Journal of Computational Physics 331 (2017) 227–256

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A Newton–Krylov method with an approximate analytical Jacobian for implicit solution of Navier–Stokes equations on staggered overset-curvilinear grids with immersed boundaries

Hafez Asgharzadeh, Iman Borazjani *

337 Jarvis Hall, The State University of New York at Buffalo, Buffalo, NY, United States

ARTICLE INFO

Article history:
Received 12 January 2016
Received in revised form 3 October 2016
Accepted 22 November 2016
Available online 25 November 2016

Keywords:
Implicit method
Complex geometry
Newton–Krylov method
Matrix-free
Analytical Jacobian
Overset
Curvilinear

ABSTRACT

The explicit and semi-implicit schemes in flow simulations involving complex geometries and moving boundaries suffer from time-step size restriction and low convergence rates. Implicit schemes can be used to overcome these restrictions, but implementing them to solve the Navier–Stokes equations is not straightforward due to their non-linearity. Among the implicit schemes for non-linear equations, Newton-based techniques are preferred over fixed-point techniques because of their high convergence rate but each Newton iteration is more expensive than a fixed-point iteration. Krylov subspace methods are one of the most advanced iterative methods that can be combined with Newton methods, i.e., Newton–Krylov Methods (NKMs) to solve non-linear systems of equations. The success of NKMs vastly depends on the scheme for forming the Jacobian, e.g., automatic differentiation is very expensive, and matrix-free methods without a preconditioner slow down as the mesh is refined. A novel, computationally inexpensive analytical Jacobian for NKM is developed to solve unsteady incompressible Navier–Stokes momentum equations on staggered overset-curvilinear grids with immersed boundaries. Moreover, the analytical Jacobian is used to form a preconditioner for matrix-free method in order to improve its performance. The NKM with the analytical Jacobian was validated and verified against Taylor–Green vortex, inline oscillations of a cylinder in a fluid initially at rest, and pulsatile flow in a 90 degree bend. The capability of the method in handling complex geometries with multiple overset grids and immersed boundaries is shown by simulating an intracranial aneurysm. It was shown that the NKM with an analytical Jacobian is 1.17 to 14.77 times faster than the fixed-point Runge–Kutta method, and 1.74 to 152.3 times (excluding an intensively stretched grid) faster than automatic differentiation depending on the grid (size) and the flow problem. In addition, it was shown that using only the diagonal of the Jacobian further improves the performance by 42–74% compared to the full Jacobian. The NKM with an analytical Jacobian showed better performance than the fixed point Runge–Kutta because it converged with higher time steps and in approximately 30% less iterations even when the grid was stretched and the Reynold number was increased. In fact, stretching the grid decreased the performance of all methods, but the fixed-point Runge–Kutta performance decreased 4.57 and 2.26 times more than NKM with a diagonal and full Jacobian, respectivley, when the stretching factor was increased. The NKM with a diagonal analytical Jacobian and matrix-free method with an analytical preconditioner are the fastest methods and the superiority of one to another depends on the flow problem. Furthermore, the implemented methods are fully parallelized with parallel efficiency of

* Corresponding author.
E-mail address: iman@buffalo.edu (I. Borazjani).

http://dx.doi.org/10.1016/j.jcp.2016.11.033

80–90% on the problems tested. The NKM with the analytical Jacobian can guide building preconditioners for other techniques to improve their performance in the future.

1. Introduction

Simulating flows involving arbitrarily moving complex geometries are of interest in many areas, such as flow inside the vascular systems [1–7], swimming/flying in nature [8–13], flows with suspended particles [14–16], etc. Currently, one of the main challenges in simulating such flows is the small time-step size restriction, e.g., Griffith [17] discussed the severe restriction on the time-step size imposed by his explicit scheme in simulations of an aortic heart valve using the immersed boundary method. Mittal et al. [18] used an implicit Crank–Nicolson scheme for the diffusion terms to eliminate the numerical instabilities, while the non-linear convective terms were treated explicitly in their immersed boundary method. The simulations of Mangual et al. [19], Domenichini and Pedrizzetti [20], de Tullio et al. [21], and Tytell et al. [22] all used the immersed boundary method, which sets small time-step restrictions due to their explicit nature. Robust implicit time integrations are needed to reduce the time-step limitations of such simulations, which will lower the computational cost and allow for higher spatial resolution. However, developing fully implicit methods for Navier–Stokes equations is not straightforward due to their non-linearity. In these cases, the methods for solving non-linear systems of equations need to be employed. There are two main categories of implicit solvers for non-linear systems of equations: 1) fixed-point methods; and 2) Newton-based methods. The applied mathematics community has emphasized on Newton-based methods for boundary value problems in which the steady-state solution of non-linear equations is desired [23]. However, the computational physics community has focused on fixed-point type methods for initial value problems, where dynamic evolution of the equations is required [23].

The first category is interchangeably named as fixed-point, Picard linearization, and successive substitution method [24]. The fixed-point method for solving the system of non-linear algebraic equations $F(\overrightarrow{U}) = 0$ can be written as:

$$F(\overrightarrow{U}^{(k)}) = \overrightarrow{U}^{(k+1)} - \overrightarrow{U}^{(k)} = \triangle\overrightarrow{U}^{(k)} \tag{1}$$

where k is the fixed-point iteration index. Dual time-stepping methods can be viewed as a fixed-point iteration as well [25,26]. Fixed-point methods are implemented in conjunction with the pressure–velocity coupling algorithms such as artificial compressibility [26], SIMPLE family [27], PISO [28], and fractional-step [29] for solving incompressible Navier–Stokes equation. Fractional-step requires smaller time-step size in comparison to SIMPLE and PISO, while it requires less correction iterations than SIMPLE and PISO [30].

The second category is the Newton method, which is a classical algorithm for finding a solution to the system of non-linear algebraic equations $F(\overrightarrow{U}) = 0$ [31,32]. Given an initial guess $\overrightarrow{U}^{(0)}$, a sequence of steps $\triangle\overrightarrow{U}^{(k)}$ is computed as follows:

$$\begin{aligned} &A(\overrightarrow{U}^{(k)})\triangle\overrightarrow{U}^{(k)} = -F(\overrightarrow{U}^{(k)}) \\ &\overrightarrow{U}^{(k+1)} = \overrightarrow{U}^{(k)} + \triangle\overrightarrow{U}^{(k)} \end{aligned} \tag{2}$$

where k is Newton iteration index and $A = \frac{d\overrightarrow{F}}{d\overrightarrow{U}}$ is the associated Jacobian matrix [31,33]. The Newton Eq. (2) is a system of linear algebraic equations which must be solved at each Newton iteration (k). Newton-based methods have been previously implemented in conjunction with the pressure–velocity coupling algorithms such as velocity–vorticity method [34,35], as a multigrid smoother in a SIMPLE method [36], and fractional-step [37] for solving incompressible Navier–Stokes equation. The main advantage of Newton methods over fixed-point type methods is the ability to take large time-steps and super-linear convergence rates [38]. On the other hand, requiring a sufficiently good initial condition is the Newton methods' drawback [39,38]. Considering the advantages of Newton methods, they are implemented in this framework and are systematically compared against a fixed-point method.

The Newton equation at each k can be solved by either direct or iterative methods. Because A is typically approximated (the approximation of A does not affect the solution as long as the Newton iterations converge) and the initial condition is typically far from the solution, the exact solution at each iteration (k) may not be justified. Therefore, an iterative method is typically used for solving Newton equations at each iteration. In this case, the exact solution of Eq. (2) is not obtained at each iteration k, which is referred to as the inexact Newton method [40]. Therefore, a strict quadratic convergence is typically not achieved [23]. Some of the most advanced iterative methods are the Krylov subspace methods [41–44] including the generalized minimal residual method (GMRES) [45] and preconditioned GMRES [46]. Krylov subspace methods were introduced by Hestenes et al. [47] and Reid [48] as iterative methods to solve large linear systems of equations.

For solving Newton equations using a Krylov subspace method either the Jacobian (A) or the Jacobian-vector product ($A\triangle\overrightarrow{U}$) should be formed by: (a) a matrix-free (MF); (b) an automatic differentiation (AD); or (c) an analytical Jacobian