IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 14 December 2024, accepted 2 January 2025, date of publication 9 January 2025, date of current version 17 January 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3527630

RESEARCH ARTICLE

Accelerated Convergence Method for Flow Field Based on DMD-POD Combined Reduced-Order Optimization Model

JIANHUI LI [1], JUN HUANG [2], YAHUI SUN [2], AND GUOQIANG LI [1]
[1]School of Manufacturing Science and Engineering, Southwest University of Science and Technology, Mianyang, Sichuan 621010, China
[2]School of Computer Science and Technology, Southwest University of Science and Technology, Mianyang, Sichuan 621010, China
Corresponding author: Guoqiang Li (guoqli@ustc.edu.cn)

ABSTRACT This work presents a novel acceleration method that achieves more efficient convergence of steady-state flow fields. This method involves conducting dynamic mode decomposition (DMD) and proper orthogonal decomposition (POD) model reduction on the field snapshots. Subsequently, the residual of the reduced-order model is optimized in the POD modal space to obtain a more accurate solution. This optimized solution is then used as the initial field, and the solver continues iterating until the residual converges. Taking full advantage of both DMD and POD, the proposed approach removes the interference of high-frequency oscillatory flow components and concentrates on the main energy components. This effectively overcomes the problems of slow convergence and residual jumps caused by system stiffness, thereby accelerating the convergence process. The results show that for linear equations, the proposed method achieves a significant acceleration, with a convergence speed five times faster than traditional numerical methods. For the nonlinear Burgers equation, the proposed method also reduces the number of convergence steps by nearly 70%. Additionally, the performance of the proposed accelerated convergence method was further validated through the complex flow around a high-dimensional dual ellipsoid.

INDEX TERMS Computational fluid dynamics, combined reduced order model, accelerated convergence method.

I. INTRODUCTION

Recent decades have witnessed significant developments in computational fluid dynamics (CFD) that have permeated almost all areas of fluid mechanics, such as the simulation of supersonic flows [1], [2], [3], active flow control [4], [5], and power estimation of wind farms [6]. However, despite the increasing availability of computational resources, the cost of CFD is still prohibitively large in many engineering applications. Even for the Reynolds-averaged Navier–Stokes (RANS) equations [7], which are simpler than other methods, the simulation cost for complex problems remains unacceptably high [8]. Therefore, the development of efficient convergence algorithms is crucial in contemporary CFD research.

To date, numerous CFD methods have been developed for accelerating the model convergence process. The development of efficient time advancement algorithms stands out as a mainstream approach. The explicit Runge–Kutta method [9] is widely used for time advancement, but its stability limitations often result in low efficiency. In comparison, implicit time advancement techniques such as the Gauss–Seidel method [10], [11], [12], [13] and lower-upper symmetric Gauss–Seidel method [14], [15] have no restrictions on the time step size, leading to significantly improved computational efficiency. In addition, specific measures can be adopted during the flow field solution process to accelerate convergence, such as local time-stepping [16], [17] and residual smoothing [18]. Local time-stepping is a variant in which different time steps are used for each grid cell to satisfy the stability conditions. This approach substantially reduces the computational workload and enhances the

The associate editor coordinating the review of this manuscript and approving it for publication was Bijoy Chand Chatterjee.

convergence efficiency. In contrast, residual smoothing involves the filtering and averaging of residual terms to smooth out high-frequency oscillations and instabilities during the iterative process, thereby accelerating convergence. A second category of acceleration methods accelerate the iteration process by continuously optimizing the initial iterative value. The multigrid method [19] is representative of this approach. This method transfers solutions and residuals between grids of varying coarseness, eliminating low-frequency errors on finer grids by converting them into high-frequency errors on coarser grids. This continual improvement of the iterative initial values accelerates the convergence process. In a similar vein to the multigrid method, there are a number of acceleration methods based on reduced-order models [20], [21], [22]. Reduced-rank extrapolation [23] accelerates iterative solvers of the Euler equations and other computational problems [24], [25]. The core idea of this method is to utilize limited historical flow field information during the numerical iterations, and reconstruct the flow field in a lower-dimensional space using techniques like polynomial extrapolation or similar feature extraction techniques. This reconstruction provides a more accurate estimate of the initial values for iterations, significantly enhancing the convergence speed of numerical solutions.

Modal decomposition methods such as dynamic mode decomposition (DMD) [26], [27], [28] and proper orthogonal decomposition (POD) [29], [30], [31] have emerged as the primary techniques for reduced-order modeling in flow fields. Under linear assumptions, DMD is used to decompose the flow into a linear superposition of modes characterized by single-frequency flow patterns. By filtering out high-frequency oscillatory modes and utilizing the remaining low-frequency modes for flow field prediction, DMD substantially accelerates the convergence speed of steady-state solutions. Andersson [32] applied DMD to accelerate non-embedded solvers for steady-state problems, achieving a reduction of nearly 30% in the required number of iterations by predicting converged solutions under linear iterative assumptions. Liu [33] proposed a dual-time-stepping method based on DMD extrapolation. This method uses DMD to predict the initial conditions for the inner iterations within the dual-time-stepping format, thus accelerating the solution of the RANS equations. The mode multigrid (MMG) method [34], which combines a multigrid approach with vector extrapolation based on DMD, is another effective acceleration approach. Subsequent advances such as the improved mode multigrid method [35] and the adaptive mode multigrid method [36] have expanded the applicability of these approaches. The abovementioned acceleration methods for DMD are predicated on the assumption of linear system behavior. However, stiffness in linear systems limits the applicability of iterative methods constrained by explicit time-stepping, where small time steps are required over large intervals [37]. This constraint not only compromises the effectiveness of acceleration, but may also lead to residual divergence.

Unlike DMD, which decomposes the flow based on frequency, POD primarily analyzes the energy distribution within the flow field to extract the dominant flow modes that capture the vast majority of flow energy. The reduced-order modeling extrapolation based on POD involves extracting POD modes from snapshot data collected during iterations, thereby constructing a low-dimensional POD modal space. Within this reduced modal space, efficient solutions to the governing equations are sought, yielding lower residuals [20]. Kaminsky et al. [38] used POD to improve the convergence of explicit time-stepping formats, specifically in solving the Navier–Stokes equations. In contrast, Cao et al. [39] optimized the residuals of the governing equations using gradient optimization algorithms within the POD modal space, thus deriving more precise initial values for iterations and enhancing the convergence of traditional iterative methods through online dimensional reduction. Initially applied in fully implicit solvers, POD-based reduced-order extrapolation techniques have been further developed and refined, and has been successfully applied to accelerate the solution of both steady-state flows [40], [41] and transient flows [42], [43]. While POD effectively decomposes the flow into modes of different energy intensities and minimizes the average residual in the reconstructed flow field, it struggles to address the slow convergence caused by high-frequency oscillatory flows, as flow modes of different energy intensity inherently contain multiple flow frequencies.

In conclusion, for flow problems that exhibit both high-frequency oscillations and stiffness issues, using either DMD or POD alone is insufficient for accelerating the flow field solution. This paper proposes a novel method that combines DMD-based reduced-order extrapolation with POD-based online dimensional reduction. The resulting accelerated convergence method based on DMD-POD is used to accelerate the solution process for steady-state flow fields. The DMD reduced-order extrapolation method filters out high-frequency oscillatory components in the flow field, effectively reducing the number of iterations and the computational cost. The POD-based online dimensionality reduction optimization (ODRO) focuses on the dominant energy components of the flow field, thereby reducing unnecessary computational effort, and effectively addresses the slow convergence and residual jumps caused by the stiffness characteristics assumed in the DMD linear system hypothesis. This capability enables efficient acceleration and convergence of flow fields.

The remainder of this paper is organized as follows. Section II briefly introduces the numerical methods and theoretical foundations of the DMD-ODRO algorithm. Section III validates the performance of the DMD-ODRO algorithm using classical numerical examples. Finally, the conclusions from this study are presented in Section IV.

II. NUMERICAL METHOD

In CFD, high-frequency oscillatory flow components caused by turbulence, vortex shedding, and boundary layer instability directly impact the convergence rate of flow fields. These components exhibit rapid variations and large fluctuations, which hinder the decay of residuals during simulations, thereby prolonging the convergence times. To capture and simulate high-frequency oscillatory flow components, it is typically necessary to use finer grids, higher temporal resolutions, and more accurate numerical methods. However, this can significantly increase the computational costs. While these high-frequency oscillatory components undoubtedly influence the overall structure and performance of the flow, an excessive focus on them can, in some cases, reduce the accuracy of the solution. In practical applications, the average or low-frequency characteristics of the flow are often of greater interest than high-frequency fluctuations. Thus, effective strategies are needed to accelerate the decay of high-frequency components and reduce their influence on flow convergence. One approach is to employ appropriate numerical techniques. For instance, filtering can be applied to smooth out high-frequency fluctuations in the flow field. Alternatively, introducing suitable turbulence models can simulate high-frequency components in turbulent flows. DMD-based reduced-order models decompose the flow field into individual frequency components, facilitating the removal of high-frequency oscillatory modes and thereby accelerating flow field convergence. By effectively managing these high-frequency oscillatory components, such methods improve the convergence speed of flow fields while maintaining a reasonable level of computational efficiency. In this paper, the acceleration effects of the proposed method are compared against the original iterative method without reduced-order processing. The original iterative method employs the finite difference approach for discretization, where the inviscid and viscous terms are discretized using the central difference scheme and the temporal term is discretized using the forward difference scheme.

A. DMD TECHNIQUE

The DMD technique [26] is a data-driven approach used to extract the coherent flow structures and analyze the dominant flow dynamics. For flow field data obtained through experiments or numerical simulations, DMD can be expressed as a sequence of snapshots over N samples, denoted as $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \ldots, \boldsymbol{x}_N\}$, where $\boldsymbol{x}_i$ represents the ith snapshot. The DMD method assumes that a linear mapping matrix $\boldsymbol{A}$ connects two adjacent snapshots in time:

$$\boldsymbol{x}_{i+1} = \boldsymbol{A}\boldsymbol{x}_i, \tag{1}$$

where $\boldsymbol{A}$ is the system matrix of the high-dimensional flow field. According to the assumed linear mapping relationship, the matrix $\boldsymbol{A}$ reflects the dynamic characteristics of the system. First, using snapshots 1-N, two snapshot matrices can be constructed. Combining the assumption in (1), it can be inferred that:

$$\boldsymbol{Y} = [\boldsymbol{x}_2\ , \boldsymbol{x}_3, \ldots \boldsymbol{x}_N] = [\boldsymbol{A}\boldsymbol{x}_1\ , \boldsymbol{A}\boldsymbol{x}_2, \ldots \boldsymbol{A}\boldsymbol{x}_{N-1}] = \boldsymbol{A}\boldsymbol{X}. \tag{2}$$

To determine the orthogonal subspace of $\boldsymbol{A}$ for similarity transformations, a singular value decomposition (SVD) is performed on $\boldsymbol{A}$ to obtain:

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H \tag{3}$$

and

$$\boldsymbol{A} = \boldsymbol{U}\widetilde{\boldsymbol{A}}\boldsymbol{U}^H, \tag{4}$$

where $\boldsymbol{\Sigma}$ is a diagonal matrix with r singular values along the diagonal. The matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ obtained from the SVD satisfy $\boldsymbol{U}^H\boldsymbol{U} = \boldsymbol{I}$ and $\boldsymbol{V}^H\boldsymbol{V} = \boldsymbol{I}$. The computation of $\widetilde{\boldsymbol{A}}$ can be viewed as the minimization problem of the Frobenius norm:

$$\underset{A}{minimize} \left\|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}\right\|_F^2 . \tag{5}$$

Combining (3) and (4), (5) can be expressed as:

$$\underset{A}{minimize} \left\|\boldsymbol{Y} - \boldsymbol{U}\widetilde{\boldsymbol{A}}\boldsymbol{\Sigma}\boldsymbol{V}^H\right\|_F^2 . \tag{6}$$

Then, $\boldsymbol{A}$ can be approximated as:

$$\boldsymbol{A} \approx \widetilde{\boldsymbol{A}} = \boldsymbol{U}^H\boldsymbol{Y}\boldsymbol{V}\boldsymbol{\Sigma}^{-1}. \tag{7}$$

Because $\widetilde{\boldsymbol{A}}$ is similar to $\boldsymbol{A}$, the eigenvalues of $\widetilde{\boldsymbol{A}}$ are the dominant eigenvalues of $\boldsymbol{A}$. The eigenvalues of $\widetilde{\boldsymbol{A}}$ can be obtained through eigenvalue decomposition:

$$\widetilde{\boldsymbol{A}} = \boldsymbol{W}\boldsymbol{N}\boldsymbol{W}^{-1}, \boldsymbol{N} = \operatorname{diag}(\mu_1, \ldots, \mu_r) . \tag{8}$$

At this point, the snapshot at any instant in time can be expressed as:

$$\begin{aligned} x_i &= \boldsymbol{A}x_{i-1} = \boldsymbol{U}\widetilde{\boldsymbol{A}}\boldsymbol{U}^H x_{i-1} = \boldsymbol{U}\boldsymbol{W}\boldsymbol{N}\boldsymbol{W}^{-1}\boldsymbol{U}^H x_{i-1} \\ &= \boldsymbol{U}\boldsymbol{W}\boldsymbol{N}^{i-1}\boldsymbol{W}^{-1}\boldsymbol{U}^H x_1. \end{aligned} \tag{9}$$

Defining each column of $\boldsymbol{\Phi}$ as a DMD mode, we have:

$$\boldsymbol{\Phi} = \boldsymbol{U}\boldsymbol{W}, \tag{10}$$

and define the mode amplitude $\boldsymbol{\alpha}$ as:

$$\boldsymbol{\alpha} = \boldsymbol{W}^{-1}z_1 = \boldsymbol{W}^{-1}\boldsymbol{U}^H x_1, \boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_r]^T . \tag{11}$$

The amplitude represents the contribution of each mode to the initial snapshot. By sorting the DMD modes according to their amplitudes, the flow field at any instant in time can be predicted as follows:

$$x_i = \boldsymbol{\Phi}\boldsymbol{\Lambda}^{i-1}\boldsymbol{\alpha} = \sum_{j=1}^{r} \boldsymbol{\Phi}_j(\mu_j)^{i-1}\alpha_j. \tag{12}$$

As the DMD method sorts the flow field based on frequencies, the reconstructed flow field is the superposition of all frequency components in the modal space. Therefore, by performing DMD analysis on the flow field obtained during the iteration process and truncating the higher-frequency modes projected onto the modal space (i.e., ignoring all other modes except for the dominant low-order modes), the influence of these higher-order perturbations on the overall flow field can be effectively reduced. The truncated flow field is then back-projected into physical space to generate a new flow field as the initial condition for the next pseudo-time iteration. Because the higher-order perturbations have been removed, this new initial flow field will be closer to the physical state of the steady flow field, thus accelerating the convergence speed of the iteration process.

When using the DMD-based extrapolation method for solving numerical examples, especially nonlinear flows, the DMD-based method accelerates convergence compared with the original numerical method without dimension-reduction extrapolation, but may lead to significant jumps in residuals during the descent process. In this case, the original numerical method utilizes explicit finite difference schemes. As DMD assumes a linear system, it fundamentally considers the evolution of dynamical systems as changing linearly over time. When dealing with nonlinear flows, the dynamic characteristics of the system cannot be accurately captured because of the inherent stiffness features. Therefore, when predicting the flow field state at the next physical time step, the uncertainty of nonlinear characteristics may generate significant errors in the initial values of the predicted flow field, leading to residual jumps.

Based on (1), it is evident that DMD assumes a linear system. For a general linear system:

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{A}\boldsymbol{x}\left(t\right), \tag{13}$$

where $\boldsymbol{x} = \{x_1, x_2, \cdots, x_N\}^T \in \mathbb{R}^N, \boldsymbol{A} \in \mathbb{R}^{N\times N}$.

When $\boldsymbol{A}$ is non-singular, its condition number is defined as:

$$\kappa\left(\boldsymbol{A}\right) = \|\boldsymbol{A}\| \cdot \left\|\boldsymbol{A}^{-1}\right\|. \tag{14}$$

When the condition number κ is large, even small changes in the initial conditions can result in substantial fluctuations in the outcome. At this point, $\boldsymbol{A}$ is considered ill-conditioned. If $\boldsymbol{A}$ is a normal matrix, the condition number κ can be expressed as the ratio of the absolute values of its maximum and minimum eigenvalues:

$$\kappa\left(\boldsymbol{A}\right) = \left|\lambda_{\max}\left(\boldsymbol{A}\right)\right| / \left|\lambda_{\min}\left(\boldsymbol{A}\right)\right|, \tag{15}$$

where $\lambda_{\max}\left(\boldsymbol{A}\right)$ and $\lambda_{\min}\left(\boldsymbol{A}\right)$ are the maximum and minimum eigenvalues of $\boldsymbol{A}$, respectively.

If the eigenvalues of $\boldsymbol{A}$ in linear system (13) are $\lambda_j = \alpha_j + i\beta_j \left(j = 1, 2, \cdots, N, i = \sqrt{-1}\right)$, then the general solution to the system (13) can be expressed as:

$$\boldsymbol{x}\left(t\right) = \sum_{j=1}^{N} c_j \mathrm{e}^{\lambda_j t}\varphi_j + \psi\left(t\right), \tag{16}$$

where c_j is a constant that can be determined by the initial condition $\boldsymbol{x}\left(0\right)$ and $\psi\left(t\right)$ is a particular solution. If the real part of λ_j is $\alpha_j = Re\left(\lambda_j\right) < 0$, then as $t \to \infty, \boldsymbol{x}\left(t\right) \to \psi\left(t\right)$; in this case, $\psi\left(t\right)$ is a steady-state solution.

Therefore, when $Re\left(\lambda_j\right) < 0$ and

$$s = \max_{1\leq j\leq N}\left|Re\left(\lambda_j\right)\right| / \min_{1\leq j\leq N}\left|Re\left(\lambda_j\right)\right| \gg 1, \tag{17}$$

system (14) is termed an ill-conditioned, or stiff, equation, where s denotes the stiffness ratio.

When the stiffness ratio $s \geq 10$, the system is considered stiff. This often manifests as certain components of the equation changing much faster than others [44]. In the case of numerical solutions, this necessitates very small time steps to ensure stability and accuracy because the variables within the system have vastly different rates of change. When applying the DMD method to accelerate the processing of flow fields, $s \leq 1$ in the linear mapping matrix $\boldsymbol{A}$ can lead to unstable numerical solutions, characterized by oscillatory residuals or explosive growth. To maintain the stability of the numerical solution, extremely small time steps are typically required, significantly increasing the computational cost and time [45]. In addressing the stiffness effects caused by the DMD method, this paper integrates POD-based ODRO to smooth the residuals, thereby preserving the stability of the numerical solution.

B. POD TECHNIQUE

POD [46] is a linear method that projects high-dimensional systems onto low-dimensional state spaces using orthogonal modal projections while ensuring the minimum residual with a given number of modes. Consider a snapshot matrix:

$$\boldsymbol{X} = \left[\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots \boldsymbol{x}_N\right]. \tag{18}$$

The snapshot at any time can be expressed as a combination of the mean flow and the fluctuating flow $\boldsymbol{x}_i'$:

$$\boldsymbol{x}_i = \bar{\boldsymbol{x}} + \boldsymbol{x}_i'. \tag{19}$$

The fluctuating flow is represented by a linear combination of low-order POD modes:

$$\boldsymbol{x}_i' = \sum_{j=1}^{N} a_j^i \xi_j, \tag{20}$$

where N represents the number of flow field snapshots, a_j^i represents the modal coefficient corresponding to the jth POD mode at time i, and ξ_j represents the jth POD mode. $\boldsymbol{X}' = \left[x'_1, x_2', \ldots x'_N\right]$ represents the snapshot matrix composed of the temporal sequence of fluctuating flow snapshots. The correlation matrix $\boldsymbol{C}$ can be defined as:

$$\boldsymbol{C} = \boldsymbol{X}'^T\boldsymbol{X}'. \tag{21}$$

To solve for the eigenvalue matrix $\lambda = diag(\lambda_1, \lambda_2, \ldots, \lambda_N)$ and eigenvector matrix $\boldsymbol{Q} = \left[\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots \boldsymbol{q}_N\right]$ of a correlation matrix $\boldsymbol{C}$, we have:

$$\boldsymbol{C}\boldsymbol{Q} = \lambda\boldsymbol{Q}. \tag{22}$$

The POD modes $\xi = \left[\xi_1, \xi_2, \ldots \xi_N\right]$ are represented as:

$$\xi_j = \frac{1}{\sqrt{\lambda_j}}\boldsymbol{X}'\boldsymbol{q}_j. \tag{23}$$

Correspondingly, the modal coefficient a^i for the jth POD mode can be expressed as:

$$\boldsymbol{a}_j = \left[a_j^1, a_j^2, \ldots a_j^N\right] = \xi_j{}^T\boldsymbol{X}'. \tag{24}$$

The magnitudes of the eigenvalues correspond to the contribution of the respective mode's energy. By sorting the modes based on eigenvalue magnitudes, the leading m dominant modes that occupy a significant portion of the energy are obtained. Therefore, the low-order representation of the flow field at any time instant can be expressed as:

$$\boldsymbol{x}_i = [\xi_1 , \xi_2, \dots \xi_m] \left[a_1^i , a_2^i, \dots a_m^i\right] + \bar{\boldsymbol{x}}. \tag{25}$$

After dimensionality reduction through POD, the evolution of the original high-dimensional flow field can be effectively represented as a linear combination in the reduced POD modal space. Despite reducing the degrees of freedom to the number of POD modes, POD alone cannot fundamentally address the slow convergence caused by disturbances from high-frequency oscillatory flows. This limitation arises because POD sorts modes by their energy magnitudes and truncates based on the dominant flow patterns, potentially retaining components with varying flow frequencies. Thus, POD may preserve disturbance frequency information without effectively eliminating the effects of high-frequency oscillations.

POD-based online optimization method [39] aims to mitigate some effects of high-frequency oscillations by minimizing the flow field residuals. This method optimizes the modal coefficients within the low-dimensional POD modal space. The objective shifts from directly solving the original high-dimensional governing equations to minimizing the residuals in the POD modal space. The residual is considered as the absolute average of all grid residuals:

$$R_{total}(\xi) = \frac{1}{n} \sum\nolimits_{i=1}^{n} \|R_i(\xi)\|, \tag{26}$$

where $R_i(\xi)$ represents the residual of the control equations at the grid in the CFD solver.

In the POD modal space, the coefficients ξ serve as optimization variables. The objective function $R_{total}(\xi)$ is computed by evaluating the total residual obtained from inputting the POD-reconstructed flow variables $\boldsymbol{x} = \boldsymbol{a}\xi + \bar{\boldsymbol{x}}$, calculated using the optimized POD modal coefficients ξ, back into the CFD solver. The goal is to adjust ξ such that the reconstructed flow field in the POD modal space satisfies the optimization criteria, i.e., minimizing the objective function $R_{total}(\xi)$. The optimization process employs the Nelder–Mead algorithm [47], a gradient-free optimization method that is suitable for multidimensional unconstrained optimization tasks due to its high efficiency in low-dimensional spaces. Subsequently, the optimized modal coefficients ξ are used to reconstruct the flow variables, which are then fed back into the solver for subsequent iterations. The reduced residual in the reconstructed initial flow field significantly reduces the number of iterations required for convergence, thereby enhancing the convergence efficiency. By optimizing $R_{total}(\xi)$ across the entire grid, this process smooths out extreme values of the local residuals, effectively mitigating or alleviating the effects of high-frequency oscillations and accelerating the convergence of steady-state solutions.

Algorithm 1 DMD-POD Combined Reduced-Order Optimization Acceleration Method

```
Input: Initial field variable $u_initial$, sampling interval $K$,
    number of snapshots $N$, DMD truncation order $r$, POD
    retention order $o$, convergence criterion $\epsilon$
Output: Process residual $res$, converged result $u$
Initialize $u$ as the initial field variable
repeat
    Compute the current flow field residual $res = Res(u)$
    Update the flow field variable $u$ =
    $TimeMarching(u, res)$ iteratively in the solver
    Record the average value of the current residual $res$
    if iteration count $i\%K == K - 1$ then
        Convert $u$ into vector form and store it in the
        corresponding column of the snapshot matrix $S$
    else if iteration count $i\%(N * K) == N * K - 1$ then
        Perform DMD on the snapshot matrix $S$,
        reconstructing the predicted flow field $U$
        Perform POD on the reconstructed flow field $U$ for
        dimensionality reduction
        Minimize the total flow field residual $R_{total}(\xi)$ in
        the POD modal space and update the flow field $u$
    else
        $i = i + 1$
    end if
until $res \leq \epsilon$
```

In contrast to the residue jumping phenomenon that may occur with DMD reduced-order extrapolation methods under nonlinear flow conditions, the convergence curve of POD-based online dimensionality reduction and acceleration methods tends to be smooth, resulting in more pronounced acceleration effects. Although POD operates under linear assumptions [48], the integration of optimization algorithms effectively suppresses computational instabilities caused by system stiffness. However, compared with DMD reduced-order extrapolation methods, POD-based online dimensionality reduction and acceleration methods show only incremental acceleration. There remains significant potential for improvement, especially when dealing with complex engineering scenarios.

C. DMD-ODRO

Eliminating the influence of high-frequency oscillations in the flow field significantly improves the convergence efficiency of numerical computations. In this regard, DMD has a unique advantage as it transforms complex dynamic systems into low-dimensional representations and extracts modes characterized by singular frequencies. This facilitates the straightforward filtering of high-frequency oscillatory modes based on the single-frequency nature of DMD modes, thereby effectively enhancing the computational efficiency for steady-state flow fields. However, the linearity assumption means that DMD encounters challenges when applied to nonlinear or unstable flows, where rigidities in the process can lead to significant discrepancies between predicted and actual flow behaviors, imposing stringent limitations on the time step size. While POD also operates under a linear assumption, its ODRO process smooths out residual minima across the entire grid, thereby mitigating instabilities arising from rigidities. POD cannot fundamentally eliminate the effects of high-frequency oscillations, so its acceleration effects remain somewhat unsatisfactory.

We wish to fundamentally eliminate the influence of high-frequency oscillatory components in flow fields while avoiding potential instabilities due to the linear system assumptions, and to effectively achieve accelerated convergence of steady-state flow field solutions. For this purpose, the present paper proposes a flow field accelerated convergence method based on a combined DMD-POD reduced-order optimization model, named DMD-ODRO. The CFD solver and the combined reduced-order optimization model are used alternately to solve flow field problems. The snapshots required for flow field reduction are obtained through CFD iterations, and high-frequency modes are filtered by the DMD method. The initial acceleration is achieved through extrapolation and reconstruction using low-frequency modes. After low-frequency mode reconstruction, the snapshots are further reduced by POD and the modal coefficients are optimized in the low-order POD modal space to achieve an additional acceleration of the flow field calculations. After reduced-order optimization, the flow field is substituted back into the CFD solver for further iterations.

In order to demonstrate the effectiveness of the model in solving stiff systems of equations, a simple ordinary differential equation (ODE) with strong stiffness is now used for calculation. There is a given system:

$$\begin{cases} u' = -1000.25u + 999.75v + 0.5 \\ v' = 999.75u - 1000.25v + 0.5 \end{cases} \tag{27}$$

and the initial condition is:

$$u\,(0) = 1, v\,(0) = -1. \tag{28}$$

The eigenvalues of the coefficient matrix $\boldsymbol{A}$ on the right side of the equation can be obtained as $\lambda_1 = -0.5$, $\lambda_2 = -2000$. The stiffness ratio $s = \frac{\max\limits_{1\le j\le N} |\lambda_j|}{\min\limits_{1\le j\le N} |\lambda_j|} = 4000 \gg 1$, it is a strongly stiff ODE.

The analytical solution of the equation is:

$$\begin{cases} u\,(t) = -\mathrm{e}^{-0.5t} + \mathrm{e}^{-2000t} + 1 \\ v\,(t) = -\mathrm{e}^{-0.5t} - \mathrm{e}^{-2000t} + 1 \end{cases} \tag{29}$$

when $t \to \infty$, the stationary solution of the equation is $u\,(t) \to 1, v\,(t) \to 1$. Both u and v contain fast-varying component e^{-2000t} and slow-varying component $\mathrm{e}^{-0.5t}$, the rate of change of the solution components differ greatly, When $t = 0.005$, the slow-varying component can decay to $\mathrm{e}^{-10} \approx 0$, but only when $t = 20$ can the fast-varying component decay to $\mathrm{e}^{-10} \approx 0$. That is to say, the system of equations must be calculated until $t = 20$ in order to obtain the stationary solution. If an explicit calculation method is used for calculation, the time step should be made as small as possible to ensure the stability of the calculation. And to obtain the stationary solution, a large number of iteration steps are required. In contrast, the DMD-ODRO method can effectively avoid the instability caused by the stiffness problem even in the case of block iteration steps. As shown in Fig. 1, the residual is the difference between the numerical solution and the stationary solution. Under the same time step, the DMD-ODRO method can obtain the stationary solution faster than the traditional explicit method. The absolute error between the steady-state solution obtained by the traditional method and the analytical solution is $1.99\mathrm{e}^{-3}$, while the absolute error of the DMD-ODRO method is only $1.87\mathrm{e}^{-6}$, demonstrating higher computational accuracy.

The pseudo-code for the proposed DMD-ODRO method is presented in Algorithm 1.

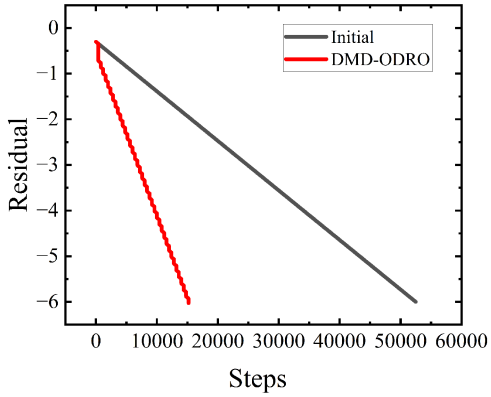


FIGURE 1. **The change of residuals when solving the stationary solution of stiff ordinary differential equation.**

III. RESULTS

This section validates the effectiveness and sensitivity of the proposed accelerated convergence method using typical model equation examples, namely a 1D diffusion equation, 1D advection–diffusion equation, and 1D Burgers' equation. Explicit time-stepping methods are employed for all cases, with the time term discretized using a first-order forward difference scheme, the advection term discretized using a first-order central difference scheme, and the diffusion term discretized using a second-order central difference scheme.

A. ONE-DIMENSIONAL DIFFUSION EQUATION

Consider the 1D constant-coefficient diffusion equation:

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}, \tag{30}$$

where u represents the flow velocity and α denotes the diffusion coefficient. The solution domain is defined as $x \in [-1, 1]$, with the boundary conditions and initial conditions

TABLE 1. Computation times for different methods under various diffusion coefficients α in the 1D diffusion equation.

	$\alpha = 0.05$		$\alpha = 1$		$\alpha = 10^4$		$\alpha = 10^7$	
	Iterations	Time(s)	Iterations	Time(s)	Iterations	Time(s)	Iterations	Time(s)
Initial	58420	19	60074	19	60074	19	60074	21
DMD	26558	7	27371	8	27335	8	27400	9
ODRO	19928	7	22101	9	19575	7	22254	10
DMD-ODRO (ours)	11256	4	12097	4	11904	4	11588	5

TABLE 2. Average error of different methods under various diffusion coefficients α in the 1D diffusion equation.

	Error ($\alpha = 0.05$)	Error ($\alpha = 1$)	Error ($\alpha = 10^4$)	Error ($\alpha = 10^7$)
Initial	1.9958×10^{-13}	9.8575×10^{-14}	9.8575×10^{-14}	9.8575×10^{-14}
DMD	2.0193×10^{-13}	9.7801×10^{-14}	9.7644×10^{-14}	9.6786×10^{-14}
ODRO	1.9950×10^{-13}	9.8471×10^{-14}	9.8552×10^{-14}	9.8646×10^{-14}
DMD-ODRO (ours)	1.8476×10^{-13}	9.7748×10^{-14}	9.8419×10^{-14}	9.7533×10^{-14}

specified as:

$$\begin{aligned} u(-1,t) = 0.5, u(1,t) = -0.5 \\ u(x,0) = x. \end{aligned} \tag{31}$$

The Courant–Friedrichs–Lewy (CFL) number is defined as:

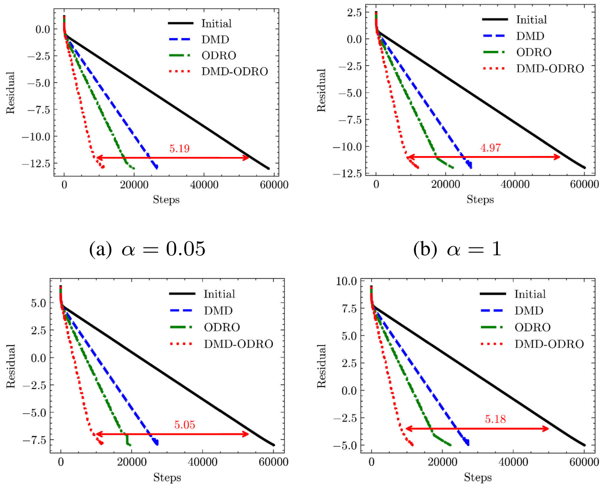
$$\mathrm{CFL} = \alpha \frac{\Delta t}{\Delta x^2}, \tag{32}$$

where Δt denotes the time step size and Δx represents the spatial step size.

1) EXPERIMENT 1: NUMERICAL SIMULATIONS WITH DIFFERENT DIFFUSION COEFFICIENTS α AND A FIXED GRID PARTITIONING

The grid is set to have 200 divisions and the CFL number is set to 0.5. Flow field snapshots are saved every 50 iterations, and sets of five consecutive snapshots are used to construct a flow field matrix for model reduction. The convergence history of flow field residuals for different methods and parameters is shown in Fig. 2. The computation times for different methods with varying diffusion coefficients are summarized in Table 1.

As shown in Fig. 2 and Table 1, among the various methods considered herein, DMD-ODRO exhibits the best acceleration effect. Compared with the traditional finite difference method, DMD-ODRO achieves a sixfold acceleration ratio and a fivefold reduction in computation time. Additionally, the convergence curves for the different acceleration methods retain consistent shapes across various diffusion coefficients. This indicates that, under a constant CFL number, the sensitivity of the acceleration methods to changes in the diffusion coefficient is not significant. With an increase in the diffusion coefficient, the final converged residual gradually increases. This is because an increased diffusion coefficient leads to faster convergence of the physical process towards a stable or uniform state, thereby accelerating the numerical solution process.



(a) $\alpha = 0.05$

(b) $\alpha = 1$

(c) $\alpha = 10^4$

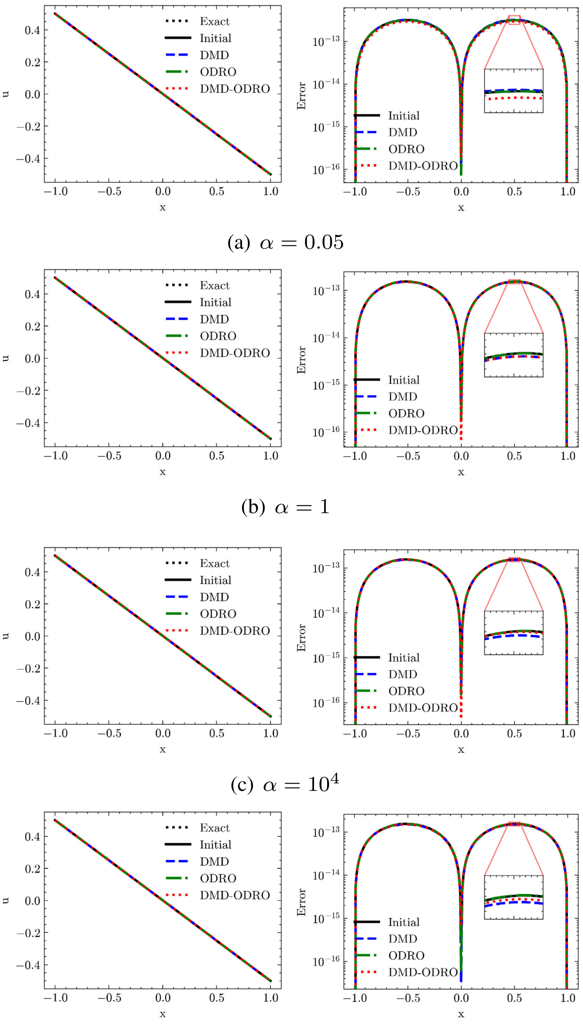
(d) $\alpha = 10^7$

FIGURE 2. Convergence of residual for different methods under various diffusion coefficients α in the 1D diffusion equation.

Fig. 3 compares the final converged results of the different methods with the analytical solution. Table 2 presents the average error between the different convergence results and the analytical solution. Although DMD-ODRO undergoes two model-reduction steps, there is no significant decrease in the accuracy of the model. In fact, the snapshot matrix used in the DMD and POD model-reduction steps is obtained during the CFD calculation process, and the reconstructed flow field snapshots are continuously updated and iterated back into the CFD calculations. Therefore, the accuracy of the acceleration method primarily depends on the CFD calculation process, and the model reduction does not introduce additional errors.

TABLE 3. Acceleration ratio of the 1D diffusion equation with different grid sizes.

	Iterations	Acceleration Ratio	CPU Time (s)	Acceleration Ratio	Memory Usage (KB)
Initial-100	15860	-	4	-	2.91
Initial-200	58420	-	19	-	6.33
Initial-400	224644	-	69	-	15.73
Initial-800	895545	-	313	-	34.48
DMD-ODRO-100	4006	3.96	1	4	7.01
DMD-ODRO-200	11256	5.19	4	4.75	14.33
DMD-ODRO-400	39008	5.76	15	4.6	31.57
DMD-ODRO-800	149547	5.99	62	5.05	65.94



(a) $\alpha = 0.05$

(b) $\alpha = 1$

(c) $\alpha = 10^4$

(d) $\alpha = 10^7$

FIGURE 3. Convergence results and error comparison of different methods under various diffusion coefficients α in the 1D diffusion equation. Left: convergence results; right: error comparison.

2) EXPERIMENT 2: NUMERICAL SIMULATIONS WITH A FIXED α AND DIFFERENT GRID SIZES

For this experiment, the diffusion coefficient α is fixed to 0.05. Flow field snapshots are saved every 50 iterations, and sets of five consecutive snapshots are used to construct a flow field matrix for model reduction. The CFL number is kept constant at 0.5. The average residual of the reduced-order models is calculated using 100, 200, 400, and 800 grid divisions to investigate the effect of the grid size on the acceleration method. The convergence history of the flow field residuals and the computation times for different grid sizes are shown in Fig. 4 and Table 3, respectively.

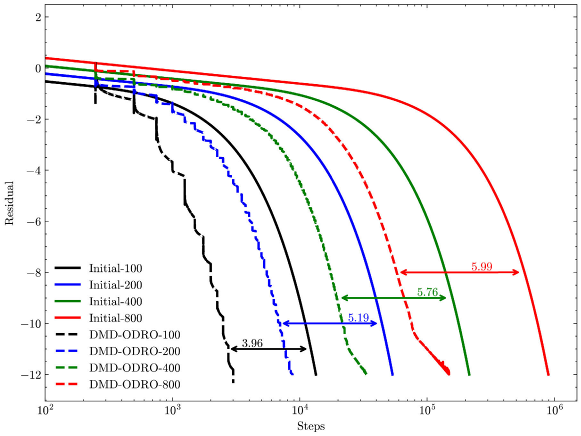


FIGURE 4. Convergence process of acceleration methods for the 1D diffusion equation with different grid sizes.

For the 1D diffusion equation, the DMD-ODRO method achieves a sixfold acceleration in convergence speed. This means that, compared with traditional methods, it significantly reduces the number of computational iterations under similar conditions. Moreover, the computation time also maintains a four to fivefold acceleration ratio. To provide a more comprehensive analysis of the computational cost and better demonstrate the model's performance, we also examined its spatial complexity. By measuring the actual memory usage during algorithm execution, we conducted quantitative experiments across different input scales. The results indicate that as the grid resolution increases, the input size of the model also grows, and the actual memory usage exhibits a linear relationship with the input size. Additionally, the DMD-ODRO model, which incorporates DMD and POD-based reduced-order methods, requires approximately

TABLE 4. Computation time for different methods under various diffusion coefficients *b* in the 1D advection–diffusion equation.

	$a=1,b=0.05$		$a=1,b=1$		$a=1,b=10$		$a=1,b=100$	
	Iterations	Time(s)	Iterations	Time(s)	Iterations	Time(s)	Iterations	Time(s)
Initial	32338	10	78902	25	102070	32	120802	39
DMD	14750	4	35750	10	46250	13	54750	16
ODRO	10750	3	22118	8	20551	8	23500	9
DMD-ODRO (ours)	5000	1	10480	4	12956	5	16000	6

two to three times more memory compared to traditional numerical methods. This is due to the need to process multiple sets of snapshot data during the dimensionality reduction process, leading to higher memory usage and increased spatial complexity. All of these results indicate that, even when handling more complex grids, the DMD-ODRO method utilizes computational resources more effectively to improve efficiency. As the number of grid points increases, ensuring the stability and accuracy of numerical solutions typically requires more complex numerical computations and additional iterative steps. Consequently, the number of iterations increases with the number of grid points. Simultaneously, the acceleration effect of the DMD-ODRO method improves as the grid becomes more refined, demonstrating the ability of the proposed method to effectively handle higher-resolution grids and expedite the convergence process.

B. ONE-DIMENSIONAL ADVECTION–DIFFUSION EQUATION

The 1D advection–diffusion equation is a fundamental partial differential equation. We consider this equation in the form:

$$\frac{\partial u}{\partial t}+a\frac{\partial u}{\partial x}=b\frac{\partial^2 u}{\partial x^2}, \tag{33}$$

where u represents the flow velocity, a denotes the advection coefficient, and b denotes the diffusion coefficient. The solution domain is defined as $x \in [-1, 1]$, with the boundary conditions and initial conditions specified as:

$$\begin{aligned} u(-1,t)=0.5,\, u(1,t)=-0.5 \\ u(x,0)=x. \end{aligned} \tag{34}$$

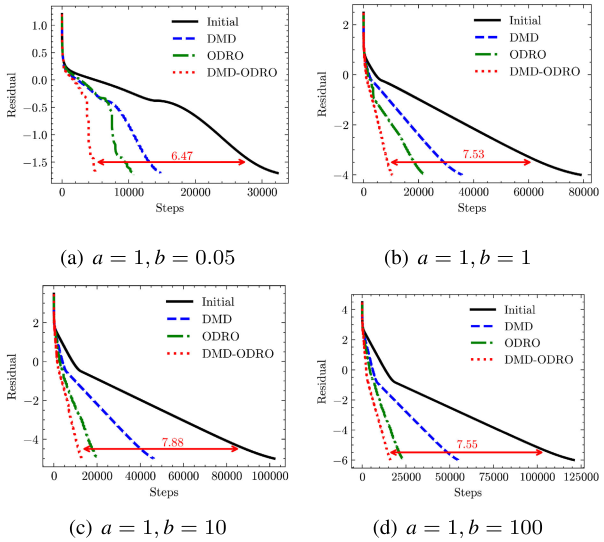
The CFL number is defined as:

$$\mathrm{CFL}=b\frac{\Delta t}{\Delta x^2}, \tag{35}$$

where Δt denotes the time step size and Δx represents the spatial step size.

1) EXPERIMENT 1: NUMERICAL SIMULATIONS WITH DIFFERENT DIFFUSION COEFFICIENTS *b* AND A FIXED GRID PARTITIONING

Using the same settings with 200 grid points and a CFL number of 0.5, snapshots of the flow field are saved every 50 iterations, and every set of five snapshots forms a flow field snapshot matrix for model reduction. The convergence histories of the flow field residuals under different methods and parameters are illustrated in Fig. 5, while the time statistics for various methods under different parameters are summarized in Table 4.



(a) $a=1,b=0.05$

(b) $a=1,b=1$

(c) $a=1,b=10$

(d) $a=1,b=100$

FIGURE 5. Convergence process of residual for different methods under various diffusion coefficients *b* in the 1D advection–diffusion equation.

As shown in Fig. 5 and Table 4, for the 1D advection–diffusion equation, the DMD-ODRO method exhibits superior acceleration effects. From the convergence curves, the DMD-ODRO method achieves an acceleration ratio of approximately 8×, and demonstrates a sixfold acceleration in computation time. As the diffusion coefficient increases and the Peclet number decreases, the flow transitions from advection-dominated to diffusion-dominated. Consequently, the convergence curves gradually become smoother. Increased diffusion enhances the smoothing of steep gradients or oscillations in the solution. Therefore, during the numerical solution process, increasing the diffusion coefficient typically leads to smoother solutions, reducing the residuals. This explains why increasing the diffusion coefficient results in progressively lower residual values after convergence.

As depicted in Fig. 6 and Table 5, when the diffusion coefficient corresponds to advection-dominated flow conditions (Fig. 5(a)), there are significant reductions in the gradient near boundary points during the convergence process.

TABLE 5. Average error of different methods under various diffusion coefficients b in the 1D advection–diffusion equation.

	Error ($b = 0.05$)	Error ($b = 1$)	Error ($b = 10$)	Error ($b = 100$)
Initial	7.7457×10^{-4}	4.9273×10^{-6}	9.1836×10^{-8}	9.3476×10^{-10}
DMD	7.4888×10^{-4}	5.1101×10^{-6}	9.5184×10^{-8}	9.6845×10^{-10}
ODRO	3.7690×10^{-4}	4.8225×10^{-6}	8.9101×10^{-8}	7.3911×10^{-10}
DMD-ODRO (ours)	2.6520×10^{-4}	4.8636×10^{-6}	9.0843×10^{-8}	8.5409×10^{-10}

TABLE 6. Acceleration ratio of the 1D advection–diffusion equation with different grid sizes.

	Iterations	Acceleration Ratio	CPU Time (s)	Acceleration Ratio
Initial-100	18728	-	5	-
Initial-200	78902	-	24	-
Initial-400	340168	-	112	-
Initial-800	1472944	-	646	-
DMD-ODRO-100	3250	5.76	1	5
DMD-ODRO-200	10480	7.53	4	6
DMD-ODRO-400	42110	8.08	16	7
DMD-ODRO-800	165250	8.91	71	9.10

Consequently, the errors between the numerical solution and the analytical solution are primarily concentrated in these regions. Under advection-dominated conditions, the DMD-ODRO method not only demonstrates the most effective acceleration of the convergence process, but also effectively suppresses the oscillations and fluctuations inherent in numerical solutions due to advection dominance, leading to smoother and more stable convergence curves. As the diffusion coefficient increases, the average errors of all methods decrease. This trend occurs because, at the same CFL number, increasing the diffusion coefficient enhances the spatial smoothness of solutions to the 1D advection–diffusion equation, thereby reducing the numerical oscillations and mitigating the occurrence of non-physical phenomena.

2) EXPERIMENT 2: NUMERICAL SIMULATIONS WITH A FIXED b AND DIFFERENT GRID SIZES

For this experiment, the advection coefficient a and diffusion coefficient b are both set to 1. Snapshots of the flow field are saved every 50 iterations, with every five snapshots forming a snapshot matrix for model reduction. The CFL number is maintained at 0.5. Different grid sizes of 100, 200, 400, and 800 are employed to compute the average residuals of the reduced-order models, allowing us to evaluate the impact of grid size on the acceleration method. The convergence trajectories of the residual values obtained are illustrated in Fig. 7, while the time statistics for different grid sizes are summarized in Table 6.

For the 1D advection–diffusion equation, the acceleration effects of each method gradually improve as the grid resolution increases. Compared with the 1D diffusion equation, the DMD-ODRO method exhibits more significant acceleration effects, achieving a maximum acceleration ratio of 8–9× and a computation time acceleration ratio of up to 9×. The advection term contains a first-order derivative, while the diffusion term contains a second-order derivative. Consequently, the diffusion term is more sensitive to spatial step lengths, requiring smaller time steps to maintain numerical stability. This leads to a sharp increase in computational cost. As the grid becomes more refined, the time step shrinks accordingly, yet the acceleration efficiency of the DMD-ODRO method continues to improve, indicating its effectiveness in accelerating convergence for such stiff problems.

C. ONE-DIMENSIONAL BURGERS' EQUATION

To test the acceleration effect of convergence methods in nonlinear flow fields, we now consider the Burgers' equation. This equation is often used to simulate the propagation and reflection of shock waves. Consider the 1D Burgers' equation in the form:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2}, \tag{36}$$

where u represents the flow velocity and ν denotes the diffusion coefficient. The solution domain is defined as $x \in [-1, 1]$, and the boundary conditions and initial conditions are specified as:

$$\begin{aligned} u(-1,t) = 0., u(1,t) = -0.5 \\ u(x,0) = -0.5x. \end{aligned} \tag{37}$$

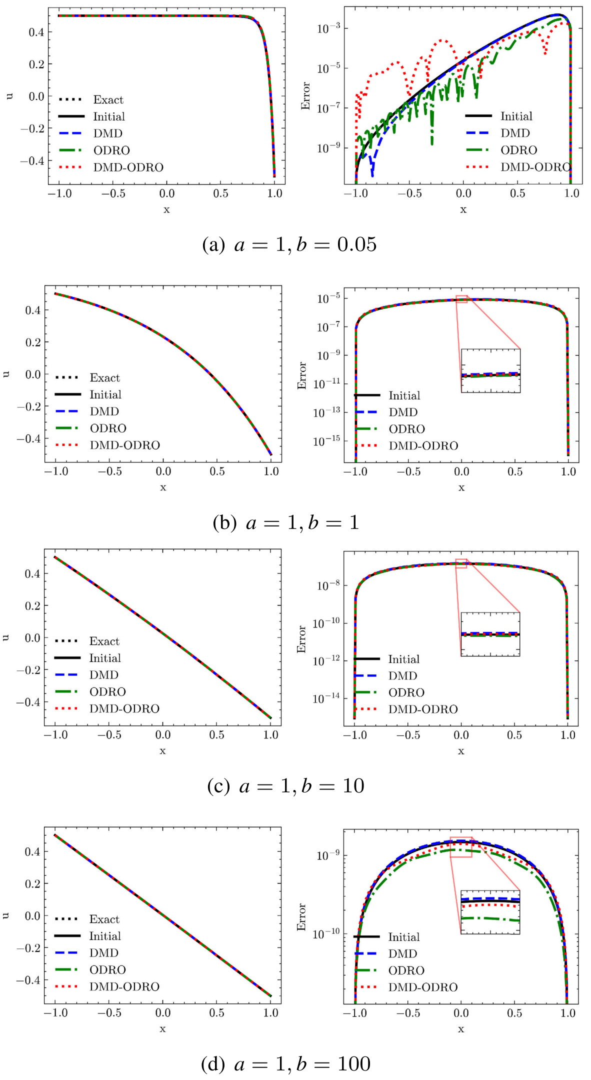
The CFL number is defined as:

$$\text{CFL} = \nu\frac{\Delta t}{\Delta x^2}, \tag{38}$$

where Δt denotes the time step size and Δx represents the spatial step size.

TABLE 7. Computation time for different methods under various diffusion coefficients ν in the 1D Burgers' equation.

	$\nu = 0.01$		$\nu = 0.05$		$\nu = 1$		$\nu = 10^4$	
	Iterations	Time(s)	Iterations	Time(s)	Iterations	Time(s)	Iterations	Time(s)
Initial	46364	20	22727	10	21834	9	19475	7
DMD	20429	8	10428	3	10000	3	8978	2
ODRO	16429	6	8569	4	8000	3	7014	2
DMD-ODRO (ours)	10500	4	4750	1	4750	1	4250	1



(a) $a = 1, b = 0.05$

(b) $a = 1, b = 1$

(c) $a = 1, b = 10$

(d) $a = 1, b = 100$

FIGURE 6. Convergence results and error comparison of different methods under various diffusion coefficients *b* in the 1D advection–diffusion equation. Left: convergence results; right: error comparison.

1) EXPERIMENT 1: NUMERICAL SIMULATIONS WITH DIFFERENT DIFFUSION COEFFICIENTS ν AND A FIXED GRID PARTITIONING

We consider a grid size of 200 with a CFL number of 0.5. Flow field snapshots are saved every 50 time step iterations, and every set of five snapshots forms a flow field snapshot matrix for model order reduction. An explicit time advancement method is used to compute the steady-state solution.

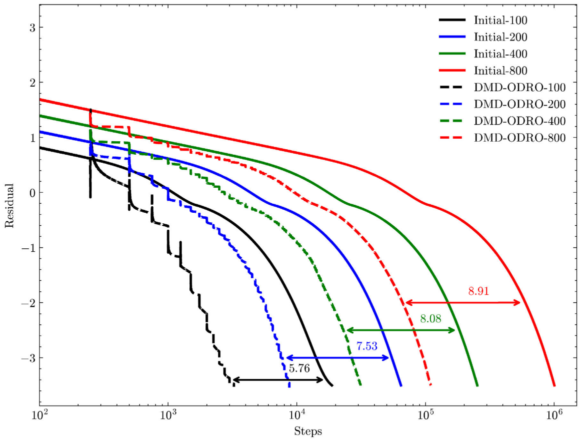
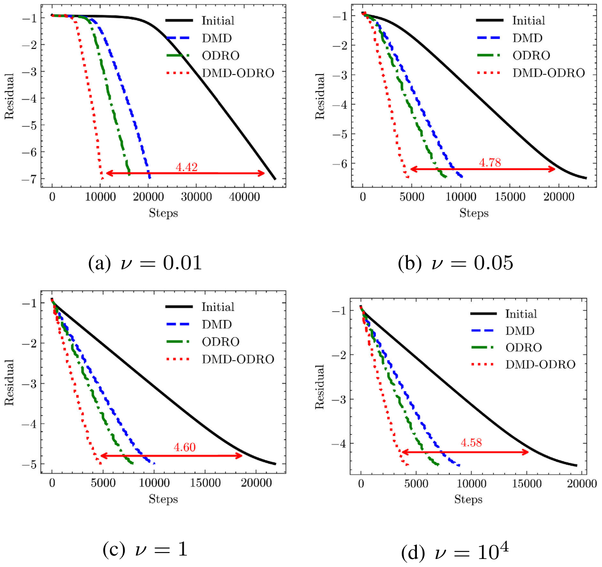


FIGURE 7. Convergence process of acceleration methods for the 1D advection–diffusion equation with different grid sizes.



(a) $\nu = 0.01$

(b) $\nu = 0.05$

(c) $\nu = 1$

(d) $\nu = 10^4$

FIGURE 8. Residual convergence process and convergence results of different methods under various diffusion coefficients ν in the 1D Burgers' equation. Left: residual convergence process; right: convergence results.

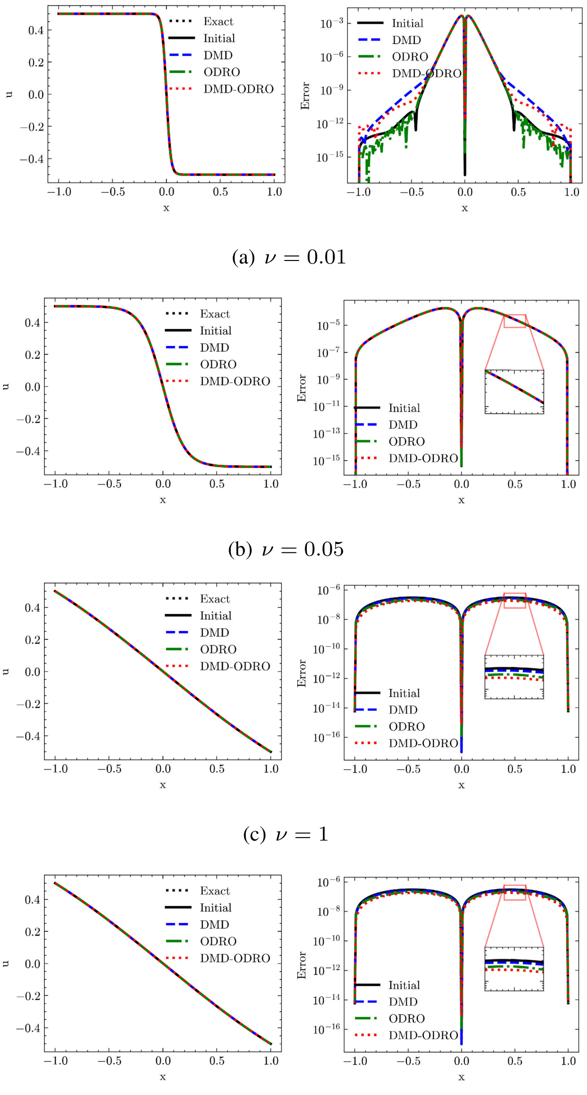
From Fig. 8 and Table 7, it is evident that accelerated convergence methods remain effective for nonlinear partial

TABLE 8. Average error of different methods under various diffusion coefficients ν in the 1D Burgers' equationv.

	Error ($\nu = 0.01$)	Error ($\nu = 0.05$)	Error ($\nu = 1$)	Error ($\nu = 10^4$)
Initial	2.9005×10^{-4}	5.7042×10^{-5}	1.8516×10^{-7}	6.8315×10^{-11}
DMD	2.9008×10^{-4}	5.7042×10^{-5}	1.6353×10^{-7}	6.8298×10^{-11}
ODRO	2.9005×10^{-4}	5.7044×10^{-5}	1.2824×10^{-7}	6.2077×10^{-11}
DMD-ODRO (ours)	2.9009×10^{-4}	5.7060×10^{-5}	1.0992×10^{-7}	4.8559×10^{-11}

differential equations. The DMD-ODRO method achieves a speedup of 4–5×. Additionally, the accelerated methods converge to identical results to those of the initial CFD methods. The convergence process and results vary with the diffusion coefficient. When the diffusion coefficient is small, the smoothing effect of the diffusion term is weak, leading to steeper velocity gradients and more pronounced fluctuations in the solution, as reflected in the steeper convergence curves. As the diffusion coefficient increases, the smoothing effect of the diffusion term strengthens, resulting in a smoother solution in space. Across different diffusion coefficients, the DMD-ODRO method consistently achieves the fastest convergence, maintaining a speedup of at least 4×. This is because the solution becomes smoother as the diffusion coefficient increases, resulting in smaller velocity gradient changes and fluctuations. During each reduction step, DMD extrapolation and online optimization of POD dimensionality reduction achieve lower residuals, thus accelerating the convergence.

An analysis of the errors of different numerical methods under various diffusion coefficients is presented, as shown in Fig. 9 and Table 8. When the diffusion coefficient is relatively small, the solution of the Burgers equation exhibits weak viscous effects, often leading to the formation of shock waves characterized by abrupt jumps in physical quantities. In this case, the gradient changes in local regions become exceptionally sharp, making it challenging to effectively smooth the shock waves and high-frequency oscillations in the solution. Due to the small diffusion coefficient, the smoothing effect of the viscous term in the equation is weakened, and the discretization error in numerical computations typically increases, failing to accurately capture the rapid variations in the solution. This results in significant numerical errors, as illustrated in Fig. 9(a). As the diffusion coefficient increases, the diffusion term in the equation becomes more prominent, effectively smoothing out sharp variations in the solution. This reduces the intensity of shock waves and gradually transforms the solution into a smoother transition region. In the visual representation, the previously existing shock waves are suppressed by the diffusion term and replaced by smoother transitions. At this point, the numerical solution's stability generally improves, approaching the true solution, and the discretization error decreases significantly. Additionally, with the increase in the diffusion coefficient, the DMD-ODRO method demonstrates a growing advantage in accuracy.



(a) $\nu = 0.01$

(b) $\nu = 0.05$

(c) $\nu = 1$

(d) $\nu = 10^4$

FIGURE 9. Convergence results and error comparison of different methods under various diffusion coefficients *b* in the 1D advection–diffusion equation. Left: convergence results; right: error comparison.

2) EXPERIMENT 2: NUMERICAL SIMULATIONS WITH A FIXED ν AND DIFFERENT GRID SIZES

With the diffusion coefficient ν fixed to 0.01, flow field snapshots are saved every 50 time step iterations, with every five snapshots forming a snapshot matrix for model order

TABLE 9. Acceleration ratio of the 1D Burgers' equation with different grid sizes.

	Iterations	Acceleration Ratio	CPU Time (s)	Acceleration Ratio
Initial-100	17335	-	6	-
Initial-200	69408	-	25	-
Initial-400	257958	-	89	-
Initial-800	1004143	-	441	-
DMD-ODRO-100	10345	1.68	3	2
DMD-ODRO-200	16395	4.23	6	4.17
DMD-ODRO-400	58750	4.39	21	4.24
DMD-ODRO-800	236379	4.25	100	4.41

reduction. A constant CFL number of 0.5 is maintained, and grid resolutions of 100, 200, 400, and 800 are utilized to compute the average residual of the reduced-order model, allowing us to investigate the impact of the grid resolution on the acceleration method. The computed flow field residual convergence history is depicted in Fig. 10, while the temporal metrics for the various grid resolutions are detailed in Table 9.

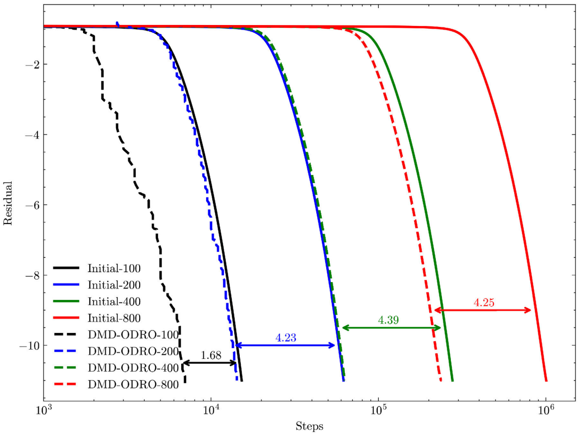


FIGURE 10. Convergence process of acceleration methods for the 1D Burgers' equation with different grid sizes.

As a nonlinear partial differential equation, the Burgers' equation is used to describe complex fluid phenomena such as shock waves and rarefaction waves. Its nonlinear convection term describes the influence of velocity gradients, with high-speed regions in the velocity field accelerating adjacent regions, leading to sharp gradients and wave phenomena. When solving the Burgers' equation numerically, low-resolution grids may fail to effectively capture rapid changes in the solution. As shown in Fig. 8, with a grid size of 100, the lower spatial resolution prevents the critical dynamic features of the nonlinear system from being accurately captured during DMD reduced-order extrapolation. This results in insufficient precision in the flow field predictions, necessitating additional iterations or corrections that increase the overall computation time. As the grid size gradually increases, enhancing the spatial resolution, the DMD-ODRO acceleration method becomes able to capture sharp changes and rapid fluctuations in the solution effectively. At this point, the acceleration effect is maintained at more than 4× the original speed.

D. FLOW AROUND A DUAL ELLIPSOID

The flow around a dual ellipsoid involves complex flow structures and is widely applied in aerodynamic shape design, flow control, and turbulence research. In this subsection, the DMD-ODRO acceleration method will be applied to the flow around a dual ellipsoid as a validation study. The experimental conditions are as follows: the incoming flow Mach number is $Ma = 10.02$, the unit Reynolds number is $Re = 1.13 \times 10^7/m$, the total pressure is $P = 7.8MPa$, and the total temperature is $T = 892K$. The angle of attack for the experiment is $\alpha = 20°$, and the flow regime is laminar. The experimental configuration and data are referenced from Li's book [49]. The computational grid is shown in Fig. 11, utilizing a hexahedral mesh with a total of 2.24 million cells. Local refinement is applied at the wall surface and the intersection of the dual ellipsoids, with the normal height of the first layer of the mesh set to $1.0\mathrm{e}^{-6}$ m. In the calculations, the spatial discretization uses the AUSM+UP scheme, and the time integration adopts the LUSGS method. Fig. 12 compares the residual convergence curves of the DMD-ODRO acceleration method and traditional numerical methods. It can be observed that for three-dimensional complex flows, the DMD-ODRO acceleration method achieves an acceleration ratio of approximately 6 times.

The numerical results are shown in Fig. 13, the heat flux results are nondimensionalized using the stagnation point heat flux of a spherical nose with a radius of 15 mm, denoted as q_{ref}, with a reference value of $568.4kW/m^2$. From the contour results on the symmetric plane of the flow field, it can be observed that when the angle of attack $\alpha = 20°$, the flow around the dual ellipsoid generates a leading-edge shock wave, which is significantly inclined. On the windward side, the shock wave is closer to the surface, resulting in a strong compression effect. While on the leeward side, the shock wave moves farther from the surface, leading to a pressure drop. In this scenario, the windward shock wave adheres closely to the ellipsoid surface, causing a significant increase in surface heat flux density in this region. Fig. 14 shows the

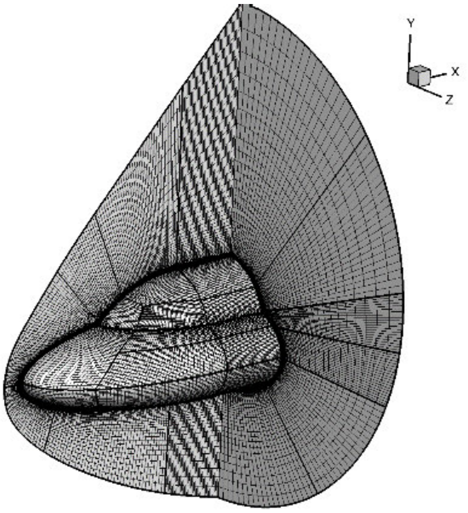


FIGURE 11. Computational grid for the dual ellipsoid model.

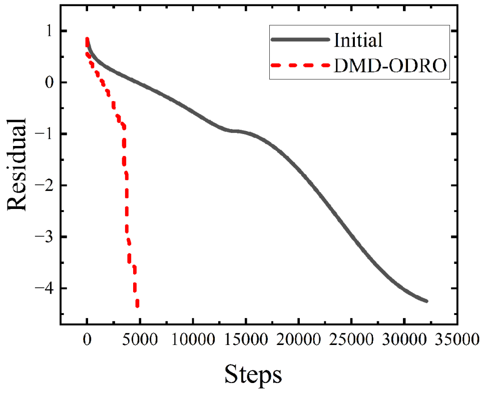
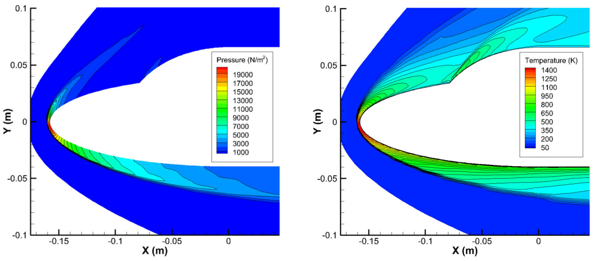


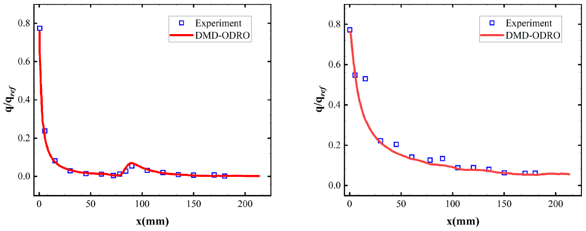
FIGURE 12. Residual convergence curve comparison.



(a) Pressure (b) Temperature

FIGURE 13. Contour plot of the flow field on the symmetric plane of the dual ellipsoid.

heat flux distribution on the central symmetry plane of the dual ellipsoid. From the heat flux distribution on the upper and lower surfaces of the dual ellipsoid, it can be seen that the numerical results agree well with the experimental heat transfer data for the dual ellipsoid model presented in the referenced literature [49], reflecting the true variations in the flow field.



(a) Upper surface (b) Lower surface

FIGURE 14. Heat flux distribution on the central symmetry plane of the dual ellipsoid.

IV. CONCLUSION

This paper has proposed an accelerated convergence method based on the combination of DMD reduced-order extrapolation and POD ODRO. The approach first builds a reduced-order model using DMD to filter the high-frequency components of flow snapshots. The most energetic modes of these new snapshots are extracted using POD. Within the low-dimensional POD mode space, a gradient-free optimization algorithm is employed to minimize the residual of the reduced-order model. The resulting lower-residual flow fields are taken as the initial conditions for iterative updates with the CFD solver, thus achieving accelerated convergence. This method effectively eliminates the influence of high-frequency oscillations in flow components. Furthermore, the proposed method addresses the slow convergence and residual jump issues caused by the system's inherent rigidity characteristics. These two advantages render the proposed method highly efficient in flow simulations. Moreover, the model-reduction extrapolation in this method primarily involves simple algebraic operations on flow field snapshots, avoiding complex interactions between different grids. Thus, it can be easily implemented as an external module to the CFD solver without modifying the solver's source code, using only the flow field snapshots saved during the iteration process.

The performance of the proposed method has been validated through three model equations and one high-dimensional complex example. demonstrating its robustness and efficiency in achieving machine-zero convergence solutions with minimal iteration steps and CPU computation times, regardless of whether the flow is convection-dominated or diffusion-dominated. Furthermore, under varying experimental parameters and grid densities, the proposed method consistently achieves notable acceleration effects. For nonlinear unstable flows, the method effectively filters out high-frequency unstable flow modes while mitigating the residual jumps and convergence issues arising from the linear system assumptions inherent in reduced-order models due to their stiffness characteristics, thereby further enhancing the convergence efficiency.

The examples presented in this paper demonstrate the effectiveness and robustness of the proposed method. Despite the relative simplicity of the selected numerical test cases, the experience and insights gained from this work shed light on the future development of accelerated convergence methods for steady-state flow fields. In subsequent work, this method will be attempt to optimize the selection strategy for reduced-order bases and the adaptive snapshot acquisition process, thereby enhancing the predictive accuracy and expanding the range of applicability of the reduced-order models. Additionally, the potential of the proposed approach for accelerating transient flow field solutions warrants further research and testing, especially for complex flows, such as transonic flow over aircraft [50], [51]. Further exploration will focus on addressing nonlinear problems, such as the transonic buffet flow over aircraft airfoils, to improve the method's adaptability to highly complex flow phenomena.

REFERENCES

[1] S. Zeng, Z. Yuan, W. Zhao, and W. Chen, "Numerical simulation of hypersonic thermochemical nonequilibrium flows using nonlinear coupled constitutive relations," *Chin. J. Aeronaut.*, vol. 36, no. 3, pp. 63–79, Mar. 2023.

[2] C. Zhang, Z. Wan, and D. Sun, "Model reduction for supersonic cavity flow using proper orthogonal decomposition (POD) and Galerkin projection," *Appl. Math. Mech.*, vol. 38, no. 5, pp. 723–736, May 2017.

[3] S. Seraj, A. Yildirim, J. L. Anibal, and J. R. R. A. Martins, "Dissipation and time step scaling strategies for low and high Mach number flows," *J. Comput. Phys.*, vol. 491, Oct. 2023, Art. no. 112358.

[4] H. Tang, J. Rabault, A. Kuhnle, Y. Wang, and T. Wang, "Robust active flow control over a range of Reynolds numbers using an artificial neural network trained through deep reinforcement learning," *Phys. Fluids*, vol. 32, no. 5, p. 53605, May 2020.

[5] Y. Li, J. Chang, C. Kong, and W. Bao, "Recent progress of machine learning in flow modeling and active flow control," *Chin. J. Aeronaut.*, vol. 35, no. 4, pp. 14–44, Apr. 2022.

[6] L. Tian, P. Xiao, Y. Song, N. Zhao, C. Zhu, and X. Lu, "An advanced three-dimensional analytical model for wind turbine near and far wake predictions," *Renew. Energy*, vol. 223, Mar. 2024, Art. no. 120035.

[7] F. Alauzet and L. Frazza, "Feature-based and goal-oriented anisotropic mesh adaptation for RANS applications in aeronautics and aerospace," *J. Comput. Phys.*, vol. 439, Aug. 2021, Art. no. 110340.

[8] D. J. Mavriplis, "Progress in computational fluid dynamics discretizations algorithms and solvers for aerodynamic flows," *AIAA J.*, vol. 59, no. 12, pp. 5374–5397, Dec. 2021.

[9] J. C. Butcher, "A history of Runge–Kutta methods," *Appl. Numer. Math.*, vol. 20, no. 3, pp. 247–260, Mar. 1996.

[10] A. Reusken, "Convergence analysis of the Gauss–Seidel preconditioner for discretized one dimensional Euler equations," *SIAM J. Numer. Anal.*, vol. 41, no. 4, pp. 1388–1405, Jan. 2003.

[11] M. Wang and L. Chen, "Multigrid methods for the Stokes equations using distributive Gauss–Seidel relaxations based on the least squares commutator," *J. Scientific Comput.*, vol. 56, no. 2, pp. 409–431, Aug. 2013.

[12] A. Ahmadi, F. Manganiello, A. Khademi, and M. C. Smith, "A parallel jacobi-embedded Gauss–Seidel method," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 6, pp. 1452–1464, Jun. 2021.

[13] S. Thomas, E. Carson, M. Rozložník, A. Carr, and K. Świrydowicz, "Iterated Gauss–Seidel GMRES," *SIAM J. Scientific Comput.*, vol. 46, no. 2, pp. S254–S279, Apr. 2024.

[14] A. Jameson and S. Yoon, "Lower-upper implicit schemes with multiple grids for the Euler equations," *AIAA J.*, vol. 25, no. 7, pp. 929–935, Jul. 1987.

[15] Z. X. M. Z. X. Meng, S. Z. L. S. Z. Li, K. P. K. Peng, and W. H. Z. W. H. Zhang, "An implicit block LU-SGS algorithm-based lattice Boltzmann flux solver for simulation of hypersonic flows," *Adv. Appl. Math. Mech.*, vol. 11, no. 1, pp. 72–90, Jan. 2019.

[16] M. Moreira Lopes, M. O. Domingues, K. Schneider, and O. Mendes, "Local time-stepping for adaptive multiresolution using natural extension of Runge–Kutta methods," *J. Comput. Phys.*, vol. 382, pp. 291–318, Apr. 2019.

[17] W. Throwe and S. Teukolsky, "A high-order, conservative integrator with local time-stepping," *SIAM J. Scientific Comput.*, vol. 42, no. 6, pp. A3730–A3760, Jan. 2020.

[18] D. J. Mavriplis, "A residual smoothing strategy for accelerating Newton method continuation," *Comput. Fluids*, vol. 220, Apr. 2021, Art. no. 104859.

[19] A. Jameson, "Solution of the Euler equations for two dimensional transonic flow by a multigrid method," *Appl. Math. Comput.*, vol. 13, nos. 3–4, pp. 327–355, Jan. 1983.

[20] R. Markovinović and J. D. Jansen, "Accelerating iterative solution methods using reduced-order models as solution predictors," *Int. J. Numer. Methods Eng.*, vol. 68, no. 5, pp. 525–541, Oct. 2006.

[21] R. Djeddi, A. Kaminsky, and K. Ekici, "Convergence acceleration of fluid dynamics solvers using a reduced-order model," *AIAA J.*, vol. 55, no. 9, pp. 3059–3071, Sep. 2017.

[22] Z. Li and P. He, "Accelerating unsteady aerodynamic simulations using predictive reduced-order modeling," *Aerosp. Sci. Technol.*, vol. 139, Aug. 2023, Art. no. 108412.

[23] M. Mešina, "Convergence acceleration for the iterative solution of the equations $X = AX + f$," *Comput. Methods Appl. Mech. Eng.*, vol. 10, no. 2, pp. 165–173, Feb. 1977.

[24] A. Sidi, "Review of two vector extrapolation methods of polynomial type with applications to large-scale problems," *J. Comput. Sci.*, vol. 3, no. 3, pp. 92–101, May 2012.

[25] Z. Luo, X. Liu, Y. Zeng, and Y. Li, "Reduced-order extrapolation technique for coefficient vectors of numerical solutions about fluid mechanics equation," *Numer. Methods Partial Differ. Equ.*, vol. 39, no. 4, pp. 2823–2839, Jul. 2023.

[26] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *J. Fluid Mech.*, vol. 656, pp. 5–28, Aug. 2010.

[27] P. J. Schmid, "Dynamic mode decomposition and its variants," *Annu. Rev. Fluid Mech.*, vol. 54, no. 1, pp. 225–254, Jan. 2022.

[28] Q. A. Huhn, M. E. Tano, J. C. Ragusa, and Y. Choi, "Parametric dynamic mode decomposition for reduced order modeling," *J. Comput. Phys.*, vol. 475, Feb. 2023, Art. no. 111852.

[29] B. García-Archilla, V. John, and J. Novo, "POD-ROMs for incompressible flows including snapshots of the temporal derivative of the full order solution," *SIAM J. Numer. Anal.*, vol. 61, no. 3, pp. 1340–1368, Jun. 2023.

[30] K. Taira, S. L. Brunton, S. T. M. Dawson, C. W. Rowley, T. Colonius, B. J. McKeon, O. T. Schmidt, S. Gordeyev, V. Theofilis, and L. S. Ukeiley, "Modal analysis of fluid flows: An overview," *AIAA J.*, vol. 55, no. 12, pp. 4013–4041, Dec. 2017.

[31] J.-L. Li, Y. Zhang, B.-B. Zhu, B. Pang, and G. Chen, "The aerodynamic optimization of hypersonic vehicles with the proper-orthogonal-decomposition-based CST method," *Aerosp. Sci. Technol.*, vol. 151, Aug. 2024, Art. no. 109295.

[32] N. Andersson, "A non-intrusive acceleration technique for compressible flow solvers based on dynamic mode decomposition," *Comput. Fluids*, vol. 133, pp. 32–42, Jul. 2016.

[33] Y. Liu, G. Wang, and Z. Ye, "Dynamic mode extrapolation to improve the efficiency of dual time stepping method," *J. Comput. Phys.*, vol. 352, pp. 190–212, Jan. 2018.

[34] Y. Liu, W. Zhang, and J. Kou, "Mode multigrid–A novel convergence acceleration method," *Aerosp. Sci. Technol.*, vol. 92, pp. 605–619, Sep. 2019.

[35] S. Tan, Y. Liu, J. Kou, and W. Zhang, "Improved mode multigrid method for accelerating turbulence flows," *AIAA J.*, vol. 59, no. 8, pp. 1–13, Aug. 2021.

[36] X. Wang, Y. Liu, and W. Zhang, "Novel approach to improve stability and convergence of flowfield solution processes: Mode multigrid," *AIAA J.*, vol. 61, no. 8, pp. 3496–3506, Aug. 2023.

[37] K. Eriksson, C. Johnson, and A. Logg, "Explicit time-stepping for stiff ODEs," *SIAM J. Scientific Comput.*, vol. 25, no. 4, pp. 1142–1157, Jan. 2004.

[38] A. L. Kaminsky, R. Djeddi, and K. Ekici, "Convergence acceleration of continuous adjoint solvers using a reduced-order model," *Int. J. Numer. Methods Fluids*, vol. 86, no. 9, pp. 582–606, Mar. 2018.

[39] W. Cao, Y. Liu, X. Shan, C. Gao, and W. Zhang, "A novel convergence enhancement method based on online dimension reduction optimization," *Phys. Fluids*, vol. 35, no. 3, p. 36124, Mar. 2023.

[40] S. Le Clainche, F. Varas, and J. M. Vega, "Accelerating oil reservoir simulations using POD on the fly," *Int. J. Numer. Methods Eng.*, vol. 110, no. 1, pp. 79–100, Apr. 2017.

[41] T. Li, D. Han, B. Yu, J. Li, and D. Sun, "Study on a POD reduced-order model for steady-state flows in fractured porous media," *Int. Commun. Heat Mass Transf.*, vol. 112, Mar. 2020, Art. no. 104489.

[42] M. L. Rapún, F. Terragni, and J. M. Vega, "Adaptive POD-based low-dimensional modeling supported by residual estimates," *Int. J. Numer. Methods Eng.*, vol. 104, no. 9, pp. 844–868, Nov. 2015.

[43] Z. Ba and Y. Wang, "Numerical analysis of transient state heat transfer by spectral method based on POD reduced-order extrapolation algorithm," *Appl. Sci.*, vol. 13, no. 11, p. 6665, May 2023.

[44] G. Wanner and E. Hairer, *Solving Ordinary Differential Equations II*, vol. 375. Berlin, Germany: Springer, 1996.

[45] J. Blazek, *Computational Fluid Dynamics: Principles and Applications*. London, U.K.: Butterworth, 2015.

[46] G. Berkooz, "The proper orthogonal decomposition in the analysis of turbulent flows," *Annu. Rev. Fluid Mech.*, vol. 25, no. 1, pp. 539–575, Jan. 1993.

[47] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the nelder-mead simplex method in low dimensions," *SIAM J. Optim.*, vol. 9, no. 1, pp. 112–147, Jan. 1998.

[48] Z.-D. Luo, Q.-L. Ou, and Z.-H. Xie, "Reduced finite difference scheme and error estimates based on POD method for non-stationary Stokes equation," *Appl. Math. Mech.*, vol. 32, no. 7, pp. 847–858, Jul. 2011.

[49] S. Li, *Typical Characteristics of Hypersonic Flow Around Aerodynamic Shapes*. Beijing, China: Nat. Defense Ind. Press, 2008.

[50] T. Xiao, H. Zhi, S. Deng, Z. Chen, and X. Li, "Enhancement on parallel unstructured overset grid method for complex aerospace engineering applications," *Chin. J. Aeronaut.*, vol. 36, no. 1, pp. 115–138, Jan. 2023.

[51] F. Lyu, T. Xiao, and X. Yu, "A fast and automatic full-potential finite volume solver on Cartesian grids for unconventional configurations," *Chin. J. Aeronaut.*, vol. 30, no. 3, pp. 951–963, Jun. 2017.

JIANHUI LI received the B.S. degree in oil and gas storage and transportation engineering from the Civil Aviation University of China, in 2021. He is currently pursuing the M.S. degree in mechanical engineering with the School of Manufacturing Science and Engineering, Southwest University of Science and Technology, Mianyang, China. His research interests include intelligent fluid dynamics, computational fluid dynamics, and machine learning.

JUN HUANG received the M.S. degree in computer science from the Southwest University of Science and Technology, Mianyang, China, in 2014. He is currently pursuing the Ph.D. degree with the Graduate School, China Academy of Engineering Physics, Mianyang. He is also working as a Teaching Assistant with the School of Computer Science and Technology, Southwest University of Science and Technology. His current research interests include optimal control, big data, scientific computing, and high-performance computing.

YAHUI SUN received the B.S. degree in network engineering from Zaozhuang University of China, in 2022. He is currently pursuing the M.S. degree in computer science with the School of Computer Science and Technology, Southwest University of Science and Technology, Mianyang, China. His research interests include intelligent fluid dynamics, big data, and machine learning.

GUOQING LI received the Ph.D. degree from the Department of Precision Machinery and Precision Instruments, University of Science and Technology of China. He is currently working as a Professor with the School of Manufacturing Science and Engineering, Southwest University of Science and Technology, Mianyang, China. His research interests include micro-nano functional structures, high-precision 3D printing, and integrated systems.