

An Implicit High-Order Preconditioned Flux Reconstruction Method for Low-Mach-Number Flow Simulation with Dynamic Meshes

Lai Wang* and Meilin Yu†

Department of Mechanical Engineering,
University of Maryland, Baltimore County, Baltimore, MD 21250

Abstract

A fully implicit high-order preconditioned flux reconstruction/correction procedure via reconstruction (FR/CPR) method is developed to solve the compressible Navier–Stokes equations at low Mach numbers. A dual-time stepping approach with the second-order backward differentiation formula (BDF2) is employed to ensure temporal accuracy for unsteady flow simulation. When dynamic meshes are used to handle moving/deforming domains, the geometric conservation law (GCL) is implicitly enforced to eliminate errors due to the resolution discrepancy between BDF2 and the spatial FR/CPR discretization. The large linear system resulting from the spatial and temporal discretizations is tackled with the restarted Generalized Minimal Residual (GMRES) solver in the PETSc (Portable, Extensible Toolkit for Scientific Computation) library. Through several benchmark steady and unsteady numerical tests, the preconditioned FR/CPR methods have demonstrated good convergence and accuracy for simulating flows at low Mach numbers. The new flow solver is then used to study the effects of Mach number on unsteady force generation over a plunging airfoil when operating in low-Mach-number flows. It is observed that weak compressibility has a significant impact on thrust generation but a negligible effect on lift generation of an oscillating airfoil.

Key Words

Flux Reconstruction/Correction Procedure via Reconstruction; Low-Mach-Number Preconditioning; High-Order Methods; Dual-Time Stepping; Unsteady Flows; Dynamic Meshes; Weak Compressibility

*Graduate student. Email: bx58858@umbc.edu

†Assistant Professor. Corresponding Author. Email: mlyu@umbc.edu

1 Introduction

Analysis of unsteady fluid flows at low Mach numbers is an indispensable component in many mechanical engineering applications, such as design and control of wind turbines, micro air vehicles, autonomous underwater vehicles, and medical devices, just to name a few. There are typically two approaches to handle low-Mach-number flows. In the first one, fluids are considered incompressible, and their motion is governed by the incompressible Navier–Stokes equations. The pressure-correction method (e.g., SIMPLE (Semi-Implicit Method for Pressure Linked Equations) families [1], and the projection method [2]), and artificial compressibility method [3, 4, 5, 6, 7, 8] are among the most popular approaches for solving incompressible flows. An alternative is to directly solve the compressible Navier–Stokes equations at low Mach numbers. In this study, we pursue the second approach to solve low-Mach-number flows. We note that the compressible flow equations become stiff to solve numerically at low Mach numbers due to the large disparity between the flow (or entropy characteristic) velocity and the speed of sound. This disparity not only degrades the convergence of numerical solvers for the compressible Navier–Stokes equations, but also results in loss of numerical accuracy [9].

Various local preconditioning methods have been proposed to decrease the stiffness of the compressible Navier–Stokes equations at low Mach numbers. In general, after local preconditioning, the physical acoustic wave speeds are modified to be of the same order of magnitude as the flow speeds. Thus, the convergence rate of the flow solver can be greatly improved. Three major groups of preconditioning methods have been reported in the literatures [10]. Inspired by the artificial compressibility method developed by Chorin [11], Turkel adopted (p, \mathbf{v}, dS) , where p is the pressure, \mathbf{v} is the velocity vector, and dS is the change of entropy, as working variables in a series of work [9, 12, 13, 14]. Choi and Merkle [15], and Weiss and Smith [16] employed (p, \mathbf{v}, T) as working variables. Herein, T is the temperature. Van Leer *et al.* [17, 18] developed a symmetric Van Leer–Lee–Roe preconditioner using $(dp/(\rho c), d\mathbf{v}, dS)$ as working variables, where ρ is the fluid density, and c is the speed of sound. We note that different sets of working variables have been assessed by Turkel *et al.* [13]. Hauke and Hughes in Ref. [19] pointed out that the conservative incompressible formulation is well defined only for the entropy variables and the primitive variables. As reported in Refs. [15, 16, 20, 21, 22], direct approximations of the gradients of primitive variables can be more accurate than those computed from gradients of conservative variables. Therefore, in this study, the primitive variables are selected as working variables, and the preconditioning approach by Weiss and Smith [16] is adopted.

Contributions. Preconditioned high-order methods have been employed to solve the compressible Navier–Stokes equations at low Mach numbers in

past decades. For example, the discontinuous Galerkin (DG) method has been developed to solve low-Mach-number flows in Refs. [23, 24, 22, 25, 26]. In this paper, an implicit preconditioned FR/CPR method is developed for the first time to solve the unsteady compressible Navier–Stokes equations at low Mach numbers involving dynamic meshes. The dual-time stepping method with BDF2 is employed to achieve second-order accuracy in time for unsteady flow simulations. Within the framework of dual-time stepping, the GCL is enforced analytically following the approach proposed in Refs. [27, 28]. In a previous work [29], we have compared the numerical performance of the preconditioned FR/CPR formulation with the incompressible FR/CPR method with artificial compressibility [8]. In this study, we conduct thorough numerical experiments to test accuracy, convergence and efficiency of the preconditioned FR/CPR formulation for solving steady and unsteady flow problems.

Article overview. The remainder of the paper is organized as follows. In 2, the non-dimensionalized compressible Navier–Stokes equations are presented first (2.1). The preconditioning methodology for steady flow problems is then explained in 2.2. After that, preconditioning for unsteady flow problems on moving grids with implicit GCL enforcement is elaborated in 2.3. Numerical methods are introduced in 3. A brief review of the FR/CPR method is given in 3.1, and 3.2 formulates the concrete discrete forms of time marching methods. Boundary conditions for preconditioned systems are then discussed in 3.3. In 4, numerical results from both steady and unsteady simulations using the preconditioned FR/CPR methods are presented. The present study is summarized in 5.

2 Low-Mach-Number Preconditioning

2.1 Governing Equations

The non-dimensionalized compressible Navier–Stokes equations can be written as

$$\frac{\partial \mathbf{q}^c}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = 0, \quad (1)$$

with the following non-dimensionalizations [30],

$$\begin{cases} t = \bar{t}/(L_{ref}/u_{ref}), & x = \bar{x}/L_{ref}, \\ \rho = \bar{\rho}/\rho_{ref}, & \mathbf{v} = \bar{\mathbf{v}}/u_{ref}, \quad p = \bar{p}/\rho_{ref}u_{ref}^2, \quad T = \bar{T}/T_{ref}, \\ \mu = \bar{\mu}/\mu_{ref}, & R = 1/(\gamma Ma_{ref}^2), \quad C_p = 1/((\gamma - 1)Ma_{ref}^2). \end{cases} \quad (2)$$

Herein, $(\bar{*})$ is the variable in the physical domain, $(*_{ref})$ is the reference variable, $\mathbf{q}^c = (\rho, \rho u, \rho v, E)^\top$ are the non-dimensionalized conservative variables, ρ is the non-dimensionalized density, $\mathbf{v} = (u, v)^\top$ is the non-dimensionalized

velocity vector, $p = \rho RT$ is the non-dimensionalized pressure, T is the non-dimensionalized temperature, $E = \frac{p}{\gamma-1} + \frac{1}{2}\rho(u^2 + v^2)$ is the non-dimensionalized total energy, the heat capacity ratio $\gamma = 1.4$, and Ma is the Mach number. The fluxes \mathbf{f} and \mathbf{g} consist of inviscid and viscous parts, i.e., $\mathbf{f} = \mathbf{f}_{inv} - \mathbf{f}_{vis}$ and $\mathbf{g} = \mathbf{g}_{inv} - \mathbf{g}_{vis}$. The components in \mathbf{f}_{inv} , \mathbf{g}_{inv} , \mathbf{f}_{vis} , \mathbf{g}_{vis} can be found below,

$$\mathbf{f}_{inv} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (E + p)u \end{pmatrix}, \mathbf{g}_{inv} = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (E + p)v \end{pmatrix}, \quad (3)$$

$$\mathbf{f}_v = \frac{1}{Re_{ref}} \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ u\tau_{xx} + v\tau_{xy} + k\frac{\partial T}{\partial x} \end{pmatrix}, \mathbf{g}_v = \frac{1}{Re_{ref}} \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ u\tau_{xy} + v\tau_{yy} + k\frac{\partial T}{\partial y} \end{pmatrix}. \quad (4)$$

Herein, the non-dimensionalized thermal conductivity $k = \frac{\mu C_p}{Pr}$, the Prandtl number $Pr = 0.72$ and μ is the non-dimensionalized dynamic viscosity, which is treated as a constant in this work. $Re_{ref} = \rho_{ref} u_{ref} L_{ref} / \mu_{ref}$ is the Reynolds number of the reference flow at far fields. Non-dimensionalized viscous stresses are given by

$$\begin{cases} \tau_{xx} = \mu \left(\frac{4}{3} \frac{\partial u}{\partial x} - \frac{2}{3} \frac{\partial v}{\partial y} \right), \\ \tau_{yy} = \mu \left(\frac{4}{3} \frac{\partial v}{\partial y} - \frac{2}{3} \frac{\partial u}{\partial x} \right), \\ \tau_{xy} = \tau_{yx} = \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right). \end{cases} \quad (5)$$

At low-Mach-number regions, the eigenvalues of the convection part of Eq. (1) are significantly different from each other. This disparity makes the hyperbolic system stiff to solve. Preconditioning methods replace the characteristic wave speeds with modified ones to decrease the stiffness of the hyperbolic system.

2.2 Preconditioning for Steady Flow Problems

A pseudo-transient continuation approach is used to handle steady flow problems. In this approach, on applying the chain-rule, Eq. (1) can be written as

$$\frac{\partial \mathbf{q}^c}{\partial \mathbf{q}^p} \frac{\partial \mathbf{q}^p}{\partial \tau} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = 0, \quad (6)$$

where τ is the pseudo-time, $\mathbf{q}^p = (p, u, v, T)^\top$ are the non-dimensionalized primitive variables, $\mathbf{M} = \frac{\partial \mathbf{q}^c}{\partial \mathbf{q}^p}$ is the Jacobian matrix of the two sets of variables, i.e., the conservative and primitive variables, which can be expressed

as

$$\mathbf{M} = \frac{\partial \mathbf{q}^c}{\partial \mathbf{q}^p} = \begin{pmatrix} \rho_p & 0 & 0 & \rho_T \\ \rho_p u & \rho & 0 & \rho_T u \\ \rho_p v & 0 & \rho & \rho_T v \\ \rho_p h - 1 & \rho u & \rho v & \rho_T h + \rho C_p \end{pmatrix}. \quad (7)$$

Herein, h is the non-dimensionalized specific total enthalpy defined as $h = C_p T + \frac{1}{2}(u^2 + v^2)$.

Recall that the state equation $p = \rho R T$. Thus the derivatives in Eq. (7) can be calculated as [22]

$$\rho_p = \left. \frac{\partial \rho}{\partial p} \right|_T = \frac{1}{RT}, \rho_T = \left. \frac{\partial \rho}{\partial T} \right|_p = -\frac{\rho}{T}. \quad (8)$$

In the preconditioning method adopted here, \mathbf{M} is replaced with a matrix $\mathbf{\Gamma}$ called preconditioning matrix, which can cluster the eigenvalues of the hyperbolic system. The preconditioning matrix is given by

$$\mathbf{\Gamma} = \begin{pmatrix} \Theta & 0 & 0 & \rho_T \\ \Theta u & \rho & 0 & \rho_T u \\ \Theta v & 0 & \rho & \rho_T v \\ \Theta h - 1 & \rho u & \rho v & \rho_T h + \rho C_p \end{pmatrix}. \quad (9)$$

Note that $\rho_p = \frac{\partial \rho}{\partial p}$ in \mathbf{M} is substituted with Θ in the preconditioning matrix $\mathbf{\Gamma}$. The definition of Θ follows that in Ref. [16]

$$\Theta = \left(\frac{1}{U_r^2} - \frac{\rho_T}{\rho C_p} \right), \quad (10)$$

where $U_r = \epsilon c$, c is the local speed of sound, and ϵ is a free parameter related to the local Mach number Ma , and the global cut-off Mach number which is usually chosen as the free stream Mach number Ma_∞ .

After introducing the preconditioning matrix into the compressible Navier-Stokes equations, the governing equations can be re-expressed as

$$\mathbf{\Gamma} \frac{\partial \mathbf{q}^p}{\partial \tau} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = 0. \quad (11)$$

The preconditioned Jacobian matrix related to inviscid fluxes in the normal direction of any surface becomes $\mathbf{\Gamma}^{-1} \mathbf{A} \mathbf{n}$, where $\mathbf{A} \mathbf{n} = n_x \frac{\partial \mathbf{f}_{inv}}{\partial q^p} + n_y \frac{\partial \mathbf{g}_{inv}}{\partial q^p}$, and $\mathbf{n} = (n_x, n_y)^\top$ is the unit normal vector of the surface. The eigenvalues of this system are

$$\begin{cases} \lambda_1 = \lambda_2 = u_n \\ \lambda_3 = u'_n + c' \\ \lambda_4 = u'_n - c', \end{cases} \quad (12)$$

where

$$\begin{cases} u_n = \mathbf{v} \cdot \mathbf{n} \\ u'_n = u_n(1 - \alpha) \\ c' = \sqrt{\alpha^2 u_n^2 + U_r^2} \\ \alpha = (1 - \beta U_r^2)/2 \\ \beta = \left(\rho_p + \frac{\rho_T}{\rho C_p} \right). \end{cases} \quad (13)$$

For an ideal gas, $\beta = (\gamma RT)^{-1} = 1/c^2$. At low speed, when U_r approaches zero, α will approach $\frac{1}{2}$. All the eigenvalues will then have the same magnitude as u_n . Thus, the stiffness of the compressible Navier–Stokes equations is significantly decreased.

In the present study, when no moving grids are involved, the parameter ϵ in $U_r = \epsilon c$ is defined as

$$\epsilon = \min(1, \max(\kappa Ma_\infty, Ma)) \quad (14)$$

where κ is a free parameter. Typically, $\kappa = 1$. Turkel suggested that κ can be up to $\sqrt{3} \sim \sqrt{5}$ in Ref. [14]. The global cut-off parameter κMa_∞ [20, 14] is employed to prevent robustness deterioration instabilities near stagnation points.

2.3 Preconditioning for Unsteady Flow Problems with Dynamic Meshes

The preconditioning matrix $\mathbf{\Gamma}$ in Eq. (6) destroys the temporal accuracy of Eq. (1). The dual-time stepping method [31] formulated as

$$\mathbf{\Gamma} \frac{\partial \mathbf{q}^p}{\partial \tau} + \frac{\partial \mathbf{q}^c}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = 0, \quad (15)$$

can be an intuitive solution for preconditioned unsteady flow problems. In this work, the BDF2 method is employed to discretize the physical time derivative term, thus achieving a second-order accuracy temporally. When dynamic meshes are involved, one can first transfer Eq. (15) from the physical domain (t, x, y) into the computational domain (t^*, ξ, η) as

$$\mathbf{\Gamma} |\mathbf{J}| \frac{\partial \mathbf{q}^p}{\partial \tau} + \frac{\partial |\mathbf{J}| \mathbf{q}^c}{\partial t^*} + \frac{\partial \mathbf{F}}{\partial \xi} + \frac{\partial \mathbf{G}}{\partial \eta} = 0, \quad (16)$$

where

$$\begin{cases} \mathbf{F} = |\mathbf{J}| (\mathbf{q}^c \xi_t + \mathbf{f} \xi_x + \mathbf{g} \xi_y) \\ \mathbf{G} = |\mathbf{J}| (\mathbf{q}^c \eta_t + \mathbf{f} \eta_x + \mathbf{g} \eta_y). \end{cases} \quad (17)$$

The GCL of the transformation can be formulated as

$$\begin{cases} \frac{\partial}{\partial \xi} (|\mathbf{J}| \xi_x) + \frac{\partial}{\partial \eta} (|\mathbf{J}| \eta_x) = 0 \\ \frac{\partial}{\partial \xi} (|\mathbf{J}| \xi_y) + \frac{\partial}{\partial \eta} (|\mathbf{J}| \eta_y) = 0 \\ \frac{\partial |\mathbf{J}|}{\partial t^*} + \frac{\partial}{\partial \xi} (|\mathbf{J}| \xi_t) + \frac{\partial}{\partial \eta} (|\mathbf{J}| \eta_t) = 0. \end{cases} \quad (18)$$

In this study, $t^* = t$, thus,

$$\mathbf{J} = \frac{\partial(x, y, t)}{\partial(\xi, \eta, t^*)} = \begin{pmatrix} x_\xi & x_\eta & x_{t^*} \\ y_\xi & y_\eta & y_{t^*} \\ 0 & 0 & 1 \end{pmatrix} \quad (19)$$

and

$$|\mathbf{J}| = x_\xi y_\eta - x_\eta y_\xi, \quad \begin{cases} \xi_t = -\mathbf{v}_g \cdot \nabla \xi, \\ \eta_t = -\mathbf{v}_g \cdot \nabla \eta. \end{cases} \quad (20)$$

Herein, $\mathbf{v}_g = (u_g, v_g)^\top$ is the grid velocity vector. Since grid velocities are only related to the physical time t or t^* , when marching along the pseudo-time τ , GCL can be enforced analytically to eliminate the error discrepancy between BDF2 and the spatial FR/CPR schemes following the approach in Ref. [27, 28]. Specifically, Eq. (16) can be written as

$$\mathbf{\Gamma} |\mathbf{J}| \frac{\partial \mathbf{q}^p}{\partial \tau} + |\mathbf{J}| \frac{\partial \mathbf{q}^c}{\partial t^*} - \mathbf{q}^c \left(\frac{\partial |\mathbf{J}| \xi_t}{\xi} + \frac{\partial |\mathbf{J}| \eta_t}{\eta} \right) + \frac{\partial \mathbf{F}}{\partial \xi} + \frac{\partial \mathbf{G}}{\partial \eta} = 0, \quad (21)$$

As a result, the GCL error only depends on the accuracy of the spatial discretization. Finally, the governing equation in the physical domain reads

$$\mathbf{\Gamma} \frac{\partial \mathbf{q}^p}{\partial \tau} + \frac{\partial \mathbf{q}^c}{\partial t^*} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} - u_g \frac{\partial \mathbf{q}^c}{\partial x} - v_g \frac{\partial \mathbf{q}^c}{\partial y} = 0. \quad (22)$$

When dynamic meshes are involved, the preconditioned Jacobian matrix in the normal direction of any surface for the convection terms becomes $\mathbf{\Gamma}^{-1} \mathbf{A}_{\mathbf{n}, mov}$, where

$$\mathbf{A}_{\mathbf{n}, mov} = n_x \frac{\partial \mathbf{f}_{inv}}{\partial \mathbf{q}^p} + n_y \frac{\partial \mathbf{g}_{inv}}{\partial \mathbf{q}^p} - u_g \frac{\partial \mathbf{q}^c}{\partial \mathbf{q}^p} n_x - v_g \frac{\partial \mathbf{q}^c}{\partial \mathbf{q}^p} n_y. \quad (23)$$

The eigenvalues of this system preserve the same form as those in Eq. (12) or Eq. (13) except a minor change, i.e., u_n in Eq. (13) is modified to

$$u_n = (\mathbf{v} - \mathbf{v}_g) \cdot \mathbf{n}, \quad (24)$$

due to the presence of the grid velocity. The preconditioning parameter ϵ for dynamic meshes is still defined as [32]

$$\epsilon = \min(1, \max(\kappa Ma_\infty, Ma_{mov})), \quad (25)$$

where Ma_{mov} is the Mach number calculated from $\mathbf{v} - \mathbf{v}_g$.

3 Numerical Methods

3.1 Spatial Discretization Method

For completeness, a brief review of the FR/CPR method [33, 34, 35, 36] is presented in this section. In FR/CPR, fluxes can be divided into two parts, namely, local fluxes constructed directly from local solutions, and correction fluxes accounting for the differences between the local fluxes and common fluxes constructed from those on element interfaces. In the current study, the spatial domain is discretized into non-overlapping quadrilateral elements. When the FR/CPR method is employed for the spatial discretization, Eq. (22) can be reformulated as

$$\begin{aligned} \mathbf{F} \frac{\partial \mathbf{q}^p}{\partial \tau} + \frac{\partial \mathbf{f}^{loc}}{\partial x} + \frac{\partial \mathbf{g}^{loc}}{\partial y} - u_g \frac{\partial \mathbf{q}^{c,loc}}{\partial x} - v_g \frac{\partial \mathbf{q}^{c,loc}}{\partial y} \\ + \frac{1}{|\mathbf{J}|} \left(\frac{\partial \mathbf{F}^{cor}}{\partial \xi} + \frac{\partial \mathbf{G}^{cor}}{\partial \eta} \right) + \frac{\partial \mathbf{q}^c}{\partial t^*} = 0. \end{aligned} \quad (26)$$

For quadrilateral elements, \mathbf{F}^{cor} and \mathbf{G}^{cor} are given by

$$\begin{cases} \mathbf{F}^{cor}(\xi, \eta) = \left(\mathbf{F}_L^{num}(\eta) - \mathbf{F}^{loc}(\xi_L, \eta) \right) g_L^c(\xi) \\ \quad + \left(\mathbf{F}_R^{num}(\eta) - \mathbf{F}^{loc}(\xi_R, \eta) \right) g_R^c(\xi), \\ \mathbf{G}^{cor}(\xi, \eta) = \left(\mathbf{G}_L^{num}(\xi) - \mathbf{G}^{loc}(\xi, \eta_L) \right) g_L^c(\eta) \\ \quad + \left(\mathbf{G}_R^{num}(\xi) - \mathbf{G}^{loc}(\xi, \eta_R) \right) g_R^c(\eta), \end{cases} \quad (27)$$

where \mathbf{F}^{num} and \mathbf{G}^{num} are the common (or numerical) fluxes on the interfaces of standard elements. Herein, subscripts ‘ L ’ and ‘ R ’ stand for left and right edges of an element in a dimension by dimension sense. g_L^c, g_R^c are correction functions which map the differences between the common and local fluxes on the element interfaces into the entire standard element. In this study, g_L^c and g_R^c are the right and left Radau polynomials, which can recover the nodal DG methods. Note that common fluxes are always calculated in the physical domain. Thus \mathbf{F}^{num} and \mathbf{G}^{num} can be expressed as

$$\begin{cases} \mathbf{F}^{num} = |\mathbf{J}| |\nabla \xi| \mathbf{f}_{\mathbf{n}}^{com} \text{sign}(\mathbf{n} \cdot \nabla \xi), \\ \mathbf{G}^{num} = |\mathbf{J}| |\nabla \eta| \mathbf{f}_{\mathbf{n}}^{com} \text{sign}(\mathbf{n} \cdot \nabla \eta), \end{cases} \quad (28)$$

where $\mathbf{f}_{\mathbf{n}}^{com}$ are the common fluxes in the normal direction \mathbf{n} of a physical surface. On element interfaces, the inviscid fluxes along \mathbf{n} can be written as

$$\mathbf{f}_{\mathbf{n},inv} = \mathbf{f} n_x + \mathbf{g} n_y - (\mathbf{v}_g \cdot \mathbf{n}) \mathbf{q}^c. \quad (29)$$

Therefore, $\mathbf{f}_{\mathbf{n},inv}^{com}$ can be formulated as, taking the Rusanov approximate Riemann solver as an example,

$$\mathbf{f}_{\mathbf{n},inv}^{com} = \frac{1}{2} \left(\mathbf{f}_{\mathbf{n},inv}^+ + \mathbf{f}_{\mathbf{n},inv}^- \right) - \frac{1}{2} |\lambda|_{max} \tilde{\mathbf{F}} (\mathbf{q}^{p,+} - \mathbf{q}^{p,-}). \quad (30)$$

where $\tilde{\mathbf{\Gamma}}$ is the local preconditioning matrix calculated by averaging primitive variables on elements interfaces, and superscripts ‘+’ and ‘-’ denote the left and right side of the interface, respectively. λ stands for the eigenvalue of the Jacobian matrix $\mathbf{\Gamma}^{-1}\mathbf{A}_{\mathbf{n}}$ or $\mathbf{\Gamma}^{-1}\mathbf{A}_{\mathbf{n},mov}$.

The preconditioning does not affect the viscous terms and $\mathbf{f}_{n,vis}^{com} = \mathbf{f}_{vis}(\mathbf{q}^{p,+}, \nabla \mathbf{q}^{p,+}, \mathbf{q}^{p,-}, \nabla \mathbf{q}^{p,-})$. To calculate the common viscous fluxes, we need to define common $\mathbf{q}^{p,com}$ and common $\nabla \mathbf{q}^{p,com}$ on element interfaces. Simply taking average of the primitive variables yields

$$\mathbf{q}^{p,com} = \frac{\mathbf{q}^{p,+} + \mathbf{q}^{p,-}}{2}. \quad (31)$$

The common gradient is calculated following the the second approach of Bassi and Rebay (BR2) [37] as

$$\nabla \mathbf{q}^{p,com} = \frac{\nabla \mathbf{q}^{p,+} + \mathbf{r}^+ + \nabla \mathbf{q}^{p,-} + \mathbf{r}^-}{2}, \quad (32)$$

where \mathbf{r}^+ and \mathbf{r}^- are the corrections to the gradients on the interfaces. See Ref. [38] for more information.

3.2 Time Marching Method

For steady flow problems, define $\mathbf{R} = -\left(\frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y}\right)$ and substitute it into Eq. (11) to obtain

$$\mathbf{\Gamma} \frac{\partial \mathbf{q}^p}{\partial \tau} = \mathbf{R}. \quad (33)$$

A backward Euler method is used to discretize the pseudo-time derivative, and the residual \mathbf{R} is linearized around \mathbf{R}^m at the pseudo-time step m . As a result, the linear system to be solved becomes

$$\left(\frac{\mathbf{\Gamma}^m}{\Delta \tau} - \left(\frac{\partial \mathbf{R}}{\partial \mathbf{q}^p}\right)^m\right) \Delta(\mathbf{q}^p)^m = \mathbf{R}^m, \quad (34)$$

where $\Delta(\mathbf{q}^p)^m = (\mathbf{q}^p)^{m+1} - (\mathbf{q}^p)^m$. The initial pseudo-time step size $\Delta \tau_0 = 0.01$, and it will grow as the residual drops following

$$\Delta \tau_{m+1} = \max\left(\tau_{min}, \min\left(\tau_{max}, \Delta \tau_m \left(\frac{Res_{m-1}/Res_0}{Res_m/Res_0}\right)^2\right)\right), \quad (35)$$

where Res_m/Res_0 is the relative residual of pressure at the pseudo-time step m , $\tau_{min} = \tau_0$ and $\tau_{max} = 10^{20}$. This approach is a simple modification of the switched evolution relaxation (SER) developed by Mulder and van Leer [39]. We have observed a good convergence with this aggressive pseudo-time step size adaptation strategy through numerical experiments.

For unsteady flow problems with the dual-time stepping method, define $\mathbf{R} = -\left(\frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} - u_g \frac{\partial \mathbf{q}^c}{\partial x} - v_g \frac{\partial \mathbf{q}^c}{\partial y}\right)$. On plugging it into Eq. (22), we obtain

$$\mathbf{\Gamma} \frac{\partial \mathbf{q}^p}{\partial \tau} = -\frac{\partial \mathbf{q}^c}{\partial t^*} + \mathbf{R}. \quad (36)$$

If n denotes the current physical-time step, and m denotes the pseudo-time step, then we have

$$\mathbf{\Gamma}^m \frac{(\mathbf{q}^p)^{m+1} - (\mathbf{q}^p)^m}{\Delta \tau} = -\left(\frac{\partial \mathbf{q}^c}{\partial t^*}\right)^{m+1} + \mathbf{R}^{m+1}, \quad (37)$$

where $(\mathbf{q}^p)^{m=0} = (\mathbf{q}^p)^n$ and when $m \rightarrow \infty$, $(\mathbf{q}^p)^{m+1} \rightarrow (\mathbf{q}^p)^{n+1}$. To achieve a second-order temporal accuracy, BDF2 is used to discretize the derivative with respect to t^* . Thus, Eq. (37) is discretized as

$$\mathbf{\Gamma}^m \frac{(\mathbf{q}^p)^{m+1} - (\mathbf{q}^p)^m}{\Delta \tau} = -\frac{3(\mathbf{q}^c)^{m+1} - 4(\mathbf{q}^c)^n + (\mathbf{q}^c)^{n-1}}{2\Delta t} + \mathbf{R}^{m+1}. \quad (38)$$

Note that $\Delta t = \Delta t^*$. After linearizing \mathbf{R}^{m+1} with respect to \mathbf{R}^m , Eq. (38) can be further written as

$$\begin{aligned} & \left(\frac{\mathbf{\Gamma}^m}{\Delta \tau} + \frac{3}{2\Delta t} \mathbf{M}^m - \left(\frac{\partial \mathbf{R}}{\partial \mathbf{q}^p} \right)^m \right) \Delta(\mathbf{q}^p)^m \\ & = -\frac{3(\mathbf{q}^c)^m - 4(\mathbf{q}^c)^n + (\mathbf{q}^c)^{n-1}}{2\Delta t} + \mathbf{R}^m, \end{aligned} \quad (39)$$

where $\Delta(\mathbf{q}^p)^m = (\mathbf{q}^p)^{m+1} - (\mathbf{q}^p)^m$. For unsteady flow problems tested here, $\Delta \tau$ is always fixed as $\tau = 10^{20}$ for pseudo-time iterations.

In the present study, the full matrix $\frac{\partial \mathbf{R}}{\partial \mathbf{q}^p}$ is computed numerically. Note that if only block diagonal terms of the Jacobian matrix were supplied, the convergence would be dramatically deteriorated [22]. The restarted GMRES solver in the PETSc library [40] is used to solve Eq. (34) and Eq. (39).

3.3 Boundary Conditions

The characteristics of the inviscid parts of the compressible Navier–Stokes equation are changed due to the low-Mach-number preconditioning. No-reflection far-field boundary conditions or simplified far-field boundary conditions have been proposed in Refs. [13, 22]. However, the far field in this study is always located more than 100 characteristic lengths (e.g., the diameter of a cylinder, the chord length of an airfoil) away from the solid wall. Therefore, we simply use $\mathbf{q}^{p,b} = \mathbf{q}^{p,\infty}$ on boundaries at the far field, and enforce boundary conditions using the approximate Riemann solver. For inviscid flows, the slip wall boundary condition is employed. The adiabatic wall boundary condition is enforced for no-slip viscous walls. To ensure high-order accuracy on wall boundaries, P^4 mesh elements are employed to handle the curvatures of wall boundaries.

4 Numerical Results

Several benchmark tests have been conducted to verify the accuracy and convergence of the preconditioned FR/CPR method in this section. The L_2 error is used for the accuracy study. Specifically, for any solution variables ϕ on domain Ω , the L_2 error is defined as

$$E_\phi = \sqrt{\frac{\int_\Omega (\phi - \phi^{exact})^2 dV}{\int_\Omega 1 dV}}. \quad (40)$$

Exclusively, the entropy error E_s is defined as

$$E_s = \sqrt{\frac{\int_\Omega (p/\rho^\gamma - p_\infty/\rho_\infty^\gamma)^2 dV}{\int_\Omega (p_\infty/\rho_\infty^\gamma)^2 dV}}. \quad (41)$$

To monitor the convergence of nonlinear systems, such as Eqs. (33) and (36), two convergence criteria are set up for the pseudo-time iteration: one is for Res_m/Res_0 , denoted as tol_{pseudo} ; the other is for the restarted GMRES linear solver, denoted as tol_{res} . For steady flow simulation using a pseudo-transient continuation (see Eq. (33)), the goal is to decrease the steady relative residual Res_m/Res_0 as much as possible with a reasonable criterion for tol_{res} ; for unsteady flow simulation (see Eq. (36)), the goal is to decrease the unsteady relative residual Res_m/Res_0 to a satisfactory level with a reasonable criterion for tol_{res} .

The drag coefficient C_d , lift coefficient C_l and thrust coefficient C_t are defined as

$$C_d = \frac{Drag}{\frac{1}{2}\rho_\infty U_\infty^2 A}, \quad C_l = \frac{Lift}{\frac{1}{2}\rho_\infty U_\infty^2 A}, \quad \text{and} \quad C_t = -C_d, \quad (42)$$

where A is the area that the force acts on, and U_∞ is the free stream velocity. The pressure is normalized as

$$p_{norm} = \frac{p - p_{min}}{p_{max} - p_{min}}, \quad (43)$$

for better visualization, where p_{min} and p_{max} are the minimum and maximum pressure in the entire flow field.

A parallel code is developed to accelerate the simulations. The block Jacobi preconditioner is employed for parallelization, and ILU(0) is used as the local preconditioner for the restarted GMRES solver in PETSc. Each process possesses only one mesh block [41]. The dimension of the Krylov subspace is 150 for steady flow problems and 30 for unsteady ones. The Jacobian matrix $\frac{\partial \mathbf{R}}{\partial \mathbf{q}^p}$ is updated every pseudo-time iteration for steady flow problems and every two pseudo-time iterations for unsteady flow problems. All the steady flow problems are simulated with one process, and all the unsteady cases are solved with 16 processes.

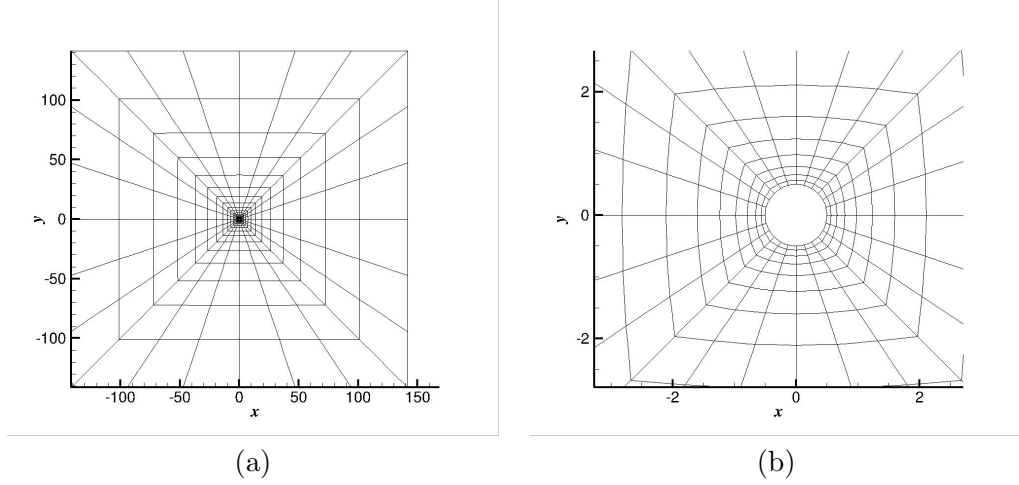


Figure 1: (a) An overview of the computational domain of the baseline 20×24 mesh, and (b) an enlarged view of the mesh close to the cylinder.

Table 1: The grid refinement study for the inviscid flow over a circular cylinder with $tol_{res} = 10^{-6}$.

Mesh	3^{rd}		4^{th}	
	C_d	E_s	C_d	E_s
20×24	5.2720×10^{-4}	7.0302×10^{-12}	9.3369×10^{-5}	2.6179×10^{-13}
40×48	1.7897×10^{-4}	6.9203×10^{-13}	6.7059×10^{-6}	2.0882×10^{-14}
80×96	2.8318×10^{-5}	7.3010×10^{-14}		

4.1 Inviscid Flow over a Circular Cylinder

The solver developed in the present study is tested for the inviscid flow over a circular cylinder (smooth curvature) in this section. The physical domain of the problem is within a square of width $200\sqrt{2}$. The diameter of the circular cylinder is one. The baseline mesh used in this section is illustrated in Figure 1, which has 20 elements in the normal direction of the wall and 24 elements in the circumferential direction. On splitting the baseline mesh once and twice, one can obtain the 40×48 and 80×96 meshes, respectively.

Grid refinements for the third- (P^2) and fourth-order (P^3) FR/CPR schemes have been conducted. The free parameter $\kappa = 1$ is employed. The convergence criterion for the relative residual of the restarted GMRES solver is $tol_{res} = 10^{-6}$ as suggested in Ref. [22]. In Figure 2, the pressure and Ma contours near the cylinder in a free stream with $Ma_\infty = 10^{-3}$ are presented. No pressure-oscillation near the stagnation point is observed. Table 1 documents the drag coefficient C_d and entropy error E_s from a grid refinement study.

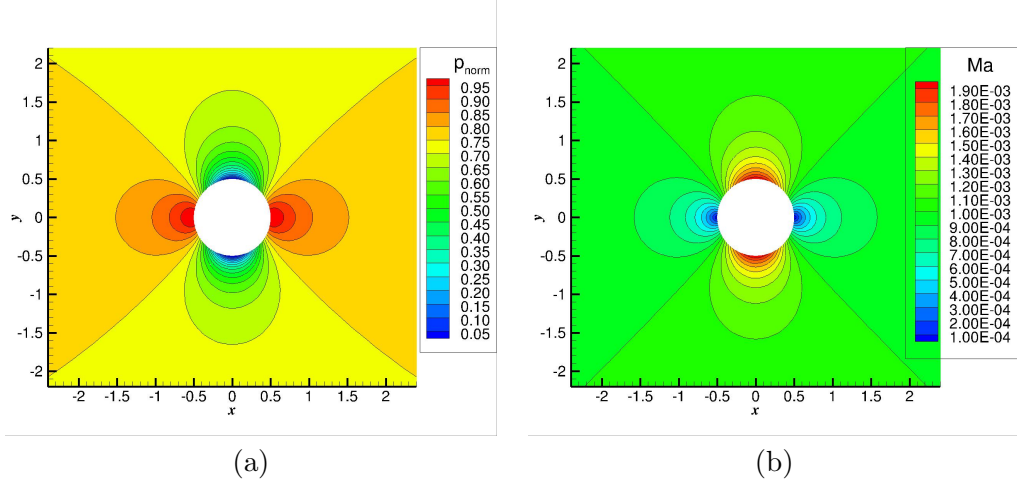


Figure 2: (a) p_{norm} and (b) Ma contours of the third-order FR/CPR scheme on the 80×96 mesh. The free stream Mach number Ma_∞ is 10^{-3} .

The convergence histories of the relative residual Res_m/Res_0 of pressures are illustrated in Figure 3. It is observed that Res_m/Res_0 can decrease to the level of 10^{-9} in steady flow simulation at low Mach numbers. The histories of the pseudo-time step size variation are presented in Figure 4. An observation is that Res_m/Res_0 will quickly drop as the pseudo-time step size increases to large values. However, as shown in Figure 5, the iteration numbers needed for the restarted GMRES solver to converge (i.e., to meet the convergence criterion $tol_{res} = 10^{-6}$) will significantly increase when $\Delta\tau \rightarrow \infty$, especially when fine meshes are used.

Effects of the Global Cut-off Parameter κ

We have tested three different values of κ , namely, $\kappa = \sqrt{0.2}$, $\kappa = 1$ and $\kappa = \sqrt{5}$ to study the effects of κ on convergence and accuracy. The convergence criterion for the relative residual of the restarted GMRES solver is set as $tol_{res} = 10^{-9}$ in these tests. For brevity, only the third-order FR/CPR scheme is considered on the 80×96 mesh. The results of C_d , C_l and E_s are presented in Table 2. The histories of relative residuals and the iteration numbers needed for each pseudo-time stepping are presented in Figure 6. When $\kappa = \sqrt{0.2}$, the global cut-off value is very small. Due to the instability induced by the two stagnation points, the linear solver is not able to converge within the maximum iteration number (5000 for this case), and the simulation quickly diverged. Even though when $\kappa = \sqrt{5}$, Res_m/Res_0 can drop to a smaller value using less pseudo-time iterations compared to that with $\kappa = 1$, the accuracy does not get better, and more iterations are needed for the linear solver to converge in one pseudo-time step. The predicted C_d and E_s both increase due to the fact that when a

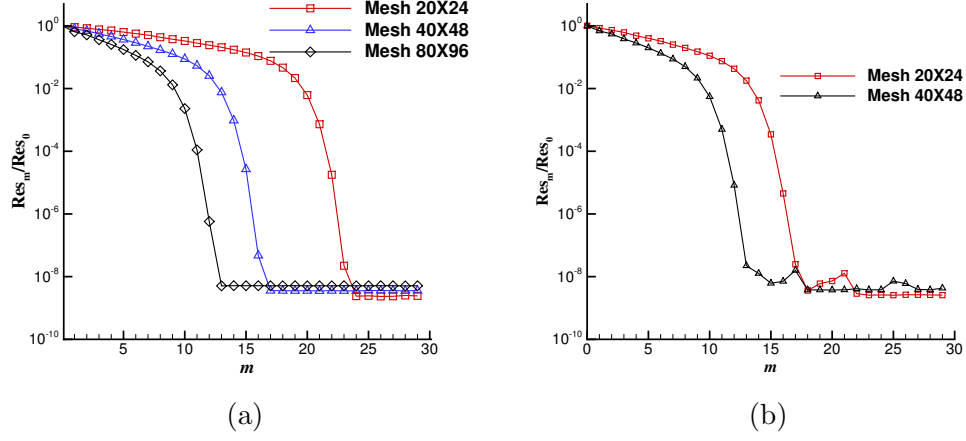


Figure 3: Convergence histories of Res_m/Res_0 of pressures for the (a) third-order and (b) fourth-order FR/CPR schemes on different meshes.

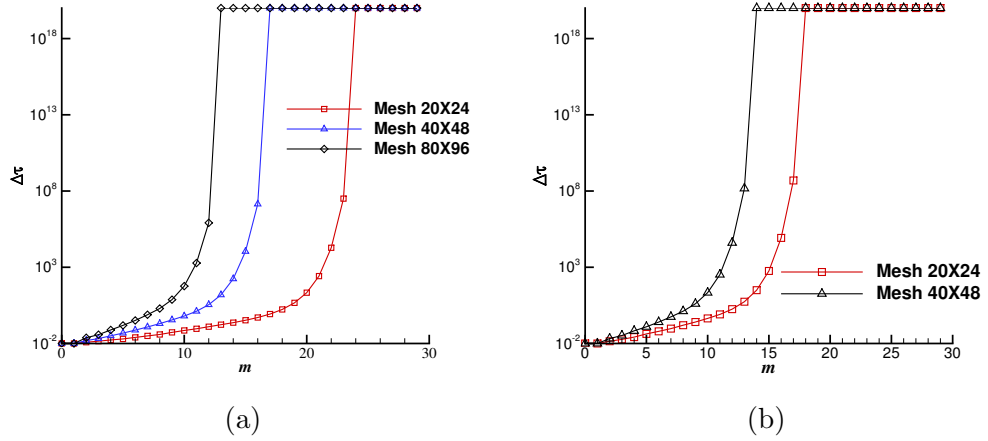


Figure 4: Pseudo time step $\Delta\tau$ histories of the pseudo-time iteration for the restarted GMRES solver on different meshes when the (a) third-order and (b) fourth-order FR/CPR schemes are used.

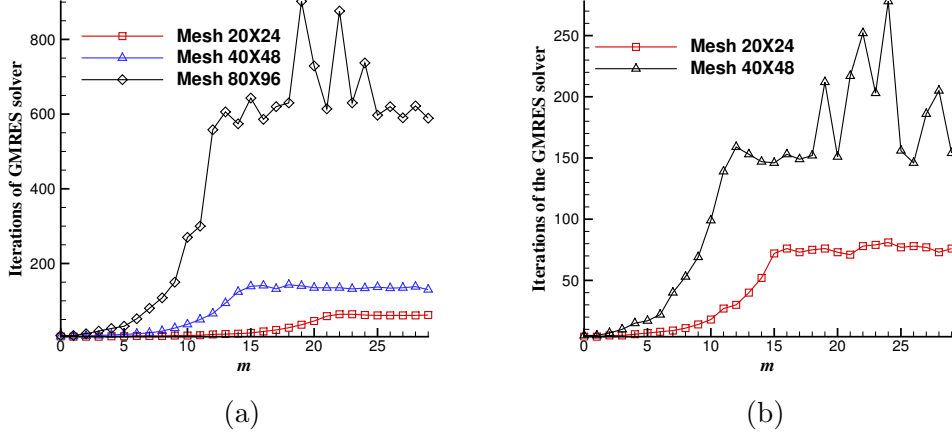


Figure 5: Iteration numbers of the restarted GMRES solver for the (a) third-order and (b) fourth-order FR/CPR schemes on different meshes.

Table 2: C_l , C_d and E_s for different κ 's using the third-order FR/CPR scheme on the 80×96 mesh with $tol_{res} = 10^{-9}$.

κ	C_d	C_l	E_s
$\sqrt{0.2}$	diverged	diverged	diverged
1	2.8319×10^{-5}	6.6839×10^{-7}	7.3013×10^{-14}
$\sqrt{5}$	5.1692×10^{-5}	1.2015×10^{-6}	1.3620×10^{-13}

larger κ is used, more numerical dissipation is added to the approximate Riemann solver.

4.2 Inviscid Flow over the NACA0012 Airfoil

In this section, the inviscid flow of $Ma = 10^{-3}$ and 0° angle of attack (AOA) over the NACA0012 airfoil is studied to verify the performance of the preconditioned FR/CPR solver when there exists a singular point on the geometry. The profile of the airfoil is expressed as

$$y = \pm 0.6 \left(0.2969\sqrt{x} - 0.1260x - 0.3516x^2 + 0.2843x^3 - 0.1036x^4 \right), \quad (44)$$

where $x \in [0, 1]$. The mesh with 5,168 quadrilateral elements is illustrated in Figure 7. The polynomial degree is refined from 2 to 4 (i.e., from the 3rd to 5th order of accuracy). As presented in Table 3, the prediction of C_d becomes more accurate when the degree of the polynomial increases. With preconditioning, instabilities near wall boundaries are suppressed and the flow field in Figure 8 does not show any pressure oscillation near wall boundaries. We have also tested the impact of different criteria tol_{res} , i.e., 10^{-3} , 10^{-6} and 10^{-9} , on the convergence.

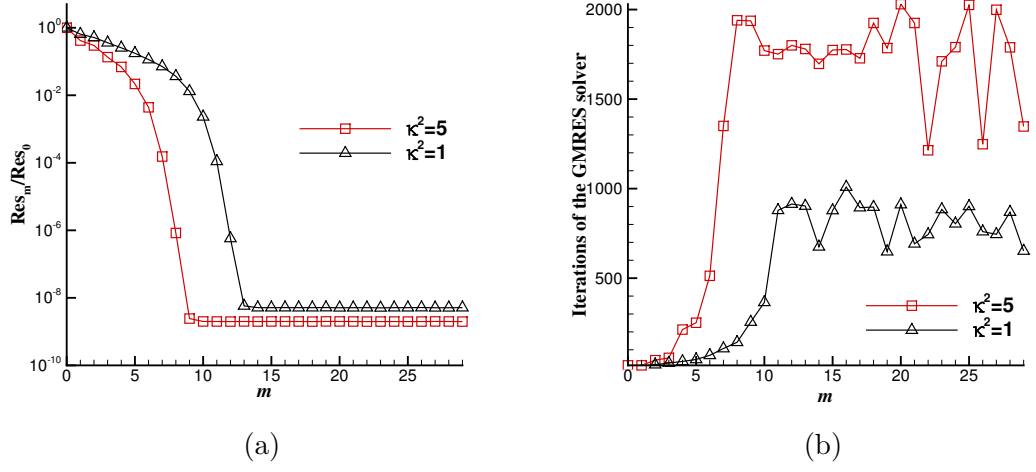


Figure 6: Histories of (a) Res_m/Res_0 , and (b) the corresponding iteration numbers of the restarted GMRES solver for different κ 's.

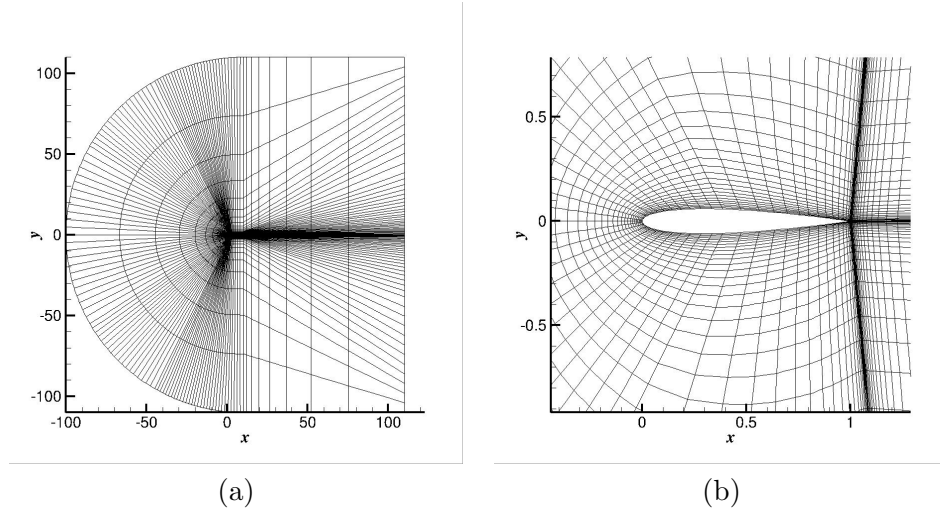


Figure 7: (a) Domain and mesh of the flow over a NACA0012 airfoil, and (b) an enlarged view of the mesh close to the airfoil.

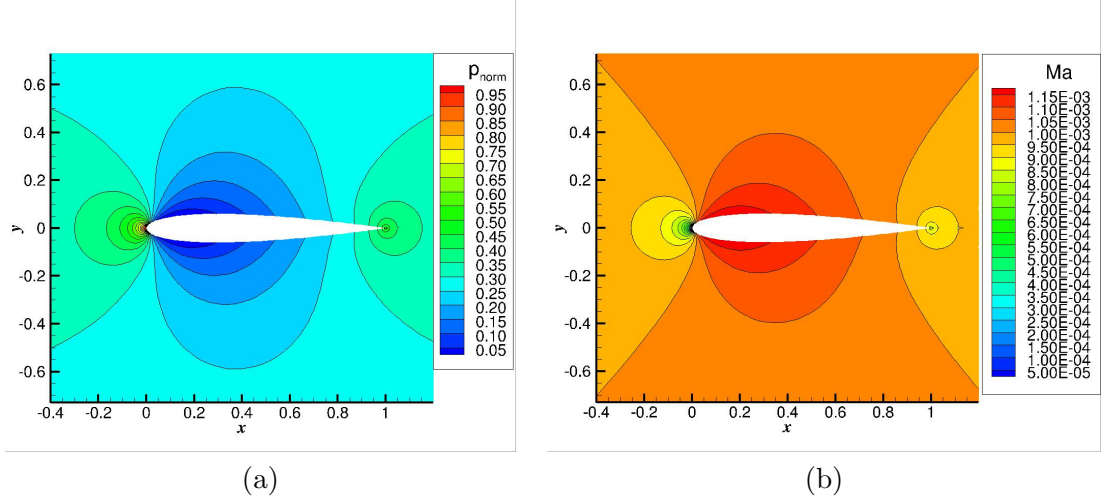
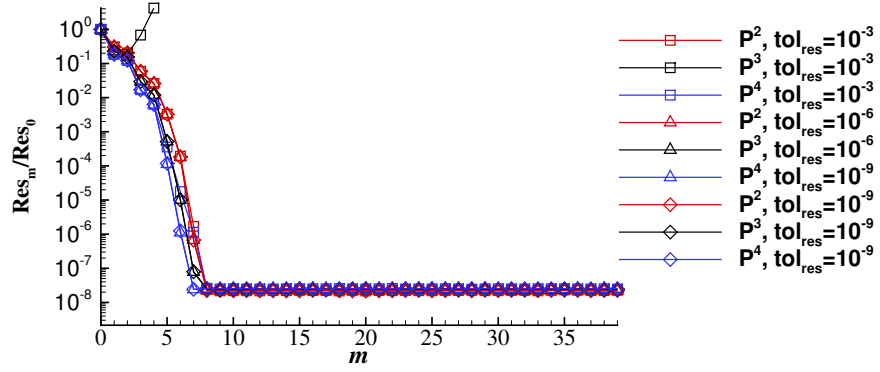


Figure 8: (a) p_{norm} and (b) Ma contours of the flow over a NACA0012 airfoil using the third-order FR/CPR method with $tol_{res} = 10^{-6}$. The free stream Mach number Ma_{∞} is 10^{-3} .

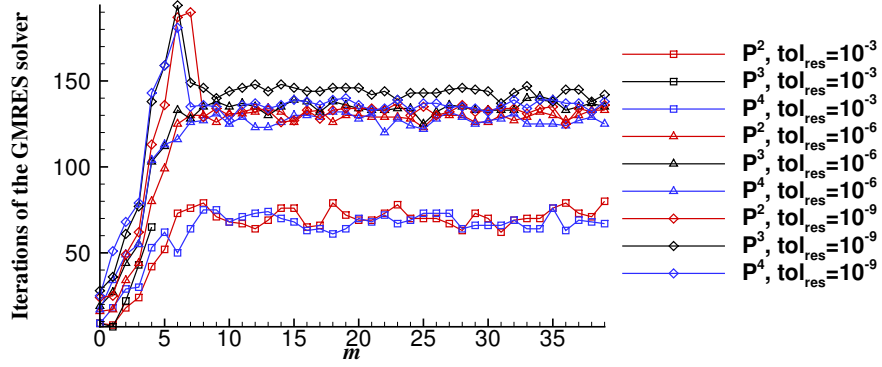
Table 3: The order of accuracy refinement study for the inviscid flow over the NACA0012 airfoil with $tol_{res} = 10^{-6}$.

	3^{rd}	4^{th}	5^{th}
C_d	7.5664×10^{-6}	2.7816×10^{-6}	2.1640×10^{-6}
E_s	2.4161×10^{-12}	4.0361×10^{-13}	1.8262×10^{-13}

In Ref. [22], a 1,792-element mesh is employed and the maximum iteration number of the restarted GMRES solver when $\Delta\tau \rightarrow \infty$ is roughly 120. In this study, a relative denser mesh is used. We observe from Figure 9(b) that the maximum iteration number is about 140 with the same $tol_{res} = 10^{-6}$ as that used in Ref. [22]. According to the convergence comparison of different criteria for the restarted GMRES solver (see Figure 9(a)), when $tol_{res} = 10^{-3}$, both the third-order and fifth-order FR/CPR schemes can converge to correct solutions. However, the simulation using the fourth-order FR/CPR scheme will immediately diverge after a few pseudo-time iterations. A smaller tol_{res} for the linear solver can guarantee convergence, but more iterations of the restarted GMRES solver are needed (see Figure 9(b)). Only negligible differences are observed on C_d and E_s among the converged results of different tol'_{res} s, which are not presented for brevity. Therefore, a proper choice of tol_{res} is a trade-off between robustness and efficiency.



(a)



(b)

Figure 9: Histories of (a) Res_m/Res_0 and (b) iteration numbers of the restarted GMRES solver of the third-, fourth-, and fifth-order FR/CPR schemes with $tol_{res} = 10^{-3}$, 10^{-6} and 10^{-9} for solving the inviscid flow over a NACA0012 airfoil.

4.3 Isentropic Vortex Propagation with Prescribed Grid Motion

The isentropic vortex propagation case is employed to verify both the spatial convergence and temporal convergence of the preconditioned FR/CPR methods for unsteady flow simulation in this section. The isentropic vortex propagation case depicts the superposition of an inviscid uniform flow and an irrotational vortex. The vortex can be regarded as a perturbation added onto the uniform flow. The free stream flow is of $(\rho, u, v, Ma) = (1.0, \sqrt{2}/2, \sqrt{2}/2, 0.05)$ and the gas constant $R = 1.0$ for this case. The perturbation is defined as [41]

$$\begin{cases} \delta u = -\frac{\alpha}{2\pi}(y - y_0)e^{\phi(1-r^2)} \\ \delta v = \frac{\alpha}{2\pi}(x - x_0)e^{\phi(1-r^2)} \\ \delta T = -\frac{\alpha^2(\gamma-1)}{16\phi\gamma\pi^2}e^{2\phi(1-r^2)} \end{cases} \quad (45)$$

where $\phi = \frac{1}{2}$ and $\alpha = 5$ are parameters that define the vortex strength. $r = (x - x_0)^2 + (y - y_0)^2$ is the distance to the center of the vortex (x_0, y_0) . The vortex is propagated in a periodic domain $[-10, 10] \times [-10, 10]$. A prescribed grid motion is employed to validate the moving grid algorithm. The prescribed grid motion is defined as

$$\begin{cases} x(t) = x_r + a_x \sin(2\pi f_n t) \sin(2\pi f_x x_r) \sin(2\pi f_y y_r) \\ y(t) = y_r + a_y \sin(2\pi f_n t) \sin(2\pi f_x x_r) \sin(2\pi f_y y_r) \end{cases} \quad (46)$$

where x_r, y_r are the coordinates at $t = 0$, i.e., coordinates of the uniformly divided grid, and $a_x = 1$, $a_y = 1$, $f_n = 1.0$, $f_x = 0.1$, $f_y = 0.1$. Periodic boundary conditions are applied to all boundaries.

The time step size Δt is refined from $T_n/100$ to $T_n/800$, where $T_n = 1/f_n$ to validate the accuracy of BDF2. A fifth-order FR/CPR scheme and a 96×96 uniformly divided mesh are employed to ensure that the error is dominated by the temporal discretization. We require the relative residual Res_m/Res_0 of the pseudo-time iterations to drop as low as possible within 100 pseudo-time iterations. The tol_{res} of the restarted GMRES solver is 10^{-6} with the maximum iteration number 100. We only simulate this case until $\frac{1}{4}T_n$ when the mesh deformation is maximized to demonstrate the convergence of BDF2 with GCL. An illustration of the pressure contour and the mesh deformation can be found in Figure 10. As observed from Figure 11(a), the optimal convergence rate of BDF2 is achieved for both p and u .

A grid refinement study is also conducted on a uniformly-divided mesh set with 12×12 , 24×24 , 48×48 and 96×96 elements, respectively. The time step size Δt is 10^{-4} to guarantee that the error is dominated by the spatial discretization. The L_2 error of pressure p and velocity u are presented in

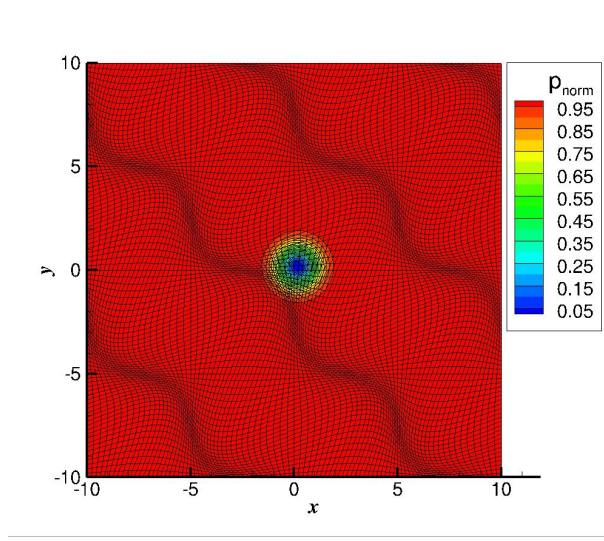


Figure 10: p_{norm} contour and the deformed mesh at $t = \frac{1}{4}T_n$. The fifth-order FR/CPR scheme, 96×96 mesh, and $\Delta t = \frac{1}{800}T_n$ are used in this case.

Figure 11(b). We observe that for both p and u , the numerical orders of a degree k solution construction are close to $k+1/2$ with and without dynamic meshes. Note that the optimal convergence rate is $k+1$. The observed order reduction is reasonable since the Rusanov approximate Riemann solver is employed to calculate the common inviscid fluxes [42]. Hence, the preconditioned FR/CPR methods can demonstrate good convergence for unsteady low-Mach-number flow simulations on dynamic meshes.

4.4 Laminar Flow over a Plunging NACA0012 Airfoil

An order of accuracy refinement study is firstly carried out for a steady case, i.e., the viscous flow of $Re = 500$, $Ma = 10^{-3}$ and 0° AOA over the stationary NACA0012 airfoil. The mesh is the same as that in Section 4.2 (see Figure 7). The drag coefficients of the third-, fourth-, and fifth-order FR/CPR schemes are 0.1723, 0.1722 and 0.1722, respectively. Therefore, we adopt the fourth-order FR/CPR scheme to simulate laminar flows over a plunging NACA0012 airfoil. The plunging motion is defined as $y = Y \sin(2\pi ft)$, where Y is the plunge amplitude, and f is the plunge frequency. The reduced frequency k is defined as $k = 2\pi fC/U_\infty$, and dimensionless height $h = Y/C$, where C is the chord length of the airfoil. In this study, $k = 2$, $h = 0.4$ [8]. The whole mesh is considered as an oscillating rigid body. At the far field, $\mathbf{q}^{p,b} = \mathbf{q}^{p,\infty}$, and the approximate Riemann solver is used to enforce boundary conditions.

The tol_{res} of the restarted GMRES solver is 10^{-6} with the maximum

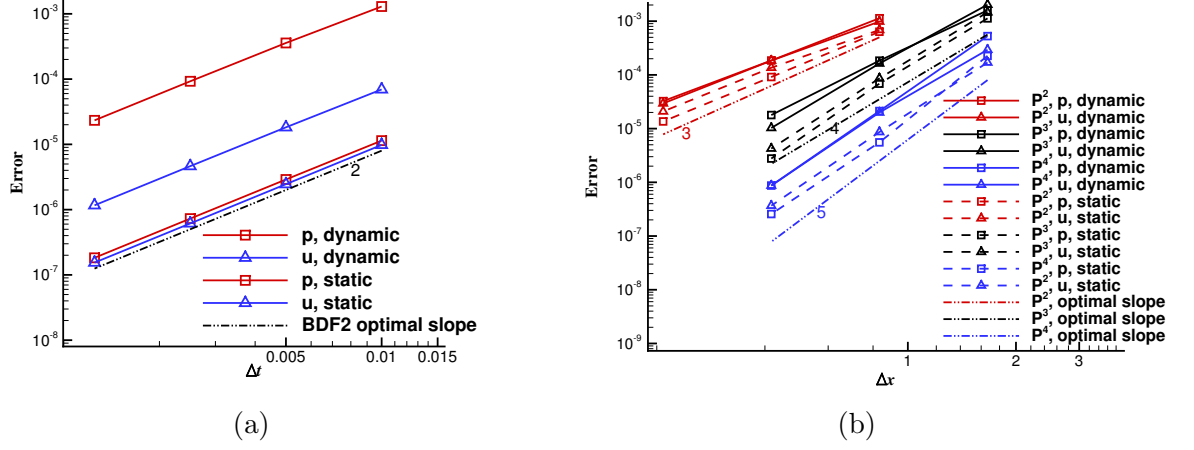


Figure 11: (a) The convergence study of BDF2 on the 96×96 mesh using the fifth-order FR/CPR scheme, and (b) the grid refinement study of the third-, fourth-, and fifth-order FR/CPR schemes.

Table 4: Study on the effects of the convergence criterion tol_{pseudo} on force prediction of a plunging NACA0012 airfoil, $\Delta t = T/100$.

tol_{pseudo}	\bar{C}_t	$C_{l,rms}$	$C_{l,max}$
10^{-2}	0.0589	2.352	3.685
10^{-4}	0.0589	2.351	3.684
10^{-6}	0.0589	2.351	3.684

iteration number 100. For the sake of efficiency, we would require the Res_m/Res_0 drop by certain orders instead of as low as possible. In this case, three criteria for Res_m/Res_0 of the pseudo-time marching have been tested, i.e., $tol_{pseudo} = 10^{-2}$, 10^{-4} , and 10^{-6} , to examine if the force predictions are sensitive to the convergence criterion of Res_m/Res_0 . The time step size Δt is set to $T/100$, and κ is set to $\sqrt{5}$. All simulations end at $t_{end} = 10T$. In Figure 12, the vortex shedding process during one typical plunging period is presented. The thrust coefficient C_t and lift coefficient C_l of the tenth period are displayed in Figure 13, and the time-averaged thrust coefficient \bar{C}_t , root mean square of the lift coefficient $C_{l,rms}$, and maximum lift coefficient $C_{l,max}$ are documented in Table 4. From Figure 13 and Table 4, we observe that a moderate tolerance (10^{-4}) for Res_m/Res_0 is sufficient to maintain the accuracy of force prediction. However, a smaller Res_m/Res_0 criterion means more iterations.

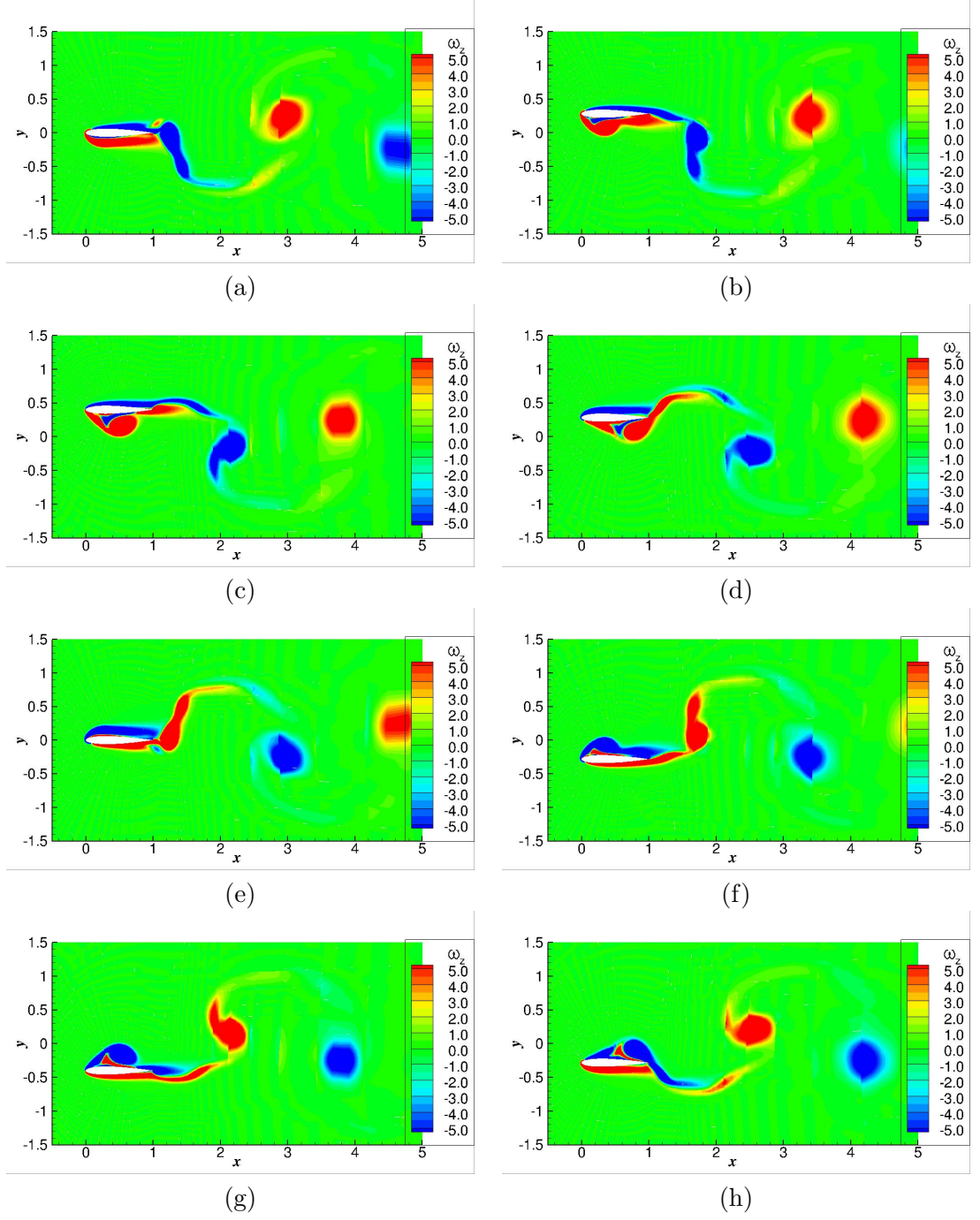


Figure 12: The vortex shedding process over a plunging NACA0012 airfoil in one period. (a) $0T$, (b) $\frac{1}{8}T$, (c) $\frac{2}{8}T$, (d) $\frac{3}{8}T$, (e) $\frac{4}{8}T$, (f) $\frac{5}{8}T$, (g) $\frac{6}{8}T$, and (h) $\frac{7}{8}T$. The contour represents the vorticity in the direction (i.e., z-direction) perpendicular to the observation window.

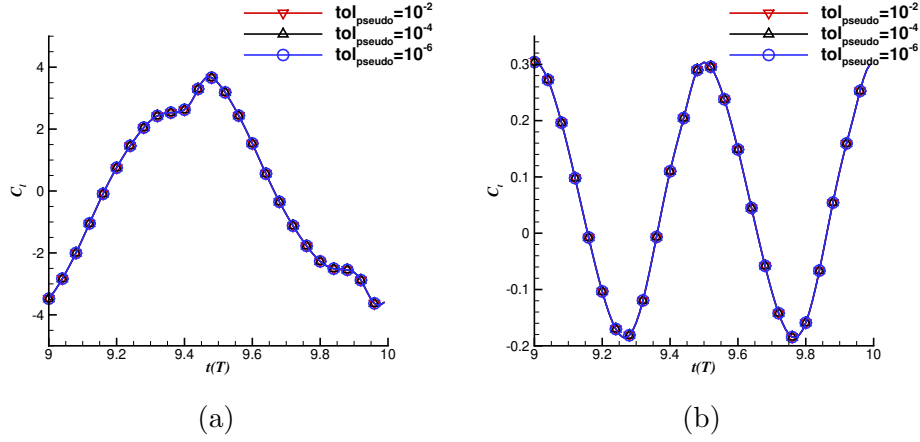


Figure 13: (a) C_l and (b) C_t of the tenth plunging period when different convergence criteria of Res_m/Res_0 are used in the simulation.

Table 5: Study on the effects of the Mach number on force prediction of a plunging NACA0012 airfoil, $\Delta t = T/1000$.

Ma	\bar{C}_t	$C_{l,rms}$	$C_{l,max}$
10^{-3}	0.0578	2.340	3.685
10^{-2}	0.0578	2.341	3.692
10^{-1}	0.0539	2.361	3.774
0.2	0.0370	2.369	3.608
Incompressible [8]	0.0557	2.348	-

Effects of the Mach Number on Force Generation

We further conducted a numerical study to test how the Mach number affects force prediction. Different Mach numbers, i.e., 10^{-3} , 10^{-2} , 10^{-1} and 0.2, are studied here. A smaller time step size is employed, i.e., $\Delta t = T/1000$, to ensure that the time integration errors are negligible. $tol_{pseudo} = 10^{-4}$ is adopted as the convergence criterion for pseudo-time iterations. The C_l and C_t of the tenth period are illustrated in Figure 14. \bar{C}_t , $C_{l,rms}$ and $C_{l,max}$ are documented in Table 5.

It is observed that when Ma is less than 10^{-2} , the compressibility is almost negligible. The force histories for $Ma = 10^{-3}$ and $Ma = 10^{-2}$ coincide with each other. When Ma is larger than 10^{-1} , phase shift is observed in the force histories. In general, when Ma is increased to a certain level (e.g., 10^{-1} in this case), the time-averaged thrust coefficient \bar{C}_t will decrease when Ma increases; at the same time, the root mean square of the lift coefficient $C_{l,rms}$ and the maximum lift coefficient $C_{l,max}$ will vary slightly. For example, when $Ma = 0.2$, the reduction of \bar{C}_t is over 36% compared to that at $Ma = 10^{-3}$; $C_{l,rms}$ will increase about 1%; $C_{l,max}$ will

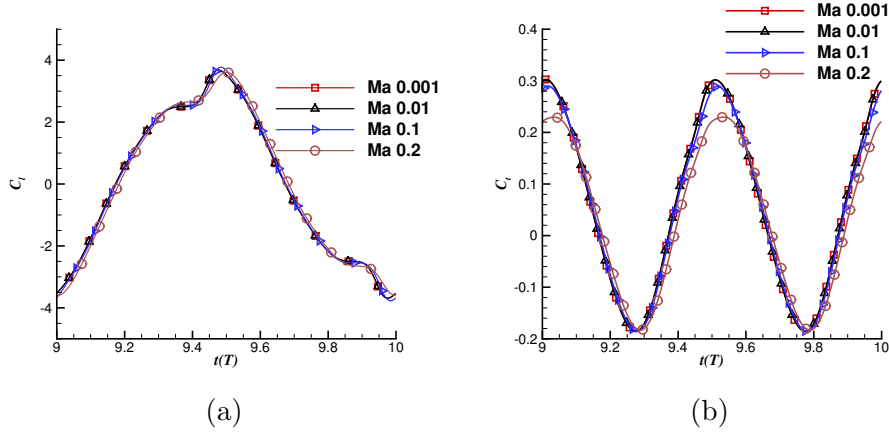


Figure 14: (a) C_l and (b) C_t of the tenth plunging period at different Mach numbers.

first increase and then decrease within a small range.

5 Conclusions

In this study, a high-order preconditioned FR/CPR method is developed to solve compressible Navier–Stokes equations at low Mach numbers on moving grids. The dual-time stepping method is used to handle unsteady flows. Specifically, BDF2 is adopted to discretize the physical time derivative term, and the restarted GMRES linear solver with the modified SER pseudo-time step size adaptation strategy is used to converge unsteady residuals at each physical time. For simulations with moving/deforming grids, the GCL is enforced implicitly to eliminate the discrepancy between the temporal and spatial discretization. The preconditioned FR/CPR method has demonstrated good accuracy and convergence through various benchmark tests.

We first study the convergence properties of the preconditioned FR/CPR method for steady inviscid flow simulations. Two cases, namely, low-Mach-number (i.e., $Ma = 10^{-3}$ in this study) flows over a cylinder and a NACA0012 airfoil are simulated. We observe that when a pseudo-transient continuation is used, the convergence criterion tol_{res} of the restarted GMRES solver is important for the sake of robustness and efficiency. In general, a loose tol_{res} may drive the simulation diverged; while a tight tol_{res} can guarantee convergence of the pseudo-transient continuation, but will lead to more GMRES iterations at each pseudo-time.

The preconditioned FR/CPR method is then used to simulate low-Mach-number unsteady flows. The temporal and spatial orders of accuracy are first verified using the isentropic vortex propagation on both static and dynamic meshes. After that, we study the numerical properties of the preconditioned

FR/CPR method with low-Mach-number flows over a plunging NACA0012 airfoil. We find that the convergence tolerance tol_{pseudo} for Res_m/Res_0 of the pseudo-time marching does not need to be very small to preserve simulation accuracy. Since less pseudo-time iterations are needed when a larger criterion for Res_m/Res_0 is used, computational cost of unsteady flows can be decreased. We also conduct numerical experiments to test the effects of the Mach number on unsteady force generation. We find that when the Mach number is small enough (e.g., $Ma = 10^{-2}$), flow compressibility has negligible effect on force prediction of a plunging airfoil. However, when the Mach number becomes larger (e.g., $Ma = 0.2$), thrust generation of a plunging airfoil will change significantly while the change of lift and its root mean square are not apparent.

Acknowledgment

The authors gratefully acknowledge the support of the Office of Naval Research through the award N00014-16-1-2735, and the faculty startup support from the department of mechanical engineering at University of Maryland, Baltimore County (UMBC).

References

- [1] S. Patankar, *Numerical Heat Transfer and Fluid Flow*. CRC press, 1980.
- [2] A. J. Chorin, “Numerical solution of the Navier-Stokes equations,” *Mathematics of Computation*, vol. 22, no. 104, pp. 745–762, 1968.
- [3] A. J. Chorin, “A numerical method for solving incompressible viscous flow problems,” *Journal of Computational Physics*, vol. 2, no. 1, pp. 12–26, 1967.
- [4] B. T. Helenbrook, “Artificial compressibility preconditioning for incompressible flows under all conditions,” in *the 44th Aerospace Sciences Meeting and Exhibit Conference*, (Reno, NV), 2006. AIAA-2006-0689.
- [5] F. Bassi, A. Crivellini, D. A. Di Pietro, and S. Rebay, “An artificial compressibility flux for the discontinuous Galerkin solution of the incompressible Navier–Stokes equations,” *Journal of Computational Physics*, vol. 218, no. 2, pp. 794–815, 2006.
- [6] A. Crivellini, V. D’Alessandro, and F. Bassi, “High-order discontinuous Galerkin solutions of three-dimensional incompressible RANS equations,” *Computers & Fluids*, vol. 81, pp. 122–133, 2013.
- [7] C. Cox, C. Liang, and M. W. Plesniak, “A high-order solver for unsteady incompressible Navier–Stokes equations using the flux reconstruction method on unstructured grids with implicit dual time stepping,” *Journal of Computational Physics*, vol. 314, pp. 414–435, 2016.
- [8] M. L. Yu and L. Wang, “A high-order flux reconstruction/correction procedure via reconstruction formulation for unsteady incompressible flow on unstructured moving grids,” *Computers & Fluids*, vol. 139, pp. 161–173, 2016.
- [9] E. Turkel, A. Fiterman, and B. Van Leer, “Preconditioning and the limit to the incompressible flow equations,” tech. rep., Institute for Computer Applications in Science and Engineering, Hampton VA, 1993.
- [10] Y. Colin, H. Deniau and J-F. Boussuge, “A robust low speed preconditioning formulation for viscous flow computations,” *Computers & Fluids*, vol. 47, no. 1, pp. 1–15, 2011.
- [11] A. J. Chorin, “A numerical method for solving incompressible viscous flow problems,” *Journal of Computational Physics*, vol. 2, pp. 12–26, 1967.

- [12] E. Turkel, “Preconditioned methods for solving the incompressible and low speed compressible equations,” *Journal of Computational Physics*, vol. 72, no. 2, pp. 277–298, 1987.
- [13] E. Turkel, R. Radespiel and N. Kroll, “Assessment of preconditioning methods for multidimensional aerodynamics,” *Computers & Fluids*, vol. 26, no. 6, pp. 613–634, 1997.
- [14] E. Turkel, “Preconditioning techniques in computational fluid dynamics,” *Annual Review of Fluid Mechanics*, vol. 31, no. 1, pp. 385–416, 1999.
- [15] Y. H. Choi and C. L. Merkle, “The application of preconditioning in viscous flows,” *Journal of Computational Physics*, vol. 105, pp. 207–233, 1993.
- [16] J. M. Weiss and W. A. Smith, “Preconditioning applied to variable and constant density Flows,” *AIAA Journal*, vol. 33, no. 11, 1995.
- [17] B. van Leer, W. Lee, and P. Roe, “Characteristic time-stepping or local preconditioning of the Euler equations,” in *the 10th Computational Fluid Dynamics Conference*, (Honolulu, HI), 1991. AIAA-91-1552-CP.
- [18] D. Lee and B. van Leer, “Progress in local preconditioning of the Euler and Navier-Stokes equations,” in *the 11th AIAA Computational Fluid Dynamics Conference*, (Orlando, FL), 1993. AIAA-93-3328-CP.
- [19] G. Hauke and T. Hughes, “A comparative study of different sets of variables for solving compressible and incompressible flows,” *Computer Methods in Applied Mechanics and Engineering*, vol. 153, pp. 1–44, 1998.
- [20] S. Venkateswaran and L. Merkle, “Analysis of preconditioning methods for the Euler and Navier-Stokes equations,” in *Lecture series-van Karemman Institute for fluid dynamics 3*, 1995. B1-B155.
- [21] D. Vigneron, G. Deliége and J. A. Essers, “Low mach number local preconditioning for unsteady viscous finite volumes simulations on 3D unstructured grids,” in *ECCOMAS CFD 2006: Proceedings of the European Conference on Computational Fluid Dynamics*, (Egmond aan Zee, The Netherlands), 2006.
- [22] F. Bassi, C. De Bartolo, R. Hartmann and A. Nigro, “A discontinuous Galerkin method for inviscid low Mach number flows,” *Journal of Computational Physics*, vol. 228, pp. 3996–4011, 2009.
- [23] H. Luo, J. D. Baum and R. Löhner, “A Fast p -Multigrid Discontinuous Galerkin Method for Compressible Flows at All Speeds,” *AIAA Journal*, vol. 46, no. 3, pp. 635–652, 2008.

- [24] M. Feistauer and V. Kucera, “On a robust discontinuous Galerkin technique for the solution of compressible flow,” *Journal of Computational Physics*, vol. 224, no. 1, pp. 208–221, 2007.
- [25] R. H. A. Nigro, C. De Bartolo and F. Bassi, “Discontinuous Galerkin solution of preconditioned Euler equations for very low Mach number flows,” *International Journal for Numerical Methods in Fluids*, vol. 63, pp. 449–467, 2010.
- [26] S. M. Renda, R. Hartmann, C. De Bartolo and M. Wallraff, “A high-order discontinuous Galerkin method for all-speed flows,” *International Journal for Numerical Methods in Fluids*, vol. 77, pp. 224–247, 2015.
- [27] M. L. Yu, Z. J. Wang and H. Hu, “A high-order spectral difference method for unstructured dynamic grids,” *Computers & Fluids*, vol. 48, pp. 84–97, 2011.
- [28] C. Liang, K. Miyaji, and B. Zhang, “An Efficient Correction Procedure via Reconstruction for Simulation of Viscous Flow on Moving and Deforming Domains,” *Journal of Computational Physics*, vol. 256, pp. 55–68, 2014.
- [29] L. Wang and M. L. Yu, “A Preconditioned Flux Reconstruction/Correction Procedure via Reconstruction Formulation for Unsteady Low Mach Number Flows on Dynamic Unstructured Meshes,” in *the 55th AIAA Aerospace Sciences Meeting*, (Grapevine, TX), 2017. AIAA-2017-0737.
- [30] R. H. Pletcher, “On solving the compressible Navier-Stokes equations for unsteady flows at very low Mach numbers,” in *the 11th Computational Fluid Dynamics Conference*, 1993. AIAA-93-3368-CP.
- [31] A. Jameson, “Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings,” in *the 10th Computational Fluid Dynamics Conference*, (Honolulu, HI), 1991.
- [32] T. H. Xiao, H. S. Ang and S. Z. Yu, “A preconditioned dual time-stepping procedure coupled with matrix-free LU-SGS scheme for unsteady low speed viscous flows with moving objects,” *International Journal of Computational Fluid Dynamics*, vol. 21, no. 3-4, pp. 165–173, 2007.
- [33] H. T. Huynh, “A flux reconstruction approach to high-Order schemes including discontinuous Galerkin methods,” in *the 18th AIAA Computational Fluid Dynamics Conference*, (Miami, FL), 2007. AIAA-2007-4079.

- [34] H. T. Huynh, “A reconstruction approach to high-order schemes including discontinuous Galerkin methods for diffusion,” in *the 47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace*, (Orlando, FL), 2009. AIAA-2009-403.
- [35] Z. J. Wang and H. Y. Gao, “A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids,” *Journal of Computational Physics*, vol. 98, pp. 209–220, 2014.
- [36] P. E. Vincent, P. Castonguay, and A. Jameson, “A new class of high-order energy stable flux reconstruction schemes,” *Journal of Scientific Computing*, no. 1, pp. 50–72, 2011.
- [37] F. Bassi and S. Rebay, “Discontinuous Galerkin solution of the Reynolds-averaged Navier-Stokes and $k\text{-}\omega$ turbulence model equations,” *Computers & Fluids*, vol. 34, pp. 507–540, 2005.
- [38] H. Y. Gao, Z. J. Wang and H. T. Huynh, “Differential formulation of discontinuous Galerkin and related methods for the Navier-Stokes equations,” *Communications in Computational Physics*, vol. 13, no. 4, pp. 1013–1044, 2013.
- [39] W. A. Mulder and B. van Leer, “Experiments with Implicit Upwind Methods for the Euler Equations,” *Journal of Computational Physics*, vol. 59, pp. 232–246, 1985.
- [40] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, D. A. May, L. C. McInnes, R. T. Mills, T. Munson, K. Rupp, P. Sanan, B. F. Smith, S. Zampini, H. Zhang, and H. Zhang, “PETSc users manual,” Tech. Rep. ANL-95/11 - Revision 3.10, Argonne National Laboratory, 2018.
- [41] F. Bassi, L. Botti, A. Colombo, A. Ghidoni and F. Massa, “Linearly implicit Rosenbrock-type Runge-Kutta schemes applied to the Discontinuous Galerkin solution of compressible and incompressible unsteady flows,” *Computers & Fluids*, vol. 118, no. 2, pp. 305–320, 2015.
- [42] J. S. Hesthaven and T. Warburton, *Nodal discontinuous Galerkin methods – algorithms, analysis and applications*. New York: Springer, 2008.