

Towards adjoint-based mesh refinement for Large Eddy Simulation using reduced-order primal solutions

Preliminary 1D Burgers study

Li, Xiaodong; Hulshoff, Steven; Hickel, Stefan

DOI

[10.1016/j.cma.2021.113733](https://doi.org/10.1016/j.cma.2021.113733)

Publication date

2021

Document Version

Final published version

Published in

Computer Methods in Applied Mechanics and Engineering

Citation (APA)

Li, X., Hulshoff, S., & Hickel, S. (2021). Towards adjoint-based mesh refinement for Large Eddy Simulation using reduced-order primal solutions: Preliminary 1D Burgers study. *Computer Methods in Applied Mechanics and Engineering*, 379, Article 113733. <https://doi.org/10.1016/j.cma.2021.113733>

Important note

To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.



Towards adjoint-based mesh refinement for Large Eddy Simulation using reduced-order primal solutions: Preliminary 1D Burgers study

Xiaodong Li*, Steven Hulshoff, Stefan Hickel

Aerodynamics group, Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 2, 2629 HS Delft, The Netherlands

Received 27 August 2020; received in revised form 9 February 2021; accepted 10 February 2021

Available online xxxx

Abstract

Adaptive Mesh Refinement (AMR) is potentially an effective way to automatically generate computational meshes for high-fidelity simulations such as Large Eddy Simulation (LES). When combined with adjoint methods, which are able to localize error contributions, AMR can generate meshes that are optimal for computing a physical quantity of interest (e.g. lift or drag). In order to apply adjoint-based AMR techniques to LES, primal flow solutions are needed to solve the adjoint problem backward in time. However, the resources required to store primal flow solutions can be huge, even prohibitive, in practical problems because of the typically very fine meshes and long averaging times for computing the statistical quantities of interest. Here, a Reduced-Order Representation (ROR) based upon proper orthogonal decomposition is introduced to address this issue. We develop an Enhanced Online Algorithm (EOA) based on incremental singular value decomposition to build this ROR online, which makes adjoint-based AMR feasible for practical applications. An adjoint-based error estimation procedure is first introduced, and verified using a manufactured solution. Then a ROR-driven AMR strategy is studied using a 1D unsteady Burgers problem with a multi-frequency forcing term. This is also used to evaluate the EOA for ROR-driven AMR. Numerical results demonstrate that the enhanced online algorithm generates RORs that are sufficiently accurate for AMR, avoiding the storage of almost all of the primal solution data.

© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Keywords: Large Eddy Simulation; Adaptive Mesh Refinement; Adjoint method; A posteriori error estimation; Order reduction; Incremental singular value decomposition

1. Introduction

Large Eddy Simulation (LES), in which one resolves large-scale turbulent structures while modeling the impact from smaller turbulent scales, has the potential to deliver reliable flow predictions for many applications, such as those involving flow separations from a smooth surface or the analysis of acoustic sources [1]. LES has been shown to be highly successful in situations where the computational mesh is fine enough to resolve an appropriate range of large scales [2]. Nonetheless, defining computational meshes that provide accurate solutions with tractable computing costs is a challenging task, particularly for practical applications with complex geometries and unknown physical features.

* Corresponding author.

E-mail addresses: X.Li-12@tudelft.nl (X. Li), S.J.Hulshoff@tudelft.nl (S. Hulshoff), S.Hickel@tudelft.nl (S. Hickel).

In practice, the construction of meshes for LES usually involves trial and error, even for engineering experts. This is because it is difficult to anticipate the effects of complex flow features, such as laminar turbulent transition, boundary layer separation or vortex interactions, on a desired Quantity of Interest (QoI). It is thus natural to consider Adaptive Mesh Refinement (AMR) techniques [3] for automatically constructing the computational mesh. Traditionally, feature-based adaptation has been used to reduce discretization errors [4,5]. For instance, Roy [6] examined the use of four features: solution gradients, solution curvature, discretization error and truncation error, and found that truncation-error based adaptation provides superior results for the 1D steady Burgers equation. Other physical-based criteria or heuristic indicators have been used to drive mesh adaptation, such as wall distance, vorticity, shock sensors, and small-scale kinetic energy [7]. However, feature-based adaptive schemes can fail to self-terminate or can produce erroneous values for the predicted functional [8]. More importantly, the final grids are usually sub-optimal with respect to the computational cost because these a priori methods treat all local flow features equally, yet not all local features are necessarily related to the QoI. For instance, Park [9] reported that feature-based AMR might not trigger trailing edge adaptation for the ONERA M6 at $Ma = 0.84$ with $AOA = 3$ degrees, where the grid has a significant influence on the drag computation. Consequently, adjoint methods have been developed to determine local contributions to the error in a chosen QoI, so that the adapted mesh can provide the highest accuracy per degree of freedom.

Adjoint-based mesh adaptation has been successfully used for steady problems (e.g. RANS) under different frameworks, including Finite-Element Methods (FEM) [10–12], Finite-Volume Methods (FVM) [8,9,13] and discontinuous Galerkin methods [14,15]. Several review papers presented applications of adjoint-based AMR to both steady linear and non-linear flows [11,16] as well as laminar and turbulent aerodynamic problems [17]. Nemeć et al. [18] applied adjoint-based AMR with cut-cell FVM on Cartesian grids to 3D inviscid compressible flows in sonic boom applications and launch vehicle configurations. Dwight [13] related the dissipation added in FVM to the QoI's accuracy via adjoint methods. Shi and Wang [19] verified the effectiveness of an adjoint-based h-adaptation approach in a high-order method. Adjoint-based AMR approaches were also extended to anisotropic mesh adaptation in flow problems [20,21], relying on anisotropic ratios and quotients from the optimal goal-oriented metric. Adjoint-based AMR has been explored with the variational multiscale method in steady test cases, such as linear advection–diffusion problems [22,23] and aerodynamic RANS cases [24], and was shown to provide reliable local indicators for various QoIs. In summary, the accuracy and efficiency of adjoint-based AMR have been demonstrated in a wide range of cases.

However, the application of adjoint-based AMR to LES poses several challenges. The most significant of these are related to the stability of the adjoint problem, and the high computing costs for its solutions. The stability of the adjoint problem has received considerable attention. Using Direct Numerical Simulation (DNS), Wang and Gao [25] observed divergence when computing drag-based adjoint solutions for the turbulent flow around a cylinder at low Reynolds. Numerous remedies have been proposed, including least-squares shadowing [26], ensemble sensitivity [27], space-split sensitivity [28], and additional artificial viscosity [29]. Conversely, Hoffman et al. [12,30] have observed stable adjoint solutions at high Reynolds numbers computed with wall-modeled boundary conditions. Thus for practical applications the stability of the adjoint remains a vital topic, and is likely to be dependent on the problem QoI and discretization used.

The current work is focused on the second challenge mentioned above, the high computing costs of adjoint solutions. In unsteady simulations, the primal flow problem is advanced forward in time while the adjoint problem is solved backward in time. Due to the non-linearity of Navier–Stokes equations, flow states from the primal LES are required for solving the adjoint problem. For short simulations, one can store complete primal solutions in memory and then use this data to solve the adjoint problem. In practice, however, the QoIs from LES normally are computed over long averaging times, leading to prohibitive memory requirements. In order to handle this difficulty, some researchers choose to store the primal solution on hard disks [31–34]. However, disk space is normally limited and the speed of data communication between hard disk and solver is much slower than access to main memory. Others choose to store flow solution snapshots at a prescribed frequency and interpolate intermediate values in time [30,35]. This has also been employed for high-Re number problems [36]. A disadvantage of this method, however, is the loss of physical features when the frequency is too low. Another solution is to apply the checkpointing technique [37], where primal flow solutions are stored at optimal selected states and then used to re-solve the primal problem locally. Such techniques have been employed to solve unsteady adjoint problems in sensitivity analyses of cylinder flows [25] and tonal noise [38]. However, this requires significant computational effort.

Chaudhry et al. [39] used reduced-order models obtained with Proper Orthogonal Decomposition (POD) to lower the cost of repeated adjoint-based error estimations for steady nonlinear convection–diffusion–reaction problems. In the current work, we propose the use of Reduced-Order Representations (RORs) to efficiently represent the primal solution when computing unsteady adjoint problems. We evaluate the performance of this approach using the 1D Burgers equation, which is often used in the development of numerical schemes for turbulent flows due to its convective nonlinearity and forward energy cascade. We place the problem in the context of AMR for LES by considering a time-averaged QoI and by employing coarse meshes, with which the influence of the subgrid-scale model is significant. Furthermore, to address the difficulty of constructing a POD-like ROR from high-dimensional datasets, we propose an enhanced online algorithm (EOA) for the incremental Singular Value Decomposition (SVD) [40] to improve its efficiency when determining RORs.

This paper is organized as follows. A paradigm for mesh adaptation in LES is described in Section 2, in which an LES model and discretization scheme, adjoint method, a posteriori error estimation, and mesh adaptation strategy are formulated. Section 3 describes the use of a standard ROR and an EOA ROR for adjoint-based AMR. In Section 4, after the validation of error estimation on a Burgers problem with a manufactured solution, we present numerical experiments with the proposed AMR approach on an unsteady non-linear Burgers problem with a multi-frequency forcing term. Results obtained using full-order solutions, a standard ROR and an EOA ROR are compared. Concluding remarks appear in Section 5.

2. A paradigm for adjoint-based mesh adaption

2.1. Problem formulation and discretization

We consider the one-dimensional (1D) Burgers equation over a space–time domain $\Omega : [0, 1] \times I : [0, T]$. The Burgers equation is often used as a mathematical model for applications that involve shock wave propagation in viscous flows or idealized turbulence [41]. The Burgers equation is expressed as

$$\mathcal{N}(u) = \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} - \nu \frac{\partial^2 u}{\partial x^2} = f, \quad (1)$$

where $\mathcal{N}(\cdot)$ is a non-linear operator and u is the solution with boundary conditions $u(0, t) = u(1, t) = 0$ and an initial condition $u(x, 0) = u_0$. ν is the viscosity coefficient and $f \in \mathbb{R}$ is a known forcing term. Note that we use a 1D problem here to explore the methodology; however, the approaches considered can be directly extended to multi-dimensional problems.

2.1.1. Variational multiscale method

We employ the finite-element method to solve the primal problem with Dirichlet boundary conditions using the weak form

$$\mathcal{R}(u, \omega) = (u_t, \omega) - (uu_x/2, \omega_x) + (\nu u_x, \omega_x) - (f, \omega) = 0, \quad \forall \omega \in \mathcal{V}, \quad (2)$$

where $\omega \in \mathcal{V}$ are weighting functions and $\mathcal{V} = \mathcal{V}(\Omega)$ denotes both the solution space and weighting space. $\mathcal{R}(\cdot, \cdot)$ denotes the weak form of the residual operator and (\cdot, \cdot) is the L_2 inner product. The inner product used in this paper is defined by the spatial domain Ω by default.

The Variational Multiscale Method (VMM) [42,43] is then used to derive a form suitable for LES. In VMM, the flow solution is split into two components, the resolved scales \bar{u} and the unresolved scales u' . The unresolved-scale equations are driven by the strong residual, i.e. $R(\bar{u}) = \mathcal{N}(\bar{u}) - f$. The simplest algebraic model for u' uses the quasi-static subscales assumption and a volume-averaged Green's function to write $u' \approx -\tau R(\bar{u})$. For the current problem, we use this approximation along with an expression for τ from Wang and Oberai [44], viz. $\tau = [\frac{4}{h^2}u^2 + 3\pi\nu^2(\frac{4}{h^2})^2]^{-1/2}$. Substituting $u = \bar{u} + u'$ into Eq. (2), the weak form is then:

$$\mathcal{A}(\bar{u}, \omega) = (\bar{u}_t, \omega) - (\frac{1}{2}\bar{u}\bar{u}_x, \omega_x) + (\nu\bar{u}_x, \omega_x) - (f, \omega) - (\bar{u}u', \omega_x) - (\frac{1}{2}u'^2, \omega_x) - (\nu u', \omega_{xx}) = 0, \quad (3)$$

where it has been assumed $u' = 0$ on the boundary. By virtue of $u' = -\tau R(\bar{u})$, \bar{u} is exclusively determined from Eq. (3). In analogy with LES, the last three terms of Eq. (3) correspond to a model for effects of the subgrid scales.

2.1.2. Discretization

When numerically solving the flow problem, we replace \bar{u} by \bar{u}_h , leading to a discrete system

$$\mathcal{A}_h(\bar{u}_h, \omega_h) = 0, \forall \omega_h \in \mathcal{V}_h, \tag{4}$$

where the subscript h denotes element size within a computational mesh with N_v degrees of freedom. $\bar{u}_h \in \mathcal{V}_h$ is abbreviated to u_h in the subsequent text where there is no conflict. The semi-discrete technique is used to discretize this unsteady model problem. Specifically, we use piecewise linear basis functions for spatial discretization and a four-stage Runge–Kutta scheme to advance the primal problem in time from $t = 0$ to $t = T$. Note that the $(\nu u', \omega_{xx})$ term in Eq. (3) is zero in this case due to the use of piecewise linear functions.

2.2. Adjoint method

The adjoint method is used to provide estimates of local contributions to the error in a chosen QoI, allowing for the construction of goal-oriented adapted meshes. In unsteady simulations, a QoI is often a statistical function of the primal solution. In this paper, we consider a volume-integrated statistical function $\bar{J}(u) = \frac{1}{T} \int_0^T J(u) dt = \frac{1}{T} \int_0^T (g, u)_\Omega dt$, where g is a real function from $\mathbb{R} \rightarrow \mathbb{R}$ and $\bar{J}(u) \in \mathbb{R}$. Note that although only results for linear QoIs will be discussed in later sections, the adjoint equation and error expressions given below are also valid for non-linear QoIs.

By virtue of a Lagrange function, the adjoint equation derived for the current problem is

$$\mathcal{L}_u^* v = -\frac{\partial v}{\partial t} - u \frac{\partial v}{\partial x} - v \frac{\partial^2 v}{\partial x^2} = g_u, \tag{5}$$

where v is the Lagrange multiplier or adjoint variable with homogeneous boundary conditions $v(0, t) = v(1, t) = 0$ and an initial condition $v(x, T) = 0$. $\mathcal{L}_u^*(\cdot)$ is a linearized adjoint operator which relies on the primal solution u . g_u is a Fréchet derivative of $J(u)$, defined as

$$(g_u, \tilde{u}) := \lim_{\epsilon \rightarrow 0} \frac{J(u + \epsilon \tilde{u}) - J(u)}{\epsilon}, \quad \forall u, \tilde{u} \in \mathcal{V}. \tag{6}$$

The adjoint residual operator is expressed as $R_{[u]}^*(\cdot) = \mathcal{L}_u^*(\cdot) - g_u$ with respect to a given u . We solve the adjoint problem using the same VMM employed for the primal problem.

2.3. A posteriori error estimation framework

Before developing an a posteriori error estimation, we consider two fundamental properties, the *adjoint identity* and the *averaging Fréchet operator*. The adjoint identity, which can be easily verified by partial integration, is

$$(\mathcal{L}_u^* v, \tilde{u})_{\Omega \times I} = (v, \mathcal{L}_u \tilde{u})_{\Omega \times I} + (v, \tilde{u})_{\Omega} |_{t=t_0}, \tag{7}$$

for $\forall v, u, \tilde{u} \in \mathcal{V}$. The last term originates from the contribution of non-zero values at statistical-starting time t_0 . \mathcal{L}_u is the Fréchet derivative of $\mathcal{N}(u)$,

$$\mathcal{L}_u \tilde{u} := \lim_{\epsilon \rightarrow 0} \frac{\mathcal{N}(u + \epsilon \tilde{u}) - \mathcal{N}(u)}{\epsilon}, \quad \forall u, \tilde{u} \in \mathcal{V}. \tag{8}$$

We can formulate the averaging Fréchet operator, $\bar{\mathcal{L}}_{(u_1, u_2)}^*(\cdot)$, by integrating Equation (5) of u from u_1 to u_2 , which enables us to estimate the error for non-linear problems. By defining $u = u_1 + \theta(u_2 - u_1)$, the adjoint equation is integrated as

$$\bar{\mathcal{L}}_{(u_1, u_2)}^* v \equiv \int_0^1 \mathcal{L}_{u=u_1+\theta(u_2-u_1)}^* v d\theta \stackrel{\text{Eq. (5)}}{=} \int_0^1 g_{u=u_1+\theta(u_2-u_1)} d\theta \equiv \bar{g}_{(u_1, u_2)}, \tag{9}$$

where the subscripts, u_1 and u_2 , denote the integration domain and $\bar{g}_{(u_1, u_2)}$ represents an averaged value of g_u on this domain. Considering the definition in Eq. (6), we can use this $\bar{g}_{(u_1, u_2)}$ to express the difference of a QoI as

$$J(u_1) - J(u_2) = (\bar{g}_{(u_1, u_2)}, u_1 - u_2). \tag{10}$$

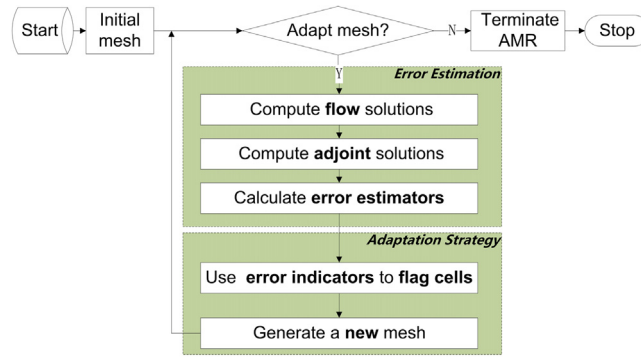


Fig. 1. A general procedure of mesh adaptation based on adjoint method for unsteady simulations.

For the unsteady Burgers problem, we have $\bar{\mathcal{L}}_{(u_1, u_2)}^* v = \mathcal{L}_{\frac{u_1+u_2}{2}}^* v$. Likewise, the average linearized operator in Eq. (9) maintains the adjoint identity as follows

$$(\bar{\mathcal{L}}_{(u_1, u_2)}^* v, u)_{\Omega \times I} = (v, \bar{\mathcal{L}}_{(u_1, u_2)} u)_{\Omega \times I} + (v, u)_{\Omega} |_{t=t_0}, \tag{11}$$

where $v, u, u_1, u_2 \in \mathcal{V}$. $\bar{\mathcal{L}}_{(u_1, u_2)}(\cdot)$ is the averaging operator of $\mathcal{L}_u(\cdot)$ by integrating u from u_1 to u_2 , which satisfies

$$\bar{\mathcal{L}}_{(u, u_h)}(u_h - u) = \mathcal{N}(u_h) - \mathcal{N}(u) = \mathcal{R}(u_h). \tag{12}$$

Substituting $u_1 = u$ and $u_2 = u_h$ into Eqs. (9) to (11), we introduce an expression for estimating the error ϵ of the QoI

$$\begin{aligned} \epsilon &= \bar{J}(u) - \bar{J}(u_h) \\ &= \underbrace{-\frac{1}{T}(v_h, R(u_h))_{\Omega \times I} - \frac{1}{T}(v_h, u_h - u)_{\Omega} |_{t=t_0}}_{\text{Adjoint correction}} + \underbrace{\frac{1}{T}(R_{[\frac{u+u_h}{2}]}^*(v_h), u_h - u)_{\Omega \times I}}_{\text{Remaining error}} = \sum_{e=1}^N \epsilon_e, \end{aligned} \tag{13}$$

where N denotes the number of elements at current AMR level. The error estimation is divided into an adjoint correction and remaining error. There is a temporal contribution to the adjoint correction due to the non-zero value of u at the starting time of the statistical time period, which does not appear in steady simulations. In practice, $R_{[\frac{u+u_h}{2}]}^*(v_h)$ is replaced by $R_{[u_h]}^*(v_h)$, where u_h is the discrete primal solution. Here, the VMM is used to approximate the exact solution u as $u_h + u'_h$. The resulting error estimation can be constrained to each element as shown in the last equal sign of Eq. (13), and expressed in terms of elemental error estimators ϵ_e .

2.4. Mesh adaptation strategy

In this paper, we start from a very coarse uniform mesh and only use refinement. A local error indicator, η_e , is determined from the elemental error estimator as $\eta_e = |\epsilon_e|$ to facilitate mesh adaptation. Different adaptation strategies [17] can be developed based on η_e . We employ a prescribed percentage (10%) of mesh cells with the largest errors as the criterion for mesh refinement [19,30]. The resulting procedure of adjoint-based AMR is presented in Fig. 1. In each mesh refinement loop, the error estimation and adaptation strategy are executed to generate a new computational mesh for the next AMR level.

Without additional treatment, the smoothness of the mesh will deteriorate during AMR. Thus, a balancing step is introduced to improve mesh smoothness, as shown in Algorithm 1. The basic principle is that a cell will be flagged for refinement if the ratio between the size of this cell and the size of its neighbors would become larger than 2. This balancing step is recursively executed until there are no more elements that need to be refined.

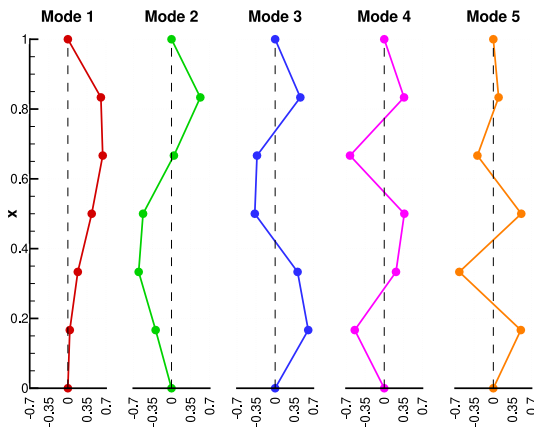
3. POD-based reduced-order representation for AMR

In order to make adjoint-based AMR affordable for LES, we represent the primal flow problem in a low-order space that can be accessed efficiently. For unsteady flow problems, Reduced-Order Models (ROM) based

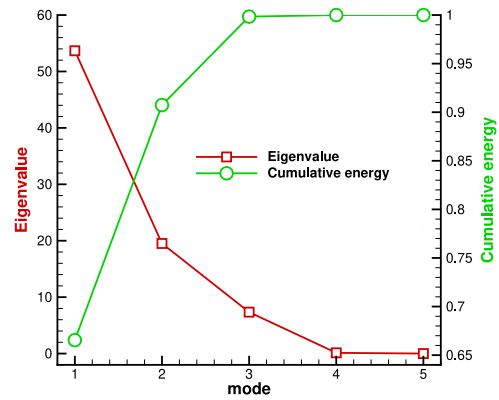
Algorithm 1 Balancing procedure for AMR

```

refined = 1
while refined > 0 do
    refined = 0, temp = cellFlag
    for i ≤ N do
        if cellFlag(i) = 1 then
            if i ≠ 1 and cellFlag(i - 1)=0 then
                if  $\frac{Length(e_i)}{Length(e_{i-1})} < 1$  then
                    temp(i - 1) = 1, refined += 1
            if i ≠ N and cellFlag(i + 1)=0 then
                if  $\frac{Length(e_i)}{Length(e_{i+1})} < 1$  then
                    temp(i + 1) = 1, refined += 1
    cellFlag = temp
    
```



(a) Distribution of POD modes



(b) Eigenvalues and cumulative energy

Fig. 2. POD analysis of a Burgers problem solved on a stationary mesh with 5 primal flow variables.

on projection have been widely used. In this case one projects representative modes (typically a truncated set of POD modes) onto the governing equations to obtain a low-order system. For the current application, however, there is no need to predict the primal solution beyond the original dataset. Thus representative modes and their coefficients can be used directly as a Reduced-Order Representation (ROR) of the primal solution in the adjoint problem. In the following, we initially construct the ROR by applying the standard SVD implemented in LAPACK [45] to stored primal solution data, which results in exact POD modes. We refer to this method as the offline SVD. The procedure then used for ROR-driven AMR is described in Section 3.1. For realistic applications, however, very large datasets would need to be considered, making the cost of an offline SVD prohibitive. Therefore, we also introduce an enhanced online algorithm to build the ROR based on an incremental SVD [40,46], described in Section 3.2. This produces approximations for the POD modes and their amplitudes.

3.1. Offline ROR

The standard approach constructs a ROR offline by gathering complete snapshots into a solution matrix and then applying the SVD analysis, which gives the POD modes (ϕ) and coefficients (α) at once. Fig. 2 shows the distribution of POD modes, eigenvalues and cumulative energy from the solution of a typical Burgers problem, which is also the starting situation for AMR. In this case, the first mode represents a significant part of the instantaneous solution as it accounts for 66.5% of the total energy.

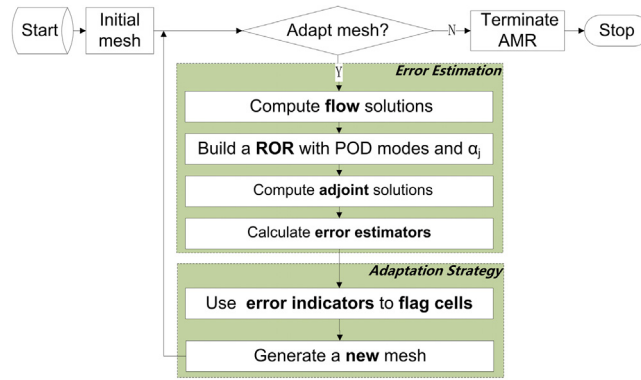


Fig. 3. A general procedure of mesh adaptation based on adjoint method and ROR for unsteady simulations.

The ROR solution is then expressed as

$$u(x, t_i) = \langle u(x) \rangle + \sum_{j=1}^M \alpha_j(t_i) \phi^{(j)}(x), \tag{14}$$

where $\phi^{(j)}$, $j = 1, 2, \dots, M$, denote a low number of selected POD modes, and $\langle u \rangle$ represents the mean value. As shown in Fig. 3, the AMR procedure is modified by replacing the primal solution in the adjoint problem by a ROR determined using the standard SVD. We refer to this as AMR using an offline ROR. Note that the adjoint problem is still solved in a full-order space. The offline ROR is utilized as a benchmark for the online RORs introduced in the next section.

3.2. Enhanced online algorithm for RORs

The modal decomposition of large flow datasets can lead to intractable storage requirements [46]. To overcome this, we introduce an enhanced algorithm based on the incremental SVD [40,46]. We first consider the core updating step of the incremental SVD in Proposition 1 before explaining the enhanced online algorithm.

Proposition 1. Suppose we have a dense matrix \mathbf{U} with a known SVD expressed as $\mathbf{U} = \mathbf{V}\mathbf{\Sigma}\mathbf{W}^T$, where $\mathbf{\Sigma} \in \mathbb{R}^{k \times k}$, $\mathbf{V} \in \mathbb{R}^{n \times k}$ and $\mathbf{W} \in \mathbb{R}^{k \times k}$ are a singular value matrix, left and right singular vector matrix, respectively. When a new column $c \in \mathbb{R}^{n \times 1}$ is added to formulate a new updated $\mathbf{U}_u = [\mathbf{U} \ c]$, a bordered-diagonal sparse matrix \mathbf{Q} is formulated as

$$\mathbf{Q} = \begin{bmatrix} \mathbf{\Sigma} & d \\ 0 & p \end{bmatrix}, \tag{15}$$

where $d = \mathbf{V}^T c$, $h = c - \mathbf{V}d$, $p = \sqrt{h^T h}$. After applying SVD on \mathbf{Q} as $\mathbf{Q} = \mathbf{V}_Q \mathbf{\Sigma}_Q \mathbf{W}_Q^T$, we can have a SVD of the new updated matrix \mathbf{U}_u as

$$\mathbf{U}_u = \mathbf{V}_u \mathbf{\Sigma}_Q \mathbf{W}_u^T, \tag{16}$$

where $\mathbf{\Sigma}_Q$, \mathbf{V}_u and \mathbf{W}_u are the new singular value matrix, left and right singular vector matrix, respectively, given as

$$\mathbf{V}_u = [\mathbf{V} \ j] \mathbf{V}_Q, \mathbf{W}_u = \begin{bmatrix} \mathbf{W} & 0 \\ 0 & 1 \end{bmatrix} \mathbf{W}_Q, \tag{17}$$

where $j = h/p$.

Proof. Considering a new column c , we can project it onto a space spanned by \mathbf{V} as $d = \mathbf{V}^T c$. Then the orthogonal component of c is given as $h = c - \mathbf{V}d = pj$, where $p = \sqrt{h^T h}$ and $j = h/p$. c can be re-expressed as a sum of

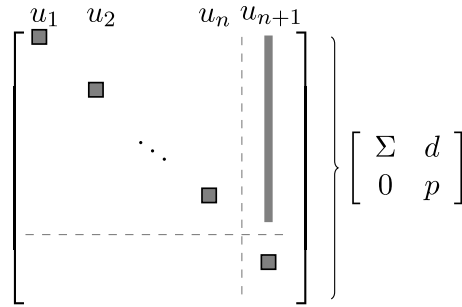


Fig. 4. The bordered-diagonal matrix \mathbf{Q} .

the orthogonal and projected components as $c = pj + \mathbf{V}d$. This leads to

$$\mathbf{U}_u = [\mathbf{U} c] = [\mathbf{V}\Sigma\mathbf{W}^T pj + \mathbf{V}d] = [\mathbf{V} j] \begin{bmatrix} \Sigma & d \\ 0 & p \end{bmatrix} \begin{bmatrix} \mathbf{W} & 0 \\ 0 & 1 \end{bmatrix}^T = [\mathbf{V} j] \mathbf{Q} \begin{bmatrix} \mathbf{W} & 0 \\ 0 & 1 \end{bmatrix}^T. \tag{18}$$

Apparently, \mathbf{Q} is a bordered-diagonal sparse matrix with only the last column fully filled as shown in Fig. 4. We can easily obtain \mathbf{Q} 's SVD as $\mathbf{Q} = \mathbf{V}_Q \Sigma_Q \mathbf{W}_Q^T$, resulting in

$$\mathbf{U}_u = [\mathbf{V} j] \mathbf{V}_Q \Sigma_Q \left(\begin{bmatrix} \mathbf{W} & 0 \\ 0 & 1 \end{bmatrix} \mathbf{W}_Q \right)^T = \mathbf{V}'_u \Sigma_Q \mathbf{W}'_u{}^T, \tag{19}$$

where $\mathbf{V}'_u = [\mathbf{V} j] \mathbf{V}_Q$, $\mathbf{W}'_u = \begin{bmatrix} \mathbf{W} & 0 \\ 0 & 1 \end{bmatrix} \mathbf{W}_Q$. Here, a decomposition of \mathbf{U}_u has been formulated with a diagonal matrix, Σ_Q . This decomposition will be a SVD only if \mathbf{V}'_u and \mathbf{W}'_u are orthogonal matrices. For \mathbf{V}'_u , we have

$$\mathbf{V}'_u{}^T \mathbf{V}'_u = \mathbf{V}_Q^T \begin{bmatrix} \mathbf{V}^T \\ j^T \end{bmatrix} [\mathbf{V} j] \mathbf{V}_Q = \mathbf{V}_Q^T \begin{bmatrix} \mathbf{V}^T \mathbf{V} & \mathbf{V}^T j \\ j^T \mathbf{V} & j^T j \end{bmatrix} \mathbf{V}_Q. \tag{20}$$

Since $\mathbf{V}^T \mathbf{V} = \mathbf{I}$, the $\mathbf{V}^T j$ is computed as

$$\mathbf{V}^T j = \mathbf{V}^T \frac{c - \mathbf{V}d}{p} = \frac{1}{p} (\mathbf{V}^T c - \mathbf{V}^T \mathbf{V}d) = \frac{1}{p} (d - d) = \vec{0}. \tag{21}$$

Thus, we have $j^T \mathbf{V} = (\mathbf{V}^T j)^T = \vec{0}^T$. The $j^T j$ is determined as

$$j^T j = \left(\frac{h}{p} \right)^T \frac{h}{p} = \frac{h^T h}{p^2} = 1. \tag{22}$$

Since $\mathbf{V}_Q^T \mathbf{V}_Q = \mathbf{I}$, we have $\mathbf{V}'_u{}^T \mathbf{V}'_u = \mathbf{I}$, i.e. $\mathbf{V}_u = \mathbf{V}'_u$ is an orthogonal matrix. Similarly, due to $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ and $\mathbf{W}_Q^T \mathbf{W}_Q = \mathbf{I}$, we can prove \mathbf{W}'_u is also an orthogonal matrix as

$$\mathbf{W}'_u{}^T \mathbf{W}'_u = \mathbf{W}_Q^T \begin{bmatrix} \mathbf{W}^T & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{W} & 0 \\ 0 & 1 \end{bmatrix} \mathbf{W}_Q = \mathbf{W}_Q^T \begin{bmatrix} \mathbf{W}^T \mathbf{W} & 0 \\ 0 & 1 \end{bmatrix} \mathbf{W}_Q = \mathbf{I}. \tag{23}$$

Consequently, Proposition 1 is confirmed. \square

Algorithm 2 describes an enhanced incremental algorithm (EIA) for building a ROR online. This EIA differs from the standard incremental algorithm (SIA) by the additional process shown in lines 14–16. The SIA is as follows: After the projection of c onto \mathbf{V} , the matrix \mathbf{Q} is formulated, where small projections less than a threshold of \mathbf{tol} [46] are neglected to prevent the impact of round-off errors. A standard SVD then is applied to \mathbf{Q} before the updating process. It is then decided if the added column will increase the rank of the updated matrix (lines 8–13). Here the subscript denotes the index of row and column with starting from 1 ($V_{Q(1:k,1:k)}$ is a sub-matrix of V_Q with first k th rows and columns, for instance). The truncation of small singular values less than a prescribed threshold \mathbf{tol}_{sv} is used to improve the efficiency without affecting the accuracy of the updated modes. Finally, the updated modes will be re-orthogonalized if non-orthogonality occurs among them, which improves the robustness of the algorithm.

In practice, the number of POD modes necessary for building an accurate ROM is usually far fewer than the number of variables (N_v) and time steps (N_t). Thus we incorporate the truncation of a selected number of POD modes (M) into the incremental algorithm as shown in Algorithm 2 between line 14 and line 16, leading to the EIA. By virtue of using the EIA for all snapshots, we obtain an enhanced online algorithm (EOA) for building a ROR.

Algorithm 2 Enhanced incremental SVD for building a ROR

Input: $V \in \mathbb{R}^{n \times k}$, $\Sigma \in \mathbb{R}^{k \times k}$, $W \in \mathbb{R}^{k \times k}$, $c \in \mathbb{R}^{n \times 1}$, **tol**, **tol_{sv}**

```

1:  $k = \text{nColumns}(V)$ 
2:  $d = V^T c$ ,  $p = (|(c - Vd)^T(c - Vd)|)^{1/2}$ 
3: if  $p < \text{tol}$  then ▷ Neglect small projections
4:    $Q = \begin{bmatrix} \Sigma & d \\ 0 & 0 \end{bmatrix}$ 
5: else
6:    $Q = \begin{bmatrix} \Sigma & d \\ 0 & p \end{bmatrix}$ 
7:  $V_Q, \Sigma_Q, W_Q = \text{SVD}(Q)$ 
8: if ( $p < \text{tol}$ ) OR ( $k \geq n$ ) then ▷ Include matrices with more columns
9:    $V = V V_{Q(1:k, 1:k)}$ ,  $\Sigma = \Sigma_{Q(1:k, 1:k)}$ ,  $W = \begin{bmatrix} W & 0 \\ 0 & 1 \end{bmatrix} W_{Q(1:k+1, 1:k)}$ 
10: else
11:    $j = (c - Vd)/p$ 
12:    $V = [Vj]V_Q$ ,  $\Sigma = \Sigma_Q$ ,  $W = \begin{bmatrix} W & 0 \\ 0 & 1 \end{bmatrix} W_Q$ 
13:    $k = k + 1$  ▷ Consider  $M$  selected modes
14: if ( $k > M$ ) then
15:    $\Sigma = \Sigma_{(1:M, 1:M)}$ ,  $V = V_{(:, 1:M)}$ ,  $W = W_{(:, 1:M)}$ 
16:    $k = M$  ▷ Neglect small singular values
17: if ( $\Sigma_{(k-1, k-1)} > \text{tol}_{sv}$ ) AND ( $\Sigma_{(k, k)} < \text{tol}_{sv}$ ) then
18:    $k = k - 1$ 
19:    $\Sigma = \Sigma_{(1:k, 1:k)}$ ,  $V = V_{(:, 1:k)}$ ,  $W = W_{(:, 1:k)}$  ▷ Orthogonalize POD modes
20: if  $|V_{(:, k)}^T V_{(:, 1)}| > \min(\text{tol}, \text{tol} \times n)$  then
21:    $V = \text{ModifiedWeightedGramSchmit}(V)$ 

```

4. Numerical results

In this section, we present the validation of the proposed mesh adaptation strategy and describe the performance of AMR computations using POD-based RORs. We consider Eq. (1) with $\nu = 0.01$ and a QoI defined as $\bar{J}(u) = \frac{1}{T} (\sin(\pi x), u)_{\Omega \times 1}$. To avoid the influence of temporal discretization errors, a small time step, $\Delta t = 10^{-3}$, is used for solving both primal and adjoint problems. This value gives negligible time discretization errors for the complete range of AMR meshes considered in the numerical results. Adjoint-based error estimations are obtained using Eq. (13) with cubic spline reconstructions for the primal and adjoint solutions. We first verify the error estimation using a manufactured solution in Section 4.1. Then, the effectiveness of the proposed AMR strategy is validated on a Burgers problem with a multi-frequency forcing term in Section 4.2. The use of offline and online RORs is studied in Section 4.3 and Section 4.4, respectively.

4.1. Verification using a manufactured solution

First of all, the error estimation procedure is verified using a force f corresponding to the manufactured solution $u(x, t) = \sin^2(\pi t) \sin(\pi x)$, from $t = 0$ to $t = 20$. This exact solution is used to compute the actual value of the QoI.

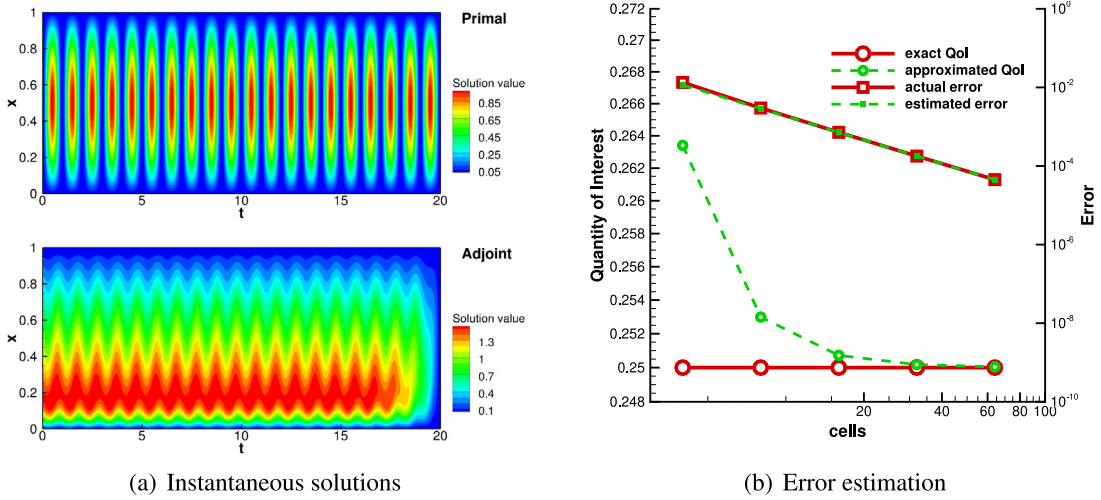


Fig. 5. Instantaneous solutions for a manufactured Burgers problem and associated error analysis of a volume-integrated output on uniformly refined meshes. Solid lines denote the values computed from analytical solutions while dashed lines are the approximation from numerical solutions and the adjoint-based error estimation with VMM.

Discrete solutions of the primal and adjoint problems are shown in Fig. 5(a). The adjoint solution is propagated in a direction opposite to the primal velocity. The most sensitive regions are not those with the largest primal solution or weighting function values, making it difficult to use feature-based AMR for this case. Fig. 5(b) depicts the QoI’s approximation and corresponding error as the mesh is refined uniformly. The computed QoI converges to the exact value and the error estimation displays a good agreement with the actual value.

4.2. Validation of the proposed adaptive strategy

We now introduce a forcing term f to produce a solution with large fluctuations and a boundary layer near the right boundary

$$f(x, t) = 1 + q(x) * \sum_{i=1}^{N_f} g_i(t) \sin(k_i x), \tag{24}$$

where $N_f = 3$ and the $g_i(t)$ are chosen so $|g_i(t)| \leq 1$, specifically,

$$g_1(t) = \sin(\pi t), k_1 = \pi, g_2(t) = \sin(2\pi t), k_2 = 2\pi, g_3(t) = \sin(3\pi t), k_3 = 3\pi. \tag{25}$$

$q(x) = 5/30$ is a coefficient used to tune the amplitude of the forcing term at various wave numbers so that the fluctuations can be controlled independently. The primal problem is advanced from $t = 0$ to $t = 20$ while the adjoint problem is solved backward from $t = 20$ to $t = 10$. The temporal interval ($t \in [0, 10]$) is sufficiently long to allow the primal flow problem to arrive to a statistically steady state. Instantaneous solutions are shown in Fig. 6. The primal solution changes periodically and a reverse propagation of the adjoint solution can be observed as well.

The mesh adaptation strategy of Section 2.4 is compared with uniform mesh refinement for this case. Here, the reference value of the QoI is calculated on a fine mesh with 256 elements, as there is no analytical solution. Fig. 7 presents the QoI and associated errors with increasing levels of mesh refinement determined by the proposed AMR procedure. The AMR technique is clearly more accurate than uniform enrichment for a given number of degrees of freedom. The corresponding error also converges faster but in a less regular manner.

4.2.1. Discussions of error estimation

The convergence of the error estimate is shown in Fig. 8(a), where we mark three different regions. In region 1, the error estimation is reasonably accurate compared to the actual error although the adjoint correction changes significantly. The adjoint correction is not enough to capture the dominant variation of error estimation in this

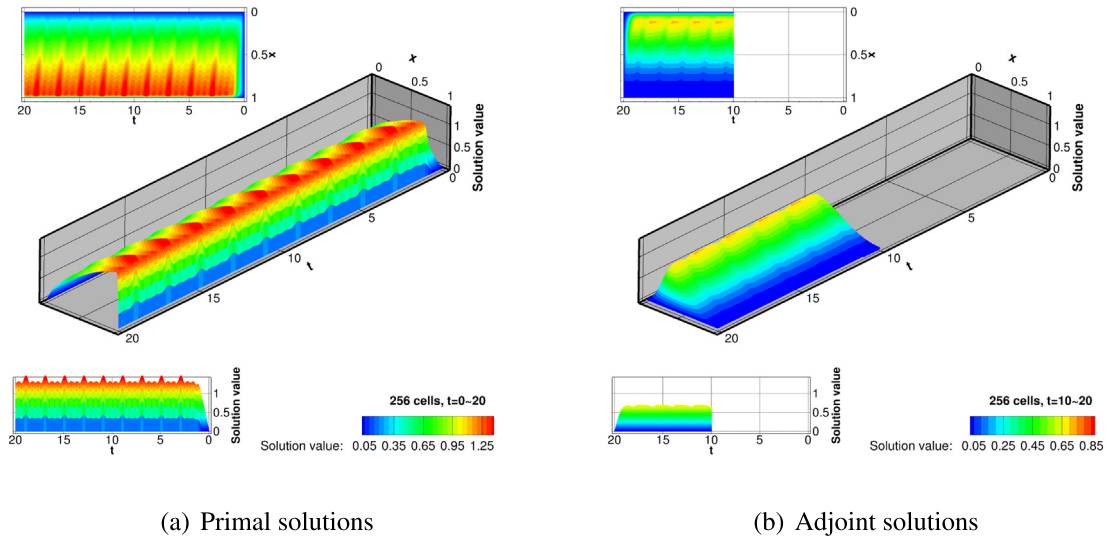


Fig. 6. The distribution of primal and adjoint solutions for the Burgers problem with a forcing term.

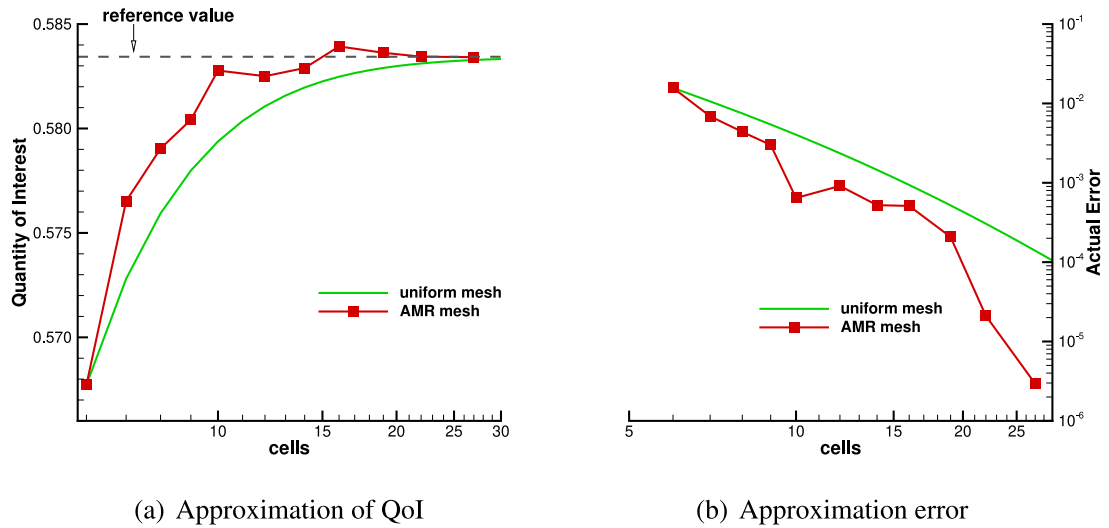


Fig. 7. AMR analyses compared with uniform refinement, (a) QoI and (b) associated approximating error.

region, which is also affected by the remaining error term. In region 2, a good error estimate is primarily obtained from the adjoint correction, although the remaining error still contributes favorably. In region 3, the error estimate is less reliable, while the adjoint correction has a trend similar to that of the total error estimation.

The computational mesh has few elements in region 1 and thus the accuracy of resolved solutions is limited. The VMM unresolved-scale model is thus vital for some cases in this region, leading to a significant influence of the remaining error. When combined with the adjoint correction, a good prediction is obtained. As the mesh is refined towards region 2, more of the solution is resolved and the adjoint correction begins to be dominant. This is actually the scenario we usually meet in literature or practical problems, i.e. we start AMR with a mesh that can capture important features based on experience and the adjoint correction is used to improve the computation of a QoI. In region 3, we reach a range of fine meshes with high spatial resolutions, where the computed QoI manifests an oscillatory convergence to the exact value. In this region, the remaining error is as important as the adjoint correction. Since the remaining error relies on the exact primal solution, u , the accuracy of the VMM unresolved-scale model can have a significant impact on its evaluation. Non-uniform meshes from AMR can affect

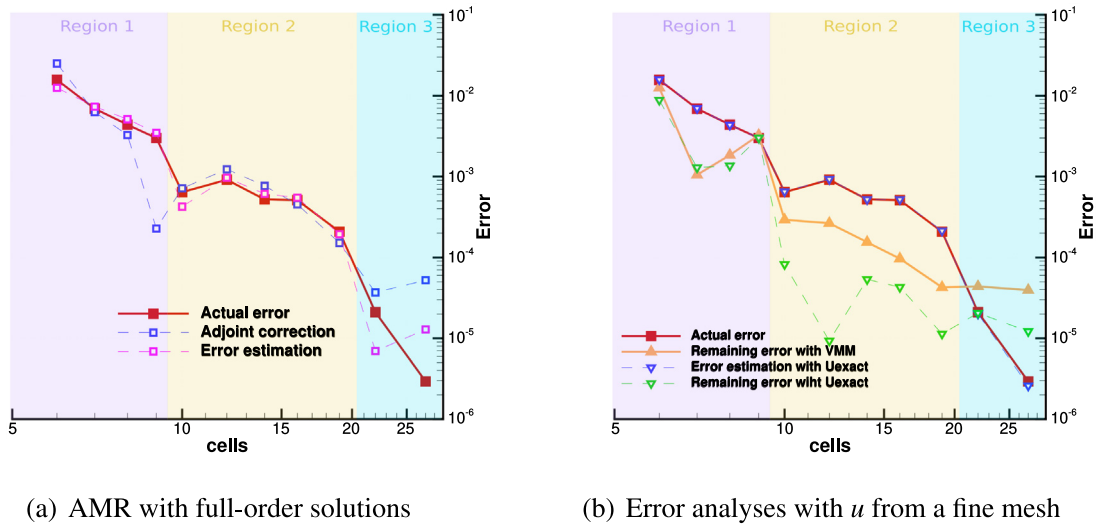


Fig. 8. Convergence of actual error, error estimation and adjoint correction during (a) AMR based on full-order primal solutions and (b) error analyses with u from a fine mesh (384 cells).

the reliability of the unresolved-scale model due to its reliance on the mesh size. When comparing the remaining error computed by VMM and an accurate reference u from a fine mesh (384 cells) as shown in Fig. 8(b), we can see that VMM overestimates the value, leading to a less reliable error estimation in region 3. In spite of this, the approximation of the QoI in region 3 is still improved as the actual error keeps reducing. This may be because the relative size of local error contributions can be recognized by the error estimation.

4.3. ROR-driven mesh refinement

We consider the same problem used in Section 4.2 to investigate how the ROR influences the performance of adjoint-based AMR. Since the error estimation in Section 4.2 is computed from full-order primal solutions, we refer to it as the baseline AMR. As an alternative, two RORs are considered, one with four POD modes and the other with one POD mode. Naturally, a ROR will produce a good representation of the primal solution if all significant POD modes are included. In this case, four POD modes are sufficient to capture 99.9% of the total energy (shown in Fig. 2). In contrast, the one-mode ROR is an extreme case, with a less accurate reconstruction of the primal solution but high computing efficiency. The mesh adaptation starts with a coarse mesh of 6 cells and terminates after 10 AMR levels with the aforementioned strategy.

4.3.1. ROR with four POD modes

The computation of the QoI from ROR-driven AMR is compared to that of the baseline AMR in Fig. 9(a). Both converge to the actual value in a quantitatively similar way, as do their approximation errors shown in Fig. 9(b). Fig. 10 demonstrates that the mesh refinement patterns computed based on the four-mode ROR agree well with those from the baseline AMR. Fig. 11(a) depicts the development of cumulative energy within various AMR levels. Four POD modes can still capture more than 99.9% of the total energy even for the fine meshes obtained during later AMR levels. Thus, the four-mode ROR-driven AMR does not behave differently from the baseline AMR.

4.3.2. ROR with one POD mode

We now study an extreme situation with only one mode used to construct the ROR. As before, Fig. 9 compares values of the QoI and their corresponding errors. The QoI's approximations from a one-mode ROR are similar to those from the baseline AMR, although there are differences at some AMR levels. But this AMR method is still much better than uniform refinement. From the mesh refinement pattern shown in Fig. 10, we can observe that the meshes obtained from the one-mode ROR AMR are not completely the same as those from the baseline AMR.

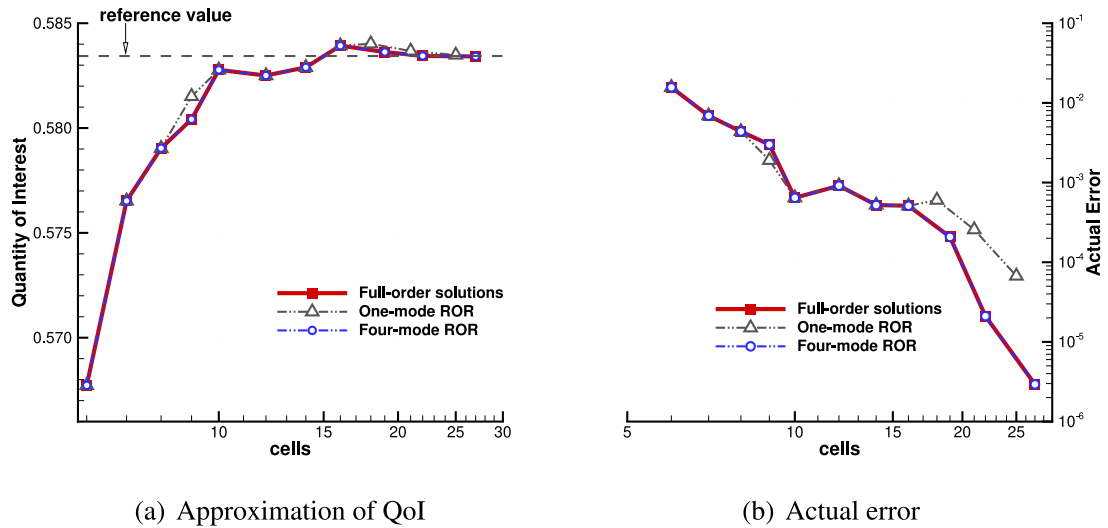


Fig. 9. Grid convergence of (a) the QoI and (b) the associated error from AMR based on full-order solutions (—■—), a four-mode ROR (—○—) and a one-mode ROR (—△—).

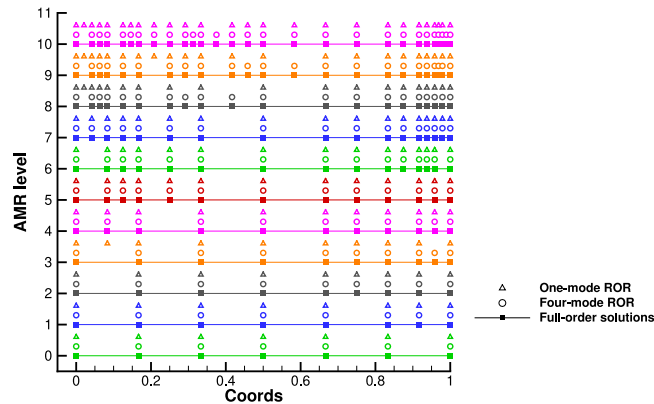


Fig. 10. Mesh refinement pattern for AMR based on full-order primal solutions (—■—), a four-mode ROR (○) and a one-mode ROR (△). Different colors denote different AMR levels.

The one-mode ROR changes the AMR sequence in coarse mesh regions at the 3rd AMR level, for example, but reaches the same computational mesh at 4th AMR level. On the other hand, the actual error is affected by using only one mode as the mesh becomes fine in later AMR levels (see Fig. 10). The cumulative energy for this one-mode ROR is shown in Fig. 11(b). Since first POD mode accounts for 65%–75% of total energy during AMR, it can describe main features of the primal solution over all AMR levels. Thus, the calculation from the one-mode ROR still considerably outperforms uniform refinement.

4.3.3. Impact of ROR truncation on adjoint solutions and error indicators

Fig. 12 presents discrete adjoint solutions on two different computational meshes encountered during AMR, a coarse mesh in Fig. 12(a) and a fine mesh in Fig. 12(b). The adjoint solutions are calculated based on full-order primal solutions, a four-mode ROR and a one-mode ROR, respectively. On a coarse mesh, the one-mode ROR is able to produce adjoint solutions with both features and magnitudes similar to those obtained using full-order solutions. This implies good error estimates. As the AMR proceeds to finer meshes, the one-mode ROR is unable to present high-wavenumber information and thus produces relatively smooth adjoint solutions, as shown in Fig. 12(b) for instance. Thus the error estimation with the one-mode ROR is affected. Conversely, the four-mode ROR includes both low- and high-wavenumber information and thus provides good estimates over both meshes.

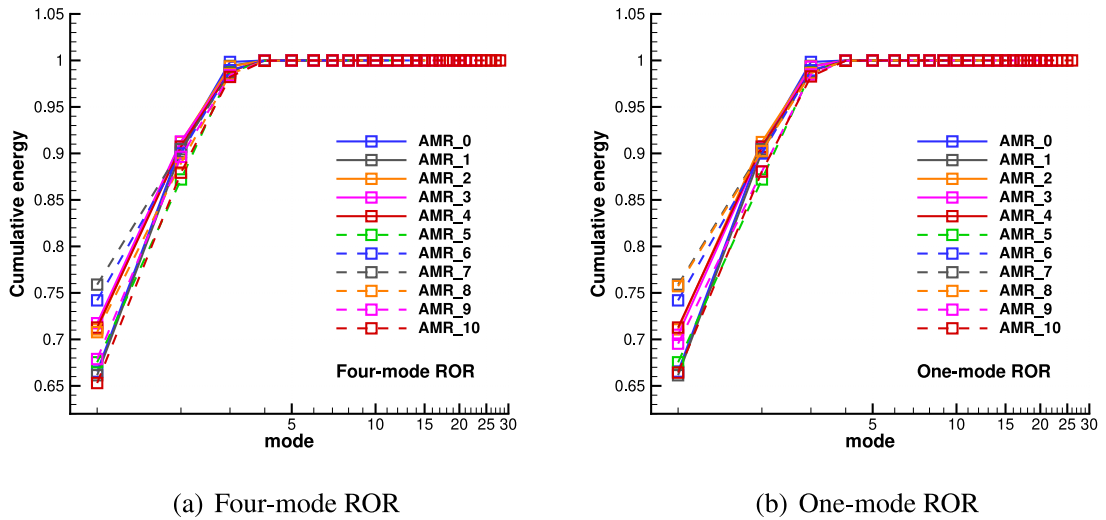


Fig. 11. Cumulative energy on different AMR levels computed with (a) a four-mode ROR and (b) a one-mode ROR.

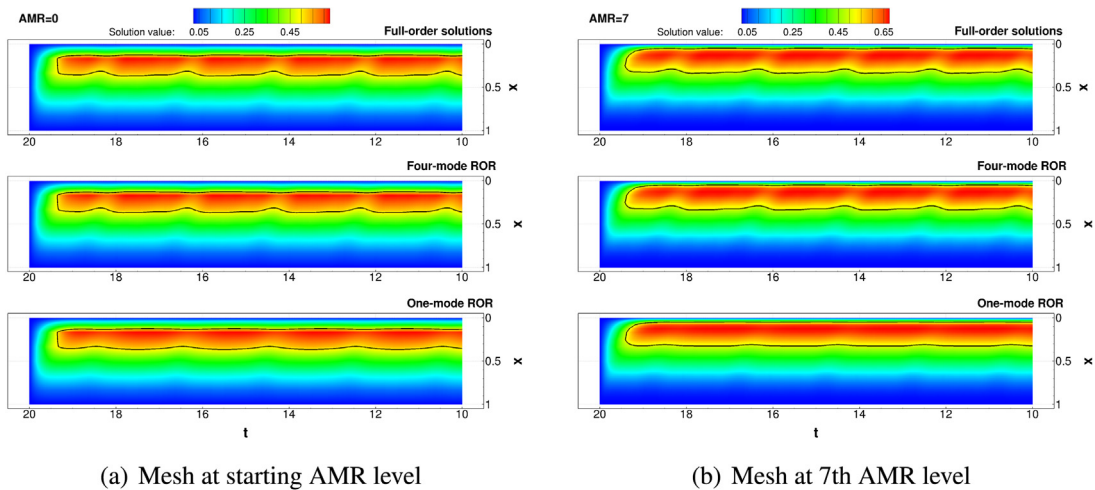


Fig. 12. Discrete adjoint solutions on (a) a coarse mesh at starting AMR level and (b) a fine mesh at the 7th AMR level, computed based on full-order primal solutions, a four-mode ROR and a one-mode ROR.

The primal solutions reconstructed from the RORs are also compared to the full-order one on the 7th AMR iteration in Fig. 13(a). We can observe that the one-mode ROR is able to give a good prediction for oscillations with large magnitudes but filters medium- and small-amplitude oscillations. The associated adjoint solution reflects this characteristic and becomes smoother as well. In that sense, their elemental error indicators η_e exhibit different distributions, as shown in Fig. 13(b). Based on full-order solutions, the algorithm marks cell 2 and cell 6 for refinement. However, the ones obtained using a one-mode ROR are cell 2 and cell 1. Consequently, a slightly different mesh is obtained in the one-mode ROR-driven AMR. This is because the high-wavenumber components of the primal solution play an important role in the error estimation on fine meshes, and those are not captured when using only one POD mode. It seems that using a very low-order ROR can be effective when the QoI’s calculation is not dependent on smaller scales of the primal solution. As shown in Fig. 13(b), a four-mode ROR, which includes higher-wavenumber features, is capable of reliably reconstructing the primal solution and the corresponding error indicators.

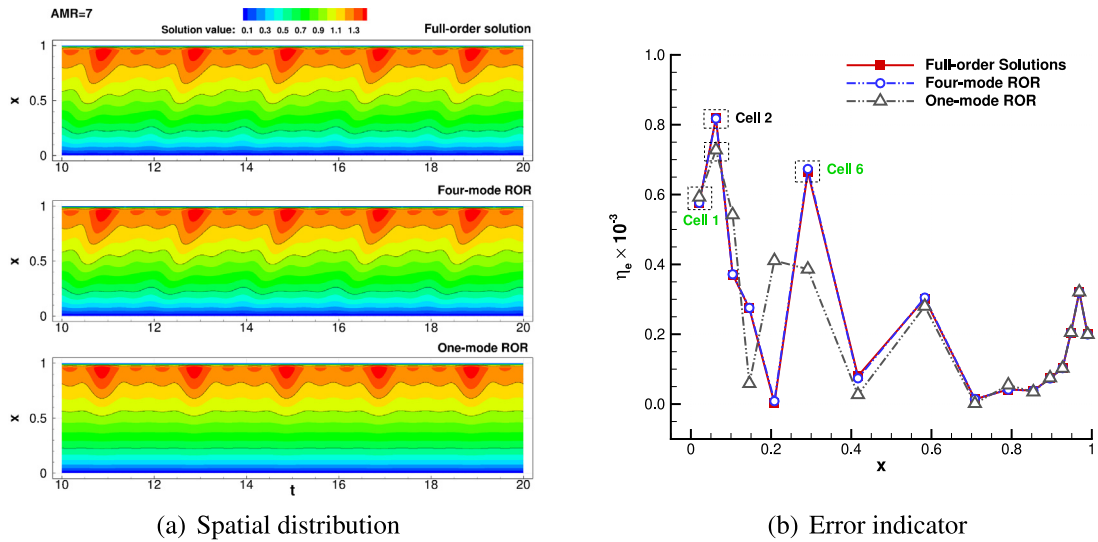


Fig. 13. Comparison of (a) the spatial distribution of full-order primal solutions and reconstructed primal solutions by a four-mode ROR and a one-mode ROR, and (b) their error indicators at the 7th AMR level.

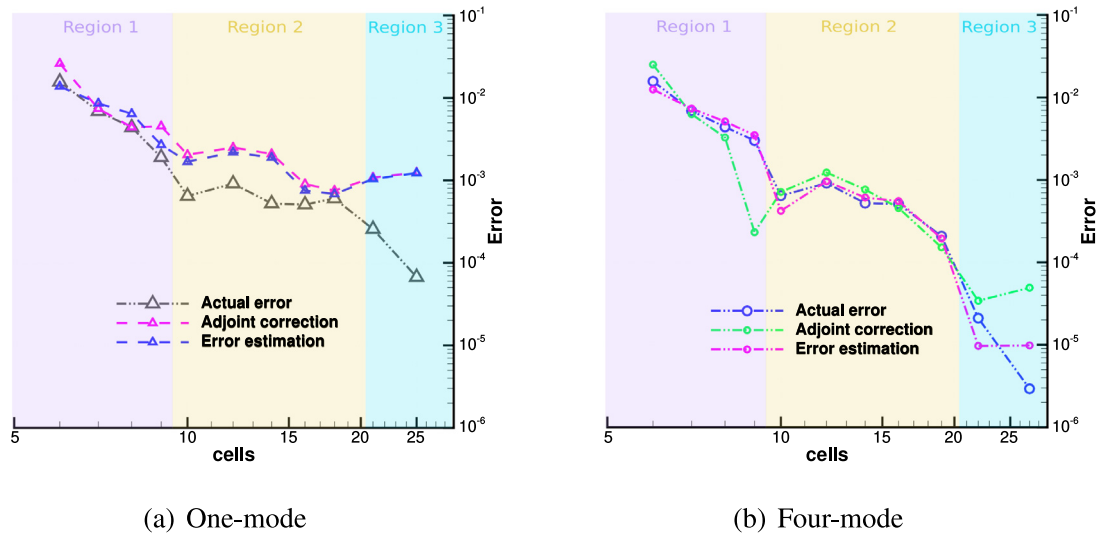


Fig. 14. Actual and estimated errors computed by ROR-driven AMR based on (a) one mode and (b) four modes.

4.3.4. Impact of ROR truncation on error estimation

Fig. 14 shows the change of error estimation and adjoint correction during AMR obtained using one-mode and four-mode RORs. The four-mode ROR-driven AMR has good error predictions in region 1 and 2, but overestimates errors in region 3, as does the baseline AMR. The error estimation from the one-mode ROR is accurate in region 1 but overestimates errors in region 2 and 3. Still, the first POD mode is capable of capturing the dominant solution feature crucial in region 2, and thus gives a sufficiently accurate estimate for AMR. But this advantage vanishes in region 3 where the high-frequency spatial information becomes important as well. The adjoint correction is the dominant error estimation term in last two regions, and is affected by the filtering of higher wavenumbers.

4.3.5. Impact of ROR truncation on memory efficiency

We define an efficiency metric, η , to quantify the memory reduction when utilizing a ROR. The memory required to store full-order primal solutions is proportional to $N_{full} = N_v N_t$ while a ROR needs $N_{ROR} = M N_v + N_v + M N_t$.

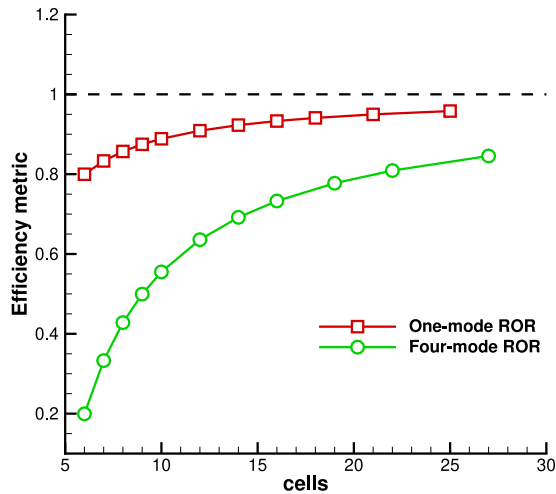


Fig. 15. Comparisons of the efficiency metric in one- and four-mode RORs during AMR ($N_t = 10001$).

The efficiency metric is then defined as

$$\eta = 1 - \frac{N_{ROR}}{N_{full}} = 1 - \frac{N_v + N_t}{N_v N_t} M - \frac{1}{N_t}. \tag{26}$$

A ROR requires less memory while $\eta > 0$, and vice versa. Furthermore, larger η indicates higher ROR efficiency, with the asymptote of $\eta_{asy} = 1 - \frac{1}{N_t}$. Fig. 15 shows the variation of this metric for one- and four-mode RORs during AMR. It is observed that the efficiency increases monotonically for both RORs as the mesh is refined, leading to a significant memory reduction. The one-mode ROR, as expected, is more memory efficient than the four-mode ROR while their difference is reduced as the mesh is refined.

4.4. Online ROR for AMR

In this section, we investigate the use of the EOA ROR and apply it on the same Burgers problem considered in Section 4.3. The online ROR without the enhanced process is referred to as the standard online ROR herein.

4.4.1. Impact of the EOA ROR

Fig. 16 compares of first eigenvalues from the offline, standard online and EOA RORs, as well as the shapes of their associated POD modes. The standard online ROR can produce results identical to those from an offline ROR using both one and four POD modes, validating the standard incremental algorithm.

The eigenvalues and POD modes from a one-mode EOA ROR agree well with the standard online ROR except for last two AMR levels. This reflects the characteristics mentioned at region 3, where high-wavenumber features become important on fine meshes. The truncation of higher-order POD modes during the single-mode EOA removes many of these features. Additionally, this truncation also has a notable impact on the distribution of POD modes at the final AMR level as shown in Fig. 16(a). In contrast, using the four-mode EOA ROR has no detrimental effect on both the eigenvalues and POD modes, as shown in Fig. 16(b). Therefore, one way to improve the representation of model interactions is to increase the number of POD modes used for the identification of EOA ROR.

The approximations of the QoI and its error from EOA ROR are presented in Fig. 17. We can observe that the EOA ROR does not alter the convergence history of the computation, preserving the same accuracy of calculations from the offline ROR-driven AMR technique, even for the one-mode case.

Although the POD mode obtained from the one-mode EOA ROR differs from that obtained with the offline ROR, the resulting QoI is still reasonably accurate, as shown in Fig. 17(a). In fact, the mesh refinement patterns are the same as those obtained with the offline SVD. In order to analyze the mechanism for this phenomenon, we choose three spatial positions with small (P1), medium (P2) and large (P3) mean values, as shown in Fig. 18. The QoI's good agreement is partially because the reconstructions of primal solutions from the offline ROR and EOA

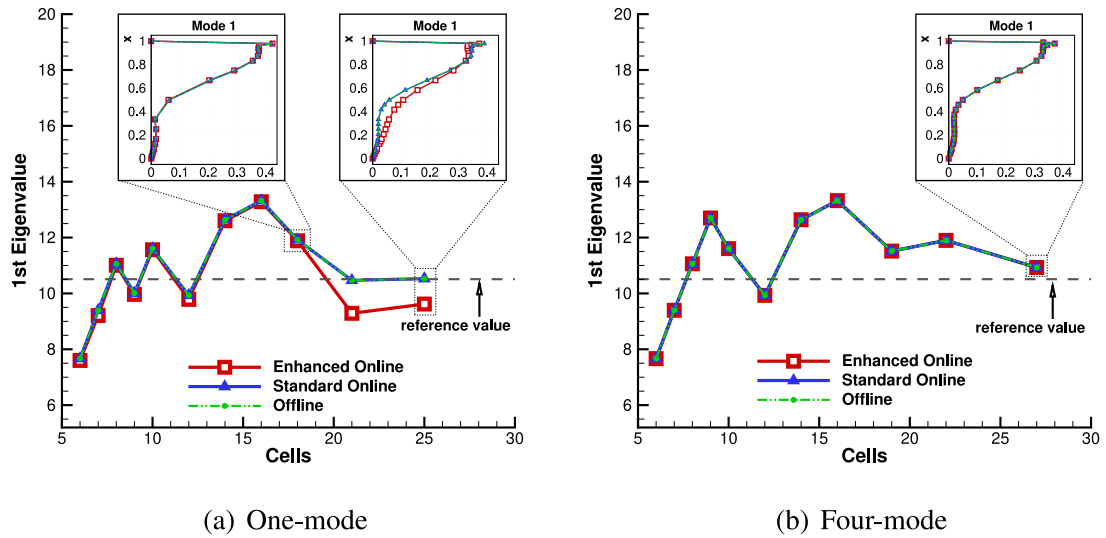


Fig. 16. Mesh convergence of the first eigenvalue and first POD mode for enhanced online (red), standard online (blue), and offline (green) ROR with (a) one and (b) four modes.

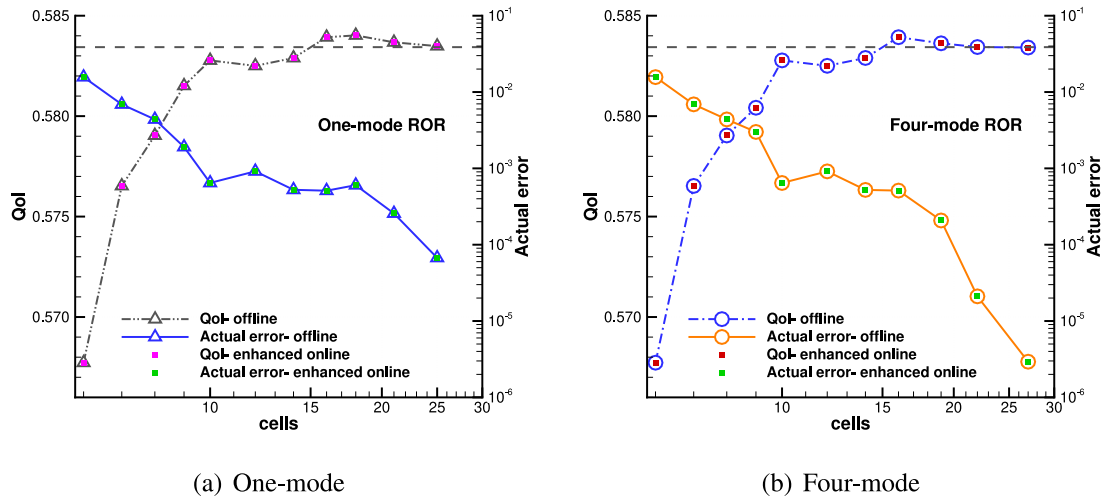


Fig. 17. Comparisons of QoI and actual error during AMR using enhanced online and offline ROR with (a) one and (b) four modes.

ROR have similar trends over time. This is true even at the 10th AMR level, as presented in Fig. 18(b), although there are differences in terms of their amplitudes at P2 and P1. The discrepancies result from only using one POD mode, which by design attempts to approximate instantaneous primal solutions with large dominant amplitudes and thus needs to sacrifice some accuracy of medium- and small-amplitude oscillations. Note that since the QoI is a statistical value over time, its calculation can benefit from the cancellation of temporal fluctuations. In addition, the EOA ROR provides the same accuracy as the standard online ROR until the 8th AMR level, as shown in Fig. 18(a).

4.4.2. Comparison of computing time

Fig. 19 compares the computing time to build a ROR by the offline, standard online and enhanced online algorithms during AMR. Fig. 19(a) shows this computing time for a one-mode ROR. The standard online ROR requires more computing time than the offline ROR. In contrast, the EOA ROR is much faster, and the advantage becomes more and more apparent as the mesh is refined. This property is also observed for the four-mode ROR, as shown in Fig. 19(b), although it is less efficient at the starting AMR levels. Generally, the computing time of

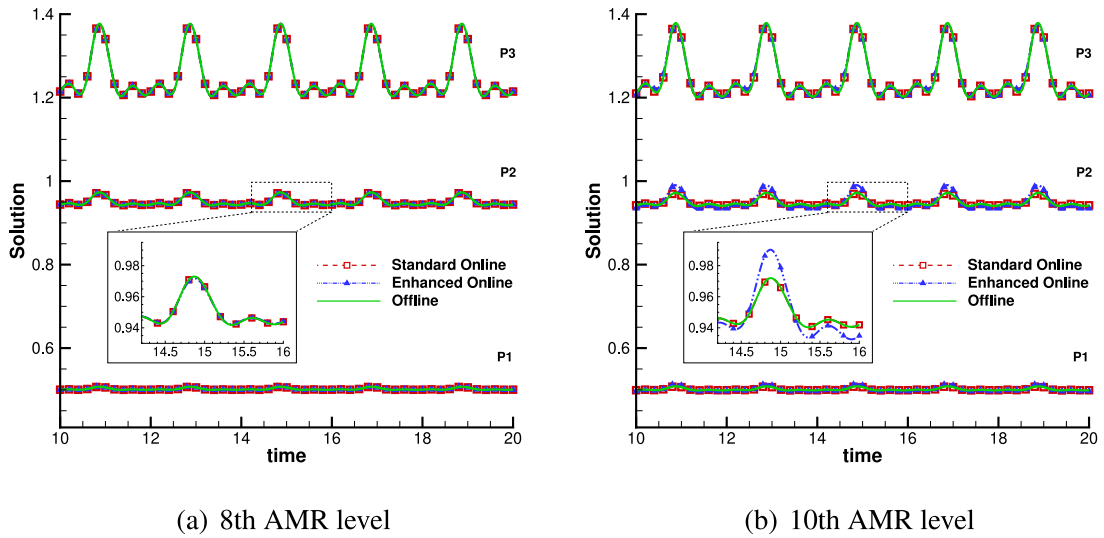


Fig. 18. Reconstructed primal solutions from standard online (red), enhanced online (blue), and offline (green) ROR with one POD mode at the 8th and 10th AMR level. P1, P2 and P3 denote three different spatial positions, i.e. P1 ($x = 1/6$), P2 ($x = 1/2$), P3 ($x = 5/6$).

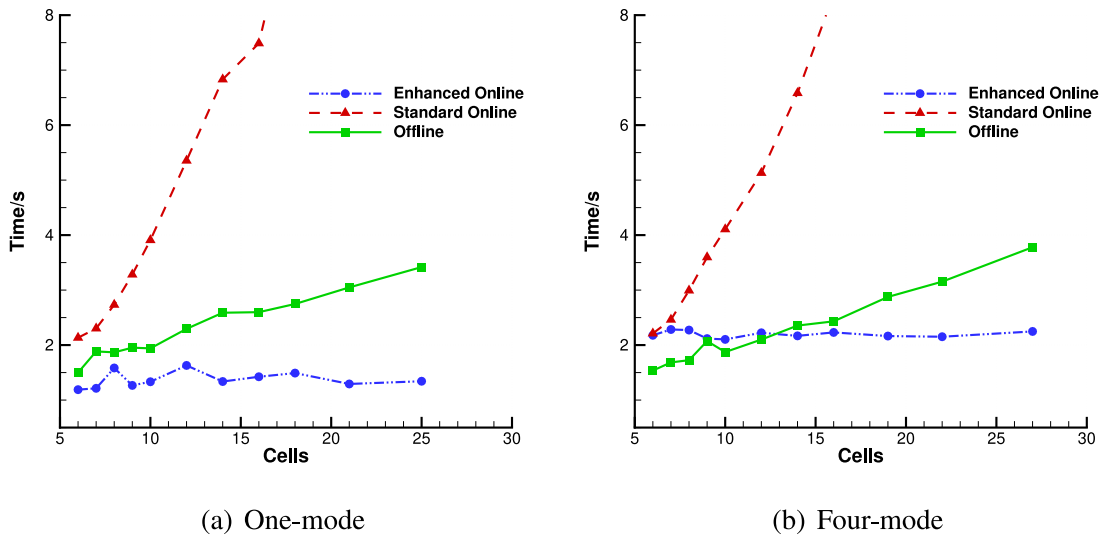


Fig. 19. Computing time of constructing offline (green), standard online (red) and enhanced online (blue) ROR with (a) one and (b) four POD modes during AMR.

the offline and standard online ROR increases as the mesh is refined. However, the computing time of the EOA ROR grows much more slowly. The results demonstrate that the EOA ROR could be promising for realistic LES applications.

4.4.3. Applicability of the enhanced online ROR to three-dimensional cases

As it is a purely data-based technique, the enhanced online ROR can be expected to deliver similar or greater reductions in the memory required for the adjoint-based mesh adaptation in three-dimensional problems.

For instance, Kitsios et al. [47] studied the turbulent flow at $Re = 30\,000$ over a NACA 0015 airfoil by LES, and applied POD analysis on 512 snapshots sampled every 250 time steps. They showed that only 200 POD modes are required to capture 90% of total energy. This translates into a storage requirement of only 0.16% of the memory required for full-order flow fields (a roughly three-order of magnitude reduction).

Pasquariello et al. [48] studied the low-frequency dynamics of shock-wave/turbulent boundary-layer interaction at $Re_{\delta_0} = 203\,000$. In this case, the full-order solution consisted of 4.2 million time steps and dynamic mode decomposition (DMD) was used to construct a low-order representation. Although DMD differs from POD, the incremental SVD technique can be applied for its determination [49]. Using one thousand of these modes, we can reconstruct the instantaneous primal solution with an L_2 -norm residual lower than 10^{-2} , cf. figure 17 in [48], while only requiring 0.24% of the memory resources of the complete flow solution. This level of accuracy in the primal solution is likely to be excessive when constructing error estimates for QoIs averaged over long time periods. In summary, one can anticipate dramatic memory savings when applying the proposed methodology to 3D problems. For many cases, such savings may in fact be essential for making adjoint-based AMR tractable.

5. Conclusions

We have developed an adjoint-based mesh adaptation strategy for unsteady problems augmented by an enhanced online algorithm for building a ROR. The effectiveness of combining adjoint-based error estimation with VMM has been verified using an unsteady Burgers problem with a manufactured solution. A POD-based ROR is introduced to represent the primal solution from the non-linear unsteady simulation so as to remove the storage bottleneck that would otherwise occur when we solve the adjoint problem backward in time. The procedure is evaluated using a Burgers problem with a multi-frequency forcing term. From numerical experiments, adjoint-based AMR is shown to be more efficient than traditional uniform refinement.

The results show that during AMR from very coarse meshes, the QoI estimate converges reliably, and reaches a high level of accuracy at moderate levels of refinement. For the cases considered here, using the ROR for primal solutions does not significantly affect the performance of AMR. Specifically, the adaptive results from a four-mode ROR, which can capture 99.9% of total energy, have good agreement with the results from a full-order solution-driven AMR. Using a single-mode ROR leads to suboptimal meshes, since the first POD mode accounts for only 65%–75% of total energy, but the AMR procedure still considerably outperforms uniform refinement.

The effectiveness of the one-mode ROR can be due to a relative insensitivity of the chosen QoI to high-frequency solution components, which tend to have smaller magnitudes in the current problem. However, this is not necessarily constrictive for LES applications, where large scales contain the majority of energy, and we are typically interested in relatively smooth time-averaged quantities, such as mean profiles. The enhanced online algorithm is shown to significantly improve the computing efficiency of the ROR-driven AMR procedure. Since the method is purely data based, and independent of the complexity of the governing equations, it is likely to also have a major impact on computational efficiency for more challenging problems, such as LES of complex flows.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors acknowledge financial support by the China Scholarship Council.

References

- [1] N.J. Georgiadis, D.P. Rizzetta, C. Fureby, Large-eddy simulation: current capabilities, recommended practices, and future research, *AIAA J.* 48 (8) (2010) 1772–1784.
- [2] J. Larsson, Q. Wang, The prospect of using large eddy and detached eddy simulations in engineering design, and the research required to get there, *Phil. Trans. R. Soc. A* 372 (2022) (2014) 20130329.
- [3] J. Slotnick, A. Khodadoust, J. Alonso, D. Darmofal, W. Gropp, E. Lurie, D. Mavriplis, *CFD Vision 2030 Study: A Path to Revolutionary Computational Aerodynamics*, Technical Report NASA/CR–2014-218178, National Aeronautics and Space Administration, Washington, DC 20546-0001, 2014.
- [4] M.J. Berger, A. Jameson, Automatic adaptive grid refinement for the Euler equations, *AIAA J.* 23 (4) (1985) 561–568.
- [5] M. Aftosmis, M. Berger, G. Adomavicius, A parallel multilevel method for adaptively refined Cartesian grids with embedded boundaries, in: 38th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, 2000.
- [6] C. Roy, Strategies for driving mesh adaptation in CFD (Invited), in: 47th AIAA Aerospace Sciences Meeting, Orlando, Florida, 2009.
- [7] A. Bazile, E. Hachem, J.C. Larroya-Huguet, Y. Mesri, Variational multiscale error estimator for anisotropic adaptive fluid mechanic simulations: application to convection–diffusion problems, *Comput. Methods Appl. Mech. Engrg.* (2018) 22.

- [8] D.A. Venditti, D.L. Darmofal, Grid adaptation for functional outputs: application to two-dimensional inviscid flows, *J. Comput. Phys.* 176 (1) (2002) 40–69.
- [9] M.A. Park, Adjoint-based, three-dimensional error prediction and grid adaptation, *AIAA J.* 42 (9) (2004) 1854–1862.
- [10] N. Pierce, M. Giles, Adjoint recovery of superconvergent functionals from PDE approximations, *SIAM Rev.* 42 (2) (2000) 247–264.
- [11] R. Becker, R. Rannacher, An optimal control approach to a posteriori error estimation in finite element methods, *Acta Numer.* 10 (1) (2001) 1–102.
- [12] J. Hoffman, C. Johnson, Stability of the dual Navier-Stokes equations and efficient computation of mean output in turbulent flow using adaptive DNS/LES, *Comput. Methods Appl. Mech. Engrg.* 195 (13–16) (2006) 1709–1721.
- [13] R.P. Dwight, Heuristic a posteriori estimation of error due to dissipation in finite volume schemes and application to mesh adaptation, *J. Comput. Phys.* 227 (5) (2008) 2845–2863.
- [14] M.A. Ceze, K.J. Fidkowski, High-order output-based adaptive simulations of turbulent flow in two dimensions, *AIAA J.* 54 (9) (2016) 2611–2625.
- [15] J. Cheng, H. Yue, S. Yu, T. Liu, Analysis and development of adjoint-based h-adaptive direct discontinuous Galerkin method for the compressible Navier–Stokes equations, *J. Comput. Phys.* 362 (2018) 305–326.
- [16] M.B. Giles, N.A. Pierce, Adjoint error correction for integral outputs, in: *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*, Springer, 2003, pp. 47–95.
- [17] K.J. Fidkowski, D.L. Darmofal, Review of output-based error estimation and mesh adaptation in computational fluid dynamics, *AIAA J.* 49 (4) (2011) 673–694.
- [18] M. Nemeec, M. Aftosmis, M. Wintzer, Adjoint-based adaptive mesh refinement for complex geometries, in: *46th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, 2008.
- [19] L. Shi, Z. Wang, Adjoint-based error estimation and mesh adaptation for the correction procedure via reconstruction method, *J. Comput. Phys.* 295 (2015) 261–284.
- [20] K.J. Fidkowski, D.L. Darmofal, Output-based adaptive meshing using triangular cut cells, *Tech. Rep. TR-06-2*, MIT Aerospace Computational Design Laboratory Report, 2006.
- [21] F. Alauzet, A. Loseille, A decade of progress on anisotropic mesh adaptation for computational fluid dynamics, *Comput. Aided Des.* 72 (2016) 13–39.
- [22] B.N. Granzow, M.S. Shephard, A.A. Oberai, Output-based error estimation and mesh adaptation for variational multiscale methods, *Comput. Methods Appl. Mech. Engrg.* 322 (2017) 441–459.
- [23] V.V. Garg, R.H. Stogner, Local enhancement of functional evaluation and adjoint error estimation for variational multiscale formulations, *Comput. Methods Appl. Mech. Engrg.* 354 (2019) 119–142.
- [24] A.C. Huang, H.A. Carson, S.R. Allmaras, M.C. Galbraith, D.L. Darmofal, D.S. Kamenetskiy, An adaptive variational multiscale method with discontinuous subscales for aerodynamic flows, in: *AIAA Scitech 2020 Forum*, Orlando, FL, 2020.
- [25] Q. Wang, J.-H. Gao, The drag-adjoint field of a circular cylinder wake at Reynolds numbers 20, 100 and 500, *J. Fluid Mech.* 730 (2013) 145–161.
- [26] Q. Wang, R. Hu, P. Blonigan, Least Squares Shadowing sensitivity analysis of chaotic limit cycle oscillations, *J. Comput. Phys.* 267 (2014) 210–224.
- [27] N. Chandramoorthy, P. Fernandez, C. Talnikar, Q. Wang, Feasibility analysis of ensemble sensitivity computation in turbulent flows, *AIAA J.* 57 (10) (2019) 4514–4526.
- [28] N. Chandramoorthy, Q. Wang, Sensitivity computation of statistically stationary quantities in turbulent flows, in: *AIAA Aviation 2019 Forum*, American Institute of Aeronautics and Astronautics, Dallas, Texas, 2019.
- [29] C. Talnikar, Q. Wang, G.M. Laskowski, Unsteady adjoint of pressure loss for a fundamental transonic turbine vane, *Trans. ASME, J. Turbomach.* 139 (3) (2017).
- [30] J. Hoffman, J. Jansson, N. Jansson, R.V. De Abreu, Towards a parameter-free method for high Reynolds number turbulent flow simulation based on adaptive finite element approximation, *Comput. Methods Appl. Mech. Engrg.* 288 (2015) 60–74.
- [31] M. Besier, R. Rannacher, Goal-oriented space–time adaptivity in the finite element Galerkin method for the computation of nonstationary incompressible flow, *Internat. J. Numer. Methods Fluids* 70 (9) (2012) 1139–1166.
- [32] K.J. Fidkowski, Y. Luo, Output-based space–time mesh adaptation for the compressible Navier-Stokes equations, *J. Comput. Phys.* 230 (14) (2011) 5753–5773.
- [33] M. Schmich, B. Vexler, Adaptivity with dynamic meshes for space–time finite element discretizations of parabolic equations, *SIAM J. Sci. Comput.* 30 (1) (2008) 369–393.
- [34] B. Flynt, D. Mavriplis, Discrete adjoint based adaptive error control in unsteady flow problems, in: *50th AIAA Aerospace Sciences Meeting*, Nashville, Tennessee, 2012.
- [35] J. Hoffman, C. Johnson, A new approach to computational turbulence modeling, *Comput. Methods Appl. Mech. Engrg.* 195 (23–24) (2006) 2865–2880.
- [36] J. Jansson, E. Krishnasamy, M. Leoni, N. Jansson, J. Hoffman, Time-resolved adaptive direct FEM simulation of high-lift aircraft configurations, in: *Numerical Simulation of the Aerodynamics of High-Lift Configurations*, Springer, Cham, 2018, pp. 67–92.
- [37] Q. Wang, P. Moin, G. Iaccarino, Minimal repetition dynamic checkpointing algorithm for unsteady adjoint calculation, *SIAM J. Sci. Comput.* 31 (4) (2009) 2549–2567.
- [38] M. Fosas de Pando, P.J. Schmid, D. Sipp, On the receptivity of aerofoil tonal noise: an adjoint analysis, *J. Fluid Mech.* 812 (2017) 771–791.
- [39] J. Chaudhry, D. Estep, M. Gunzburger, Exploration of efficient reduced-order modeling and a posteriori error estimation, *Internat. J. Numer. Methods Engrg.* 111 (2) (2017) 103–122.

- [40] M. Brand, Incremental singular value decomposition of uncertain data with missing values, in: European Conference on Computer Vision, Springer, 2002, pp. 707–720.
- [41] J. Caldwell, P. Wanless, A.E. Cook, A finite element approach to Burgers' equation, *Appl. Math. Model.* 5 (3) (1981) 189–193.
- [42] T.J. Hughes, G.R. Feijo, L. Mazzei, J.-B. Quinicy, The variational multiscale method—a paradigm for computational mechanics, *Comput. Methods Appl. Mech. Engrg.* 166 (1–2) (1998) 3–24.
- [43] Y. Bazilevs, V. Calo, J. Cottrell, T. Hughes, A. Reali, G. Scovazzi, Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows, *Comput. Methods Appl. Mech. Engrg.* 197 (1–4) (2007) 173–201.
- [44] Z. Wang, A. Oberai, Spectral analysis of the dissipation of the residual-based variational multiscale method, *Comput. Methods Appl. Mech. Engrg.* 199 (13–16) (2010) 810–818.
- [45] K. Taira, S.L. Brunton, S.T.M. Dawson, C.W. Rowley, T. Colonius, B.J. McKeon, O.T. Schmidt, S. Gordeyev, V. Theofilis, L.S. Ukeiley, Modal analysis of fluid flows: an overview, *AIAA J.* 55 (12) (2017) 4013–4041.
- [46] H. Fareed, J.R. Singler, Y. Zhang, J. Shen, Incremental proper orthogonal decomposition for PDE simulation data, *Comput. Math. Appl.* 75 (6) (2018) 1942–1960.
- [47] V. Kitsios, L. Cordier, J.-P. Bonnet, A. Ooi, J. Soria, On the coherent structures and stability properties of a leading-edge separated aerofoil with turbulent recirculation, *J. Fluid Mech.* 683 (2011) 395–416, (ISSN: 0022-1120, 1469-7645).
- [48] V. Pasquariello, S. Hickel, N.A. Adams, Unsteady effects of strong shock-wave/boundary-layer interaction at high Reynolds number, *J. Fluid Mech.* 823 (2017) 617–657, (ISSN: 0022-1120, 1469-7645).
- [49] D. Matsumoto, T. Indinger, On-the-fly algorithm for dynamic mode decomposition using incremental singular value decomposition and total least squares, 2017, [arXiv:1703.11004](https://arxiv.org/abs/1703.11004) [physics].