction of unsteady viscous flows using data assimilation schemes. *J. Comput. Phys.* **316**, 255–280 (2016)
6. Foures, D.P.G., Dovetta, N., Sipp, D., Schmid, P.J.: A data-assimilation method for Reynolds-averaged Navier–Stokes-driven mean flow reconstruction. *J. Fluid Mech.* **759**, 404–431 (2014)

7. Symon, S., Dovetta, N., McKeon, B.J., Sipp, D., Schmid, P.J.: Data assimilation of mean velocity from 2D PIV measurements of flow over an idealized airfoil. *Exp. Fluids* **58**(5), 1–17 (2017)
8. Franceschini, L., Sipp, D., Marquet, O.: Mean-flow data assimilation based on minimal correction of turbulence models: application to turbulent high Reynolds number backward-facing step. *Phys. Rev. Fluids* **5**(9), 094603 (2020)
9. Cato, A.S., Volpiani, P.S., Mons, V., Marquet, O., Sipp, D.: Comparison of different data-assimilation approaches to augment RANS turbulence models. *Comput. Fluids* **266**, 106054 (2023)
10. Brenner, O., Piroozmand, P., Jenny, P.: Efficient assimilation of sparse data into RANS-based turbulent flow simulations using a discrete adjoint method. *J. Comput. Phys.* **471**, 111667 (2022)
11. Patel, Y., Mons, V., Marquet, O., Rigas, G.: Turbulence model augmented physics-informed neural networks for mean-flow reconstruction. *Phys. Rev. Fluids* **9**(3), 034605 (2024)
12. Evensen, G.: The ensemble Kalman filter for combined state and parameter estimation. *IEEE Control Syst. Mag.* **29**(3), 83–104 (2009)
13. Kato, H., Obayashi, S.: Approach for uncertainty of turbulence modeling based on data assimilation technique. *Comput. Fluids* **85**, 2–7 (2013)
14. Kato, H., Yoshizawa, A., Ueno, G., Obayashi, S.: A data assimilation methodology for reconstructing turbulent flows around aircraft. *J. Comput. Phys.* **283**, 559–581 (2015)
15. Labahn, J.W., Wu, H., Harris, S.R., Coriton, B., Frank, J.H., Ihme, M.: Ensemble Kalman filter for assimilating experimental data into large-eddy simulations of turbulent flows. *Flow Turbul. Combust.* **104**(4), 861–893 (2020)
16. Meldi, M., Poux, A.: A reduced order model based on Kalman filtering for sequential data assimilation of turbulent flows. *J. Comput. Phys.* **347**, 207–234 (2017)
17. Kalman, R.E.: A new approach to linear filtering and prediction problems. *J. Basic Eng.* **82**(1), 35–45 (1960)
18. Hayase, T., Hayashi, S.: State estimator of flow as an integrated computational method with the feedback of online experimental measurement. *J. Fluids Eng.* **119**(4), 814–822 (1997)
19. Hayase, T., Nisugi, K., Shirai, A.: Numerical realization for analysis of real flows by integrating computation and measurement. *Int. J. Numer. Methods Fluids* **47**(6–7), 543–559 (2005)
20. Nisugi, K., Hayase, T., Shirai, A.: Fundamental study of hybrid wind tunnel integrating numerical simulation and experiment in analysis of flow field. *JSME Int. J. B* **47**(3), 593–604 (2004)
21. Yamagata, T., Hayase, T., Higuchi, H.: Effect of feedback data rate in PIV measurement-integrated simulation. *J. Fluid Sci. Technol.* **3**(4), 477–487 (2008)
22. Imagawa, K., Hayase, T.: Numerical experiment of measurement-integrated simulation to reproduce turbulent flows with feedback loop to dynamically compensate the solution using real flow information. *Comput. Fluids* **39**(9), 1439–1450 (2010)
23. Zauner, M., Mons, V., Marquet, O., Leclaire, B.: Nudging-based data assimilation of the turbulent flow around a square cylinder. *J. Fluid Mech.* **937**, 38 (2022)
24. Saredi, E., Ramesh, N.T., Sciacchitano, A., Scarano, F.: State observer data assimilation for RANS with time-averaged 3D-PIV data. *Comput. Fluids* **218**, 104827 (2021)
25. Pallas, N.-P., Bouris, D.: Calculation of the pressure field for turbulent flow around a surface-mounted cube using the SIMPLE algorithm and PIV data. *Fluids* **7**(4), 140 (2022)
26. Hayase, T.: Numerical simulation of real-world flows. *Fluid Dyn. Res.* **47**(5), 051201 (2015)
27. Kellaris, K., Pallas, N.P., Bouris, D.: Numerical calculation of the turbulent flow past a surface mounted cube with assimilation of PIV data. *Meas. Sci. Technol.* **35**(1), 015301 (2023)
28. Spalart, P., Allmaras, S.: A one-equation turbulence model for aerodynamic flows. In: 30th Aerospace Sciences Meeting and Exhibit, p. 439 (1992)
29. Kenway, G.K., Mader, C.A., He, P., Martins, J.R.: Effective adjoint approaches for computational fluid dynamics. *Prog. Aerosp. Sci.* **110**, 100542 (2019)
30. Nadarajah, S., Jameson, A.: A comparison of the continuous and discrete adjoint approach to automatic aerodynamic optimization. In: 38th Aerospace Sciences Meeting and Exhibit, p. 667 (2000)
31. Peter, J.E., Dwight, R.P.: Numerical sensitivity analysis for aerodynamic optimization: a survey of approaches. *Comput. Fluids* **39**(3), 373–391 (2010)
32. Giles, M.B., Pierce, N.A.: An introduction to the adjoint approach to design. *Flow Turbul. Combust.* **65**, 393–415 (2000)
33. Brenner, O., Plogmann, J., Piroozmand, P., Jenny, P.: A variational data assimilation approach for sparse velocity reference data in coarse RANS simulations through a corrective forcing term. *Comput. Methods Appl. Mech. Eng.* **427**, 117026 (2024)
34. He, P., Mader, C.A., Martins, J.R., Maki, K.J.: DAfoam: an open-source adjoint framework for multidisciplinary design optimization with OpenFOAM. *AIAA J.* **58**(3), 1304–1319 (2020)
35. He, P., Mader, C.A., Martins, J.R., Maki, K.J.: An aerodynamic design optimization framework using a discrete adjoint approach with OpenFOAM. *Comput. Fluids* **168**, 285–303 (2018)
36. Luenberger, D.G.: Observing the state of a linear system. *IEEE Trans. Mil. Electron.* **8**(2), 74–80 (1964)
37. Singh, A.P., Duraisamy, K., Zhang, Z.J.: Augmentation of turbulence models using field inversion and machine learning. In: 55th AIAA Aerospace Sciences Meeting, p. 0993 (2017)
38. Giannakoglou, K.C., Papadimitriou, D.I.: Adjoint methods for shape optimization. In: Thévenin, D. (ed.) *Optimization and Computational Fluid Dynamics*, pp. 79–108. Springer, Berlin, Heidelberg (2008)
39. Sagebaum, M., Albring, N.R.G.T.: High-performance derivative computations using CoDiPack. *ACM Trans. Math. Softw. (TOMS)* **45**(4), 1–26 (2019)
40. Yamashita, H.: A globally convergent primal-dual interior point method for constrained optimization. *Optim. Methods Softw.* **10**(2), 443–469 (1998)
41. Wächter, A., Biegler, L.T.: On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.* **106**, 25–57 (2006)
42. Alfeld, P.: A trivariate Clough–Tocher scheme for tetrahedral data. *Comput. Aided Geom. Des.* **1**(2), 169–181 (1984)
43. Xiao, H., Wu, J.-L., Laizet, S., Duan, L.: Flows over periodic hills of parameterized geometries: a dataset for data-driven turbulence modeling from direct simulations. *Comput. Fluids* **200**, 104431 (2020)

44. Weller, H.G., Tabor, G., Jasak, H., Fureby, C.: A tensorial approach to computational continuum mechanics using object-oriented techniques. *Comput. Phys.* **12**(6), 620–631 (1998)
45. Patankar, S., Spalding, D.: A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat Mass Transf.* **15**(10), 1787–1806 (1972)
46. Krank, B., Kronbichler, M., Wall, W.A.: Direct numerical simulation of flow over periodic hills up to $Re_h = 10$, 595. *Flow Turbul. Combust.* **101**, 521–551 (2018)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.