Available online at www.sciencedirect.com
ScienceDirect
Appl. Math. Comput. 00 (2024) 1–27
Appl. Math. Comput.

Regular article (submitted in revised form)

Efficient finite element strategy using enhanced high-order and second-derivative-free variants of Newton's method

Aymen Laadhari[a], Helmi Temimi[b]

[a]Mathematics Department, College of Computing and Mathematical Sciences, Khalifa University of Science and Technology, Abu Dhabi, UAE
[b]College of Integrative Studies, Abdullah Al Salem University, Khaldiya, Kuwait

Abstract

In this work, we propose a stable finite element approximation by extending higher-order Newton's method to the multidimensional case for solving nonlinear systems of partial differential equations. This approach relies solely on the evaluation of Jacobian matrices and residuals, eliminating the need for computing higher-order derivatives. Achieving third and fifth-order convergence, it ensures stability and allows for significantly larger time steps compared to explicit methods. We thoroughly address accuracy and convergence, focusing on the singular p-Laplacian problem and the time-dependent lid-driven cavity benchmark. A globalized variant incorporating a continuation technique is employed to effectively handle high Reynolds number regimes. Through two-dimensional and three-dimensional numerical experiments, we demonstrate that the improved cubically convergent variant outperforms others, leading to substantial computational savings, notably halving the computational cost for the lid-driven cavity test at large Reynolds numbers.

Keywords: High order Newton type methods, Non-linear problems, Solution convergence, Finite element method, Incompressible flows

1. Introduction

Nonlinear problems are prevalent across various fields and applications. This work focuses on the numerical solution of highly nonlinear partial differential equation (PDE) systems using enhanced variants of the Newton method. Several strategies have been developed to approximate solutions for nonlinear systems, with the classical Newton-Raphson method being one of the most well-known [1]. This method achieves quadratic convergence under certain conditions, relying essentially on a continuously differentiable global function and an initial guess that is sufficiently close to the solution. The Newton method has been widely applied to address highly nonlinear problems, particularly within finite element [2, 3], finite difference [4, 5, 6, 7, 8, 9] and finite volume [10, 11, 12] frameworks.

In the literature on nonlinear algebraic root-finding problems $f(x) = 0$, numerous advanced multistep Newton modifications have been developed to achieve higher convergence orders, with some methods avoiding the need for higher-order derivatives. Evaluating second Frchet derivatives is particularly time-consuming, especially for systems of coupled PDEs. While achieving higher-order convergence is mathematically appealing, the overall computational cost becomes a crucial factor when solving nonlinear coupled PDEs. The most computationally expensive operations

Email addresses: aymen.laadhari@ku.ac.ae (Aymen Laadhari), helmi.temimi@aasu.edu.kw (Helmi Temimi)

typically involve calculating higher-order derivatives and inverting the associated matrices. In the following, we reference significant contributions to high-order Newton-like methods that rely solely on first-order derivatives. For solving PDEs, these methods involve evaluating residuals and inverting Jacobian matrices.

For nonlinear algebraic root-finding problems, Homeier [13], followed by Kou et al. [14], and later Chun [15] and Li et al. [16], introduced third-order Newton variants. The method in [13] requires two first-order derivative evaluations and two function evaluations per iteration, whereas [14] requires only one first-order derivative evaluation and two function evaluations. In [17], a "memory" method achieves a convergence order of $1 + \sqrt{2}$ with just one function evaluation and one derivative computation per iteration, reusing the same derivative in two consecutive iterations. Improved fourth-order convergence variants have been developed, as shown in [18, 19]. Notably, the methods from [20] and [21] achieve this with only two function evaluations and one first-derivative evaluation. The presence of multiple roots generally reduces the convergence rate of the scheme unless modifications are made to restore it. Behl et al. [22] developed fourth-order variants for cases where the solution multiplicity is known. Similarly, Zafar et al. [23] proposed fourth-order methods using a weight function approach to handle multiple roots. Additionally, Sharma et al. [24] introduced a derivative-free fourth-order iterative method based on the Traub-Steffensen approach, which was further improved by Ahmad et al. [25] with high-order methods that do not require memory.

To further enhance convergence, Singh et al. [26] introduced fifth-order, two-step variants requiring two function evaluations, two Jacobian evaluations, and two matrix inversions per iteration. A fifth-order convergence was also achieved using homotopy perturbation in [27]. Abdulhassan et al. [28] developed a fifth-order variant by combining Halley's method and Taylor expansion with orthogonal Hermite polynomials for efficient second-order derivative approximation. Sharma et al. [29] generalized the cubically convergent method from [13] to a three-step iterative approach with fifth-order convergence, avoiding higher-order derivatives. A sixth-order Newton variant, combining Taylor series expansion with Halley's method, is proposed in [30]. In addition, recent higher-order variants are developed by Grau et al. [31], Cordero et al. [32], Behl et al. [33], Xiao et al. [34], Sharma et al. [35, 36, 37], and recent high-order schemes by [38, 39, 40, 41, 42, 43, 44], covering cases with both simple and multiple roots. Last but not least, we refer to the Newton-Krylov methods, which approximate the Jacobian matrices by using approximations of the Jacobian-vector products (see, e.g., [45, 46, 47]).

Most of the aforementioned works evaluated the computational efficiency of the methods, commonly using metrics like Ostrowski's efficiency index, computed using the convergence order and the computational cost per iteration [48]. However, their numerical tests typically consider algebraic functions or small ODE systems, often lacking a comprehensive assessment in the context of coupled nonlinear PDEs. Works on higher-order Newton variants for non-linear coupled PDEs are limited. Notable examples include [49], [50], [51], and [26], which applied a fifth-order method to a 1D reaction-diffusion problem. [52] employed a four-step high-order variant for solving the 1D Fisher PDE in population dynamics. Additionally, [32] used a high-order variant for a nonlinear 1D heat conduction equation with finite differences. Quasi-Newton approaches with accelerated convergence were utilized in [34] and [4] for transport problems within a multiscale hybrid mixed finite element framework. Furthermore, [53, 54, 55] applied a cubically convergent variant to 2D two-phase flow problems, while [56] focused on fluid-membrane interactions.

In transient problems, the classical Newton method leads to fully implicit schemes that enhance stability. Our focus is on reducing computational costscritical for 2D and 3D simulationswhile improving convergence order. To this end, we aim to generalize certain higher-order weighted-Newton variants to multidimensional cases, relying exclusively on Jacobian matrix computations and avoiding higher-order derivatives. In this work, we consider the cubic method introduced by Homeier et al. [13] and the fifth-order variant developed by Singh et al. [26]. We selected these two-step Newton variants based on preliminary testing and their straightforward extension to the multidimensional PDE framework. Homeier's method requires two Jacobian assemblies and factorizations, along with two linear solves and a single residual evaluation per iteration. In contrast, Singh's variant achieves fifth-order convergence with two residual evaluations and two Jacobian assemblies and factorizations per iteration. For the numerical investigation, we evaluate the accuracy and stability of solving the stationary and highly nonlinear p-Laplacian problem, as well as the lid-driven cavity problem involving time-dependent Navier-Stokes equations across small, mild, and high Reynolds regimes in both two and three dimensions. Our results demonstrate that implicit strategies, particularly when enhanced with a globalized BDF2-Newton variant, offer significant computational savings compared to the standard quadratically convergent Newton methods or linear fixed-point approaches.

The paper is structured as follows. Section 2 introduces the extension of higher-order iterative variants to the

multidimensional case. In Section 3, we present the problems used to evaluate the computational efficiency of the methods, as well as the globalized variant based on a BDF2-Newton continuation algorithm. In Section 4, we conduct numerical experiments to highlight the main features of the method. Concluding remarks are provided in Section 5.

2. Mathematical setting

Consider an open, bounded, connected domain $\Omega \in \mathbb{R}^d, d > 1$, and an unknown $\boldsymbol{u}$ belonging to a suitable Sobolev space $W(\Omega)$. Let $W^{-1}(\Omega)$ denote the dual space of $W(\Omega)$. The duality pairing between W^{-1} and W is denoted as $\langle ., . \rangle$. We aim to solve the generic nonlinear problem $\boldsymbol{\mathcal{F}}(\boldsymbol{u}) = \boldsymbol{0}$ with a continuously differentiable functional:

$$\begin{aligned} \boldsymbol{\mathcal{F}} : W(\Omega) &\longrightarrow W^{-1}(\Omega) \\ \boldsymbol{u} &\longmapsto \boldsymbol{\mathcal{F}}(\boldsymbol{u}). \end{aligned}$$

The residue $\boldsymbol{\mathcal{F}}(\boldsymbol{u})$, in time-dependent problems, represents the system resulting from semi-discretization in time. The variational formulation $\langle \boldsymbol{\mathcal{F}}(\boldsymbol{u}), \boldsymbol{\xi} \rangle = \boldsymbol{0}$ for all test functions $\boldsymbol{\xi}$ will be discretized within a finite element framework.

2.1. Newton-Raphson method

Let $u^{(0)}$ be the initial guess and set $k = 0$. For $k \geqslant 0$, the increment $\delta \boldsymbol{u}^{(k+1)}$ is computed iteratively. Given an operator $D\boldsymbol{\mathcal{F}}$ from $W^{-1}(\Omega)$ into $W(\Omega)$, let $D_{\boldsymbol{u}}\boldsymbol{\mathcal{F}}(\boldsymbol{u})[\delta \boldsymbol{u}]$ stands for the Fréchet derivative of $\boldsymbol{\mathcal{F}}$ around the point $\boldsymbol{u}$ along the direction $\delta \boldsymbol{u} \in W(\Omega)$. The residue is $\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)}) \in W^{-1}(\Omega)$. Newton's method converts $\boldsymbol{\mathcal{F}}(\boldsymbol{u}) = \boldsymbol{0}$ into a sequence of linear sub-problems, which can be compactly expressed as

$$\begin{aligned} k \geqslant 0 : \quad & D_{\boldsymbol{u}}\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)})[\delta \boldsymbol{u}^{(k+1)}] = -\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)}) \\ & \boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \delta \boldsymbol{u}^{(k+1)}. \end{aligned} \tag{2.1}$$

The stopping criterion can be residual-based and is given by: $\|\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)})\| < \max\left\{\epsilon_r \|\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(0)})\|, \epsilon_a\right\}$, where ϵ_r and ϵ_a represent the relative and absolute tolerances, respectively. The iterative process stops if the target level of accuracy or the maximum number of iterations is reached. Newton's iterations can be summarized as follows:

1. Evaluate the residual $\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)})$;
2. Assemble the Jacobian $\boldsymbol{J}^{(k)} = D_{\boldsymbol{u}}\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)})$;
3. Solve the Jacobian system $\boldsymbol{J}^{(k)} \delta \boldsymbol{u}^{(k+1)} = -\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)})$ for the update $\delta \boldsymbol{u}^{(k+1)}$;
4. Update the solution $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \delta \boldsymbol{u}^{(k+1)}$ and increment the iteration counter $k \leftarrow k + 1$;
5. Test convergence.

The Jacobian matrix must be nonsingular at the solution $\boldsymbol{u}^*$. Moreover, the local convergence properties require $\boldsymbol{u}^{(0)}$ to be chosen within a sufficiently close neighborhood of $\boldsymbol{u}^*$, where $\boldsymbol{\mathcal{F}}$ has a unique and simple root. To address sensitivity to the initial guess, the process can be initialized by performing a few fixed-point iterations or by solving a coarse nonlinear problem. The analysis of convergence requires conditions such as a Lipschitz continuous derivative or assumptions on the degree of non-linearity of $\boldsymbol{\mathcal{F}}$, i.e., concerning the variation of $D_{\boldsymbol{u}}\boldsymbol{\mathcal{F}}$; we refer, e.g., to [57, 58]. Under these conditions, the method converges superlinearly, often quadratically. Assuming $\boldsymbol{\mathcal{F}}$ exhibits sufficient differentiability near a simple root $\boldsymbol{u}^*$ and meets the necessary and sufficient conditions for convergence, this method will be referred to using the acronym "N2".

2.2. Newton's variant with third-order convergence

We extend the two-step Newton's variant introduced in [13] to the multidimensional case. This method achieves cubic convergence while requiring only one residual evaluation and the assembly and factorization of two Jacobian matrices per iteration. By avoiding second-order derivative evaluations, it maintains a manageable computational cost. Each iteration involves solving two linear systems, which is straightforward once a Newton solver is in place. Starting with an initial guess $u^{(0)} \in W(\Omega)$ at $k = 0$, the algorithm iterates the following steps until convergence:

$$\begin{aligned} k \geqslant 0 : \quad D_{\boldsymbol{u}}\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)})\left[\boldsymbol{u}^{(k+1/2)} - \boldsymbol{u}^{(k)}\right] &= -\frac{1}{2}\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)}), \\ D_{\boldsymbol{u}}\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k+1/2)})\left[\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k)}\right] &= -\boldsymbol{\mathcal{F}}(\boldsymbol{u}^{(k)}). \end{aligned}$$

Therefore, the method will be referred to by the acronym "N3" and is summarized as follows:

1. Calculate the residual $\mathcal{F}(\boldsymbol{u}^{(k)})$;
2. Assemble the Jacobian $\boldsymbol{J}^{(k)} = D_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{u}^{(k)})$;
3. Solve for the update $\delta\boldsymbol{u}^{(k+1/2)}$ the linear system:

$$\left\langle \boldsymbol{J}^{(k)}\delta\boldsymbol{u}^{(k+1/2)}, \boldsymbol{\xi}\right\rangle = -\frac{1}{2}\left\langle \mathcal{F}(\boldsymbol{u}^{(k)}), \boldsymbol{\xi}\right\rangle, \quad \forall \boldsymbol{\xi};$$

4. Compute an intermediate solution $\boldsymbol{u}^{(k+1/2)} = \boldsymbol{u}^{(k)} + \delta\boldsymbol{u}^{(k+1/2)}$;
5. Assemble $\boldsymbol{J}^{(k+1/2)} = D_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{u}^{(k+1/2)})$;
6. Solve for the update $\delta\boldsymbol{u}^{(k+1)}$ the linear system :

$$\left\langle \boldsymbol{J}^{(k+1/2)}\delta\boldsymbol{u}^{(k+1)}, \boldsymbol{\xi}\right\rangle = -\left\langle \mathcal{F}(\boldsymbol{u}^{(k)}), \boldsymbol{\xi}\right\rangle, \quad \forall \boldsymbol{\xi};$$

7. Update the solution $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \delta\boldsymbol{u}^{(k+1)}$;
8. Increment the iteration counter: $k \leftarrow k + 1$;
9. Test the convergence.

We note that computing the Jacobian can be both coding-intensive and computationally expensive. Various approximation techniques exist in the literature, such as neglecting certain terms or extrapolating from previous nonlinear iterations, which lead to quasi-Newton methods, see e.g. [2, 4]. In our case, we can approximate the Jacobian in step (6) using the one assembled in step (2). While this approach can reduce computational cost, it may degrade the convergence of high-order schemes, potentially leading to failure to converge. This is an important topic but is beyond the scope of the present work.

2.3. Newton's variant with fifth-order convergence

We consider an extension of Newton's variant recently proposed recently by [37] to the multidimensional case. This two-step method alternates between a standard Newton step and a second weighted-Newton iteration. Although it requires only one additional residual evaluation compared to the N3 cubic variant, the method achieves fifth-order convergence. It involves two residual evaluations, as well as the assembly and factorization of two Jacobian matrices per iteration, all without requiring evaluations of second-order derivatives. Let $u^{(0)} \in W(\Omega)$ be an initial solution. For $k \geqslant 0$, the $(k+1)$st iteration is expressed in compact form as:

$$\begin{aligned} k \geqslant 0: \quad & D_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{u}^{(k)})\left[\boldsymbol{u}^{(k+1/2)} - \boldsymbol{u}^{(k)}\right] = -\mathcal{F}(\boldsymbol{u}^{(k)}), \\ & D_{\boldsymbol{u}}\mathcal{F}\left(\boldsymbol{u}^{(k+1/2)}\right)\left[\boldsymbol{u}^{(k+1)} - \boldsymbol{u}^{(k+1/2)}\right] = -\left(1 + \frac{\mathcal{F}\left(\boldsymbol{u}^{(k+1/2)}\right)^T \mathcal{F}\left(\boldsymbol{u}^{(k+1/2)}\right)}{\mathcal{F}(\boldsymbol{u}^{(k)})^T \mathcal{F}(\boldsymbol{u}^{(k)})}\right)\mathcal{F}\left(\boldsymbol{u}^{(k+1/2)}\right). \end{aligned}$$

This variant will be denoted by the acronym "N5". It defines the sequence $\boldsymbol{u}^{(k)}$ by recurrence for $k \geqslant 0$ as follows:

1. Compute the residual $\mathcal{F}(\boldsymbol{u}^{(k)})$;
2. Assemble the Jacobian $\boldsymbol{J}^{(k)} = D_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{u}^{(k)})$;
3. Solve for the update $\delta\boldsymbol{u}^{(k+1/2)}$ the linear system:

$$\left\langle \boldsymbol{J}^{(k)}\delta\boldsymbol{u}^{(k+1/2)}, \boldsymbol{\xi}\right\rangle = -\left\langle \mathcal{F}(\boldsymbol{u}^{(k)}), \boldsymbol{\xi}\right\rangle, \quad \forall \boldsymbol{\xi};$$

4. Compute an intermediate solution $\boldsymbol{u}^{(k+1/2)} = \boldsymbol{u}^{(k)} + \delta\boldsymbol{u}^{(k)}$;
5. Compute the residual $\mathcal{F}\left(\boldsymbol{u}^{(k+1/2)}\right)$;
6. Assemble the Jacobian $\boldsymbol{J}^{(k+1/2)} = D_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{u}^{(k+1/2)})$;
7. Solve the linear system for the update $\delta\boldsymbol{u}^{(k+1)}$:

$$\left\langle \boldsymbol{J}^{(k+1/2)}\delta\boldsymbol{u}^{(k+1)}, \boldsymbol{\xi}\right\rangle = -\left\langle \left(1 + \frac{\mathcal{F}(\boldsymbol{u}^{(k+1/2)})^T \mathcal{F}(\boldsymbol{u}^{(k+1/2)})}{\mathcal{F}(\boldsymbol{u}^{(k)})^T \mathcal{F}(\boldsymbol{u}^{(k+1/2)})}\right)\mathcal{F}(\boldsymbol{u}^{(k+1/2)}), \boldsymbol{\xi}\right\rangle, \quad \forall \boldsymbol{\xi};$$

8. Update the solution $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k+1/2)} + \delta\boldsymbol{u}^{(k+1)}$;
9. Increment the iteration counter $k \leftarrow k + 1$;
10. Test the convergence.

3. Nonlinear problems and associated consistent linearization

3.1. Problem 1: p-Laplacian problem

Let $\Omega \subset \mathbb{R}^d$ be a domain with a Lipschitz continuous boundary. Consider the p-Laplace operator Δ_p with $1 < p < \infty$. This quasilinear elliptic partial differential operator of second order generalizes the conventional Laplace operator, recovered for $p = 2$ [59, 60]. The p-Laplacian problem corresponds to the Euler-Lagrange equation for the functional

$$I(u) = \int_\Omega |\boldsymbol{\nabla} u|^p dx.$$

Consider $g \in W^{1,p}(\Omega) \equiv \left\{ v \in W^{1,p}(\Omega) : v = g \text{ on } \partial\Omega \right\}$ and $f \in W^{-1,p}(\Omega)$, where $W^{-1,p}(\Omega)$ is the dual space of $W^{1,p}(\Omega)$ so that $u - g \in W_0^{1,p}(\Omega)$. Equipped with Dirichlet boundary conditions, the problem consists in finding u such that:

$$\begin{aligned} \mathcal{F}(u) \equiv -\text{div}\left(\eta\left(|\boldsymbol{\nabla} u|^2\right)\boldsymbol{\nabla} u\right) - f = & \quad 0 \quad \text{in } \Omega, \\ u = & \quad g \quad \text{on } \partial\Omega, \end{aligned} \tag{3.1}$$

where $\eta : \zeta \in \mathbb{R}^+ \longmapsto \zeta^{(p-2)/2} \in \mathbb{R}^+$ and g is regular enough. The linear Poisson problem corresponds to $p = 2$. The solutions are called p-harmonic functions, with the equation being singular for $p < 2$ and degenerate for $p > 2$ at critical points where $\boldsymbol{\nabla} u = \mathbf{0}$. The well-posedness of (3.1) has been well established in the literature; A comprehensive analysis can be found in [61, 62]. We proceed with a continuous piecewise finite element approximation. The variational formulation leads to: find $u \in W_g^{1,p}(\Omega)$ such that:

$$\int_\Omega \eta\left(|\boldsymbol{\nabla} u|^2\right)\boldsymbol{\nabla} u.\boldsymbol{\nabla} v \, \mathrm{d}x = \int_\Omega f\,\psi \, \mathrm{d}x, \qquad \forall \psi \in W_0^{1,p}(\Omega). \tag{3.2}$$

Firstly, to assess the performances of Newton's method or to compute an appropriate initial guess, the fixed-point algorithm generates a series of linear subproblems where the sequence $u^{(k)}$, $k \geqslant 0$ is computed iteratively as:

- $k = 0$: $u^{(0)} \in W_0^{1,p}(\Omega)$ be given.
- $k \geqslant 0$: $u^{(k)} \in W_0^{1,p}(\Omega)$ being known, find $u^{(k+1)} \in W_0^{1,p}(\Omega)$ such that:

$$\int_\Omega \eta\left(|\boldsymbol{\nabla} u^{(k)}|^2\right)\boldsymbol{\nabla} u^{(k+1)}.\boldsymbol{\nabla} v \;=\; \int_\Omega f\,\psi, \quad \forall v \in W_0^{1,p}(\Omega).$$

Due to the algorithm's sensitivity to the initial guess, we set $u^{(0)}$ as the solution to the linear Poisson problem obtained for $p = 2$. The convergence criterion is defined with a target accuracy of 10^{-10}, which relies on the evaluation of the residual $\mathcal{F}\left(u^{(k)}\right) \in W^{-1,p}(\Omega)$ with $k \geqslant 0$ as follows:

$$\int_\Omega \mathcal{F}\left(u^{(k)}\right)\psi = \int_\Omega \eta\left(|\boldsymbol{\nabla} u^{(k)}|^2\right)\boldsymbol{\nabla} u^{(k)}.\boldsymbol{\nabla} v \; - \int_\Omega f\,\psi, \qquad \forall \psi \in W_0^{1,p}(\Omega).$$

Secondly, we provide the tangent problem for the standard Newton's method. The variational formulation will be adapted for both enhanced variants as outlined in the aforementioned schemes. The Fréchet derivative of $\mathcal{F}$ along the direction δu writes:

$$D_u\mathcal{F}(u)[\delta u] = -\text{div}\left(\eta\left(|\boldsymbol{\nabla} u|^2\right)\boldsymbol{\nabla}\delta u + 2\eta'\left(|\boldsymbol{\nabla} u|^2\right)\left(\boldsymbol{\nabla} u \cdot \boldsymbol{\nabla}\delta u\right)\boldsymbol{\nabla} u\right).$$

Thus, the weak formulation corresponding to the N2 scheme (2.1) writes:

$$\int_\Omega \eta\left(|\boldsymbol{\nabla} u^{(k)}|^2\right)\boldsymbol{\nabla}\delta u^{(k+1)}.\boldsymbol{\nabla}\psi + \int_\Omega 2\eta'\left(|\boldsymbol{\nabla} u^{(k)}|^2\right)\left(\boldsymbol{\nabla} u^{(k)} \cdot \boldsymbol{\nabla}\delta u^{(k+1)}\right)\boldsymbol{\nabla} u^{(k)} \cdot \boldsymbol{\nabla}\psi = -\int_\Omega \mathcal{F}\left(u^{(k)}\right)\psi, \quad \forall \psi \in W_0^{1,p}(\Omega).$$

3.2. Problem 2: Lid-driven cavity benchmark with incompressible viscous flow

Consider the non-dimensionalized Navier-Stokes problem and let Re be the Reynolds number that compares viscous effects to inertia forces. The shear strain rate tensor is given by $\mathbf{D}(\boldsymbol{u}) = (\boldsymbol{\nabla u} + \boldsymbol{\nabla u}^T)/2$. Let $T > 0$ be the simulation period. Given initial and boundary conditions, the problem is to find the fluid velocity $\boldsymbol{u} = \boldsymbol{u}(t, \boldsymbol{x})$ and the pressure $p = p(t, \boldsymbol{x})$ such that

$$\begin{aligned} \mathrm{Re}\,(\partial_t \mathbf{u} + \mathbf{u}.\boldsymbol{\nabla}\mathbf{u}) - \mathbf{div}\,(2\,\mathbf{D}(\boldsymbol{u})) + \boldsymbol{\nabla} p &= \mathbf{0} \ \text{ in } (0,T)\times\Omega, && (3.3a)\\ -\mathrm{div}\,\mathbf{u} &= 0 \ \text{ in } (0,T)\times\Omega, && (3.3b)\\ \mathbf{u} &= \mathbf{u}_b \ \text{ on } \ (0,T)\times\Sigma_D, && (3.3c)\\ \mathbf{u}(0) &= \mathbf{u}_0 \ \text{ in } \ \Omega. && (3.3d) \end{aligned}$$

The computational domain is defined as a unit square in 2D and a unit cube in 3D. The flow is driven by the upper wall, while the remaining walls are held fixed. The flow starts from rest, meaning that $\mathbf{u}_0 = \mathbf{0}$. No-slip conditions are applied to all boundaries, with $\boldsymbol{u} = \boldsymbol{u}_b$ on $\Sigma_D = \partial\Omega$. The wall motion induces vorticity, which diffuses within the cavity, resulting in the driven cavity flow. We introduce the functional spaces:

$$\mathbb{V}(\boldsymbol{u}_b) = \left\{\boldsymbol{v} \in (H^1(\Omega))^d : \boldsymbol{v} = \boldsymbol{u}_b \text{ on } \Sigma_D\right\} \quad \text{and} \quad \mathbb{Q} = L^2_0(\Omega) = \left\{q \in L^2(\Omega) : \int_\Omega q\, d\boldsymbol{x} = 0\right\}.$$

For the semi-discretization in time, the time interval $[0, T]$ is divided into N subintervals $[t_n, t_{n+1}]$, with $n = 0, \cdots, N-1$, each having a constant step size Δt. The unknowns $\boldsymbol{u}_n$ and p_n are computed iteratively at t_n for $n > 0$. Thus, the resulting fully implicit scheme employs either standard or modified Newton variants. The second-order backward differentiation scheme, denoted by BDF2, is used for the momentum equation. The scheme is initialized with the conditions $\boldsymbol{u}_{-1} = \boldsymbol{u}_0 = \boldsymbol{u}(0)$, where $\boldsymbol{u}_{-1}$ is a convenient notation. For $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w} \in (H^1(\Omega))^d$ and $q \in L^2(\Omega)$, we consider the weighted multi-linear forms:

$$a(\boldsymbol{u}, \boldsymbol{v}) = \int_\Omega 2\,\mathbf{D}(\boldsymbol{u}) : \mathbf{D}(\boldsymbol{v}), \quad b(\boldsymbol{u}, q) = -\int_\Omega q\,\mathrm{div}\,\boldsymbol{u}, \quad c(\boldsymbol{u}, \boldsymbol{v}; \boldsymbol{w}) = \int_\Omega \left((\boldsymbol{u}\cdot\boldsymbol{\nabla})\,\boldsymbol{w} + (\boldsymbol{w}\cdot\boldsymbol{\nabla})\,\boldsymbol{u}\right)\cdot\boldsymbol{v}.$$

Let $\boldsymbol{\chi}_n = (\boldsymbol{u}_n, p_n)^T$ be the vector of unknowns and $\boldsymbol{V} = (\boldsymbol{v}, q)^T$ be the corresponding test functions. The global residual $\boldsymbol{\mathcal{F}}(\boldsymbol{\chi}_n)$ of the time-discretized problem writes:

$$\left\langle \boldsymbol{\mathcal{F}}\left(\boldsymbol{\chi}_n^T\right), \boldsymbol{V}^T\right\rangle \equiv \left(\langle \boldsymbol{\mathcal{F}}_\chi\left(\boldsymbol{\chi}_n^T\right), \boldsymbol{v}\rangle_{\mathbb{V}(0)',\mathbb{V}(0)}, \langle \boldsymbol{\mathcal{F}}_p(\boldsymbol{u}_n), q\rangle_{\mathbb{Q}',\mathbb{Q}},\right)^T = \mathbf{0}, \quad \forall \boldsymbol{V},$$

where for any $n > 0$, $\boldsymbol{u}_{n-1}$ and $\boldsymbol{u}_{n-2}$ being known, the residuals are given at each Newton's iteration $k \geqslant 0$ by:

$$\begin{aligned} \left\langle \boldsymbol{\mathcal{F}}_\chi\left(\boldsymbol{\chi}_n^{(k),T}\right), \boldsymbol{v}\right\rangle_{\mathbb{V}(\mathbf{0})',\mathbb{V}(\mathbf{0})} &= \frac{\mathrm{Re}}{2\Delta t}\, m\left(3\boldsymbol{u}_n^{(k)} - 4\boldsymbol{u}_{n-1} + \boldsymbol{u}_{n-2}, \boldsymbol{v}\right) + \frac{\mathrm{Re}}{2} c\left(\boldsymbol{u}_n^{(k)}, \boldsymbol{v}; \boldsymbol{u}_n^{(k)}\right) + a\left(\boldsymbol{u}_n^{(k)}, \boldsymbol{v}\right) + b\left(\boldsymbol{v}, p_n^{(k)}\right),\\ \left\langle \boldsymbol{\mathcal{F}}_p\left(\boldsymbol{u}_n^{(k)}\right), q\right\rangle_{\mathbb{Q}',\mathbb{Q}} &= b\left(\boldsymbol{u}_n^{(k)}, q\right). \end{aligned}$$

Let $n > 0$. Upon linearization with respect to $\boldsymbol{\chi}$, the tangent system for N2 is initiated with $\boldsymbol{\chi}_n^{(0)} = \boldsymbol{\chi}_{n-1}$ at t_n and is given by:

Given $\boldsymbol{\chi}_n^{(k)}$, find the step length $\delta\boldsymbol{\chi}_n^{(k+1)} = \left(\delta\boldsymbol{u}_n^{(k+1)}, \delta p_n^{(k+1)}\right) \in \mathbb{V}(\boldsymbol{u}_b) \times \mathbb{Q}$ such that

$$\begin{aligned} \frac{3\mathrm{Re}}{2\Delta t} m\left(\delta\boldsymbol{u}_n^{(k+1)}, \boldsymbol{v}\right) + \mathrm{Re}\, c\left(\delta\boldsymbol{u}_n^{(k+1)}, \boldsymbol{v}; \boldsymbol{u}_n^k\right) + a\left(\delta\boldsymbol{u}_n^{(k+1)}, \boldsymbol{v}\right) + b\left(\boldsymbol{v}, \delta p_n^{(k+1)}\right) &= -\left\langle \boldsymbol{\mathcal{F}}_X\left(X_n^{(k),T}\right), \boldsymbol{v}\right\rangle_{\mathbb{V}(\mathbf{0})',\mathbb{V}(\mathbf{0})},\\ b\left(\delta\boldsymbol{u}_n^{(k+1)}, q\right) &= -\left\langle \boldsymbol{\mathcal{F}}_p\left(\boldsymbol{u}_n^k\right), q\right\rangle_{\mathbb{Q}',\mathbb{Q}}, \end{aligned} \tag{3.4}$$

for all $\boldsymbol{v} \in \mathbb{V}(\mathbf{0})$ and $q \in \mathbb{Q}$. The solution is updated as $\boldsymbol{\chi}_n^{(k+1)} = \boldsymbol{\chi}_n^{(k)} + \delta\boldsymbol{\chi}_n^{(k+1)}$. At convergence, we set $\boldsymbol{u}_n = \boldsymbol{u}_n^{(\infty)}$ and $p_n = p_n^{(\infty)}$ and proceed to the subsequent time step. In the following, we will omit the subscript n whenever it is clear from the context.

3.3. Continuation strategy for larger Reynolds numbers

Given the sensitivity of the Newton algorithm and its variants to the initial guess and problem nonlinearity, increasing the Reynolds number in test case 2 results in a convergence loss due to a substantial deviation of the initial guess from the solution. The local convergence limitations of the method restrict the simulation of high Reynolds regimes, as discussed in 4.2.2. To achieve global convergence for medium and large Reynolds numbers, which requires an initial solution close to the desired solution, we rely on a continuation approach. The key idea is to proceed with a smooth initiation of the algorithm at the initial time step. This approach involves starting with a zero Reynolds number and incrementally increasing it until reaching the target value of Re. This differs from starting at the desired Re and decreasing the step length while maintaining the original direction of the Newton algorithm. In the subsequent time steps, the solution obtained in the current time iteration typically serves as a reliable initial guess.

To enhance the convergence by continuation [63], the original problem is encapsulated within a continuous flow represented by a differentiable solution path along the Reynolds number. In fact, Re is now treated as a continuous independent parameter, analogous to the time variable in evolutionary problems. Given a desired value Re, we intro-

Algorithm 1 Globalized Strategy for Efficiently Attaining High Re

set the target value Re
initialize the continuation parameter $\xi_0 = 0$
set the Newton tolerance $\epsilon = 10^{-9}$
$m = 0$: initialize with a computed solution $\boldsymbol{\chi}_0 = \boldsymbol{\chi}_{-1}$ of $\mathcal{F}(0;\boldsymbol{\chi}(0)) = \mathbf{0}$
$m \geqslant 0$: let $\xi_m, \boldsymbol{\chi}_m$ be known
while $\xi_m <$ Re **do**
    update $\xi_{m+1} = \min(\mathrm{Re}, \xi_m + \Delta\xi)$
    perform a BDF2 predictor step by solving

$$\left\langle D_{\boldsymbol{\chi}}\mathcal{F}(\xi_m;\boldsymbol{\chi}_m)\cdot\left[\tilde{\boldsymbol{\chi}}_{m+1}^{(0)} - 4\boldsymbol{\chi}_m + \boldsymbol{\chi}_{m-1}\right], \boldsymbol{V}\right\rangle = -2\Delta\xi\left\langle\frac{\partial\mathcal{F}}{\partial\xi}(\xi_m;\boldsymbol{\chi}_m), \boldsymbol{V}\right\rangle, \quad \forall \boldsymbol{V} \in \mathbb{V}(\mathbf{0})\times\mathbb{Q}$$

    initialize Newton residual $\epsilon_k = 2\epsilon$
    perform a correction step by N3 (analogously for N2 and N5):
    for $k = 1, \ldots$ **do**
        find the increment $\delta\tilde{\boldsymbol{\chi}}_{m+1}^{(k+1/2)}$ s.t.

$$\left\langle D_{\boldsymbol{\chi}}\mathcal{F}(\xi_{m+1};\tilde{\boldsymbol{\chi}}_{m+1}^{(k)})\cdot\left[\delta\tilde{\boldsymbol{\chi}}_{m+1}^{(k+1/2)}\right], \boldsymbol{V}\right\rangle = -\frac{1}{2}\left\langle\mathcal{F}(\xi_{m+1};\tilde{\boldsymbol{\chi}}_{m+1}^{(k)}), \boldsymbol{V}\right\rangle, \ \forall \boldsymbol{V} \in \mathbb{V}(\mathbf{0})\times\mathbb{Q}$$

        compute $\tilde{\boldsymbol{\chi}}_{m+1}^{(k+1/2)} = \tilde{\boldsymbol{\chi}}_{m+1}^{(k)} + \delta\tilde{\boldsymbol{\chi}}_{m+1}^{(k+1/2)}$
        find the increment $\delta\tilde{\boldsymbol{\chi}}_{m+1}^{(k+1)}$ s.t.

$$\left\langle D_{\boldsymbol{\chi}}\mathcal{F}(\xi_{m+1};\tilde{\boldsymbol{\chi}}_{m+1}^{(k+1/2)})\cdot\left[\delta\tilde{\boldsymbol{\chi}}_{m+1}^{(k+1)}\right], \boldsymbol{V}\right\rangle = -\left\langle\mathcal{F}(\xi_{m+1};\tilde{\boldsymbol{\chi}}_{m+1}^{(k)}), \boldsymbol{V}\right\rangle, \ \forall \boldsymbol{V} \in \mathbb{V}(\mathbf{0})\times\mathbb{Q}$$

        update $\tilde{\boldsymbol{\chi}}_{m+1}^{(k+1)} = \tilde{\boldsymbol{\chi}}_{m+1}^{(k)} + \delta\tilde{\boldsymbol{\chi}}_{m+1}^{(k+1)}$
        compute the Newton residual ϵ_{k+1}
        if $\varepsilon_k < \epsilon$ **then**
            set $\boldsymbol{\chi}_{m+1} = \tilde{\boldsymbol{\chi}}_{m+1}^{(k+1)}$
            break
        end if
    end for
    set solution $(\boldsymbol{u}^\star, p^\star) = \boldsymbol{\chi}_{m+1}$ of $\mathcal{F}(\mathrm{Re};\boldsymbol{\chi}(\mathrm{Re})) = \mathbf{0}$
end while

duce a continuous independent variable $\xi \in [0, \mathrm{Re}]$, where $\xi = 0$ corresponds to $\mathrm{Re} = 0$ and $\xi = \mathrm{Re}$ corresponds to the target value for which we seek a numerical solution. Therefore, the original steady-state or time-discretized problem, succinctly expressed as $\mathcal{F}(\mathbf{u}, p) = \mathbf{0}$ (3.3) with unknowns $\mathbf{u}^\star$, $p^\star$, is reconsidered as a family of problems:

$$\mathcal{F}(\xi; \boldsymbol{\chi}(\xi)) = \mathcal{F}(\xi; \mathbf{u}(\xi), p(\xi)) = \mathbf{0}, \tag{3.5}$$

parameterized by a real scalar field $\xi \in [0, \mathrm{Re}]$. When $\xi = 0$, a Stokes problem holds with known solution $\boldsymbol{\chi}(0)$. For $\xi = \mathrm{Re}$, the problem involves the unknown solution $\boldsymbol{\chi}(\mathrm{Re})$, which corresponds to the desired solution $\mathbf{u}^\star$, $p^\star$. The continuation method determines a pathway from the known solution to the unknown solution. The set $\{\boldsymbol{\chi}(\xi) \,\mathrm{s.\,t.}\, 0 \leqslant \xi \leqslant \mathrm{Re}\}$ can be seen as a curve in $\mathbb{V}(\boldsymbol{u}_b) \times \mathbb{Q}$ from $\boldsymbol{\chi}(0)$ to $\boldsymbol{\chi}(\mathrm{Re})$ parametrized by the continuation parameter ξ. Note that we can also start from another small value of ξ, which is not necessarily zero but allows for an easily calculated solution.

We assume that $\xi \to \boldsymbol{\chi}(\xi)$ and $\boldsymbol{\mathcal{F}}(\xi; \boldsymbol{\chi}(\xi))$ are continuously differentiable and that the Jacobian matrix is nonsingular for all $\boldsymbol{\chi}$ and ξ. Furthermore, a continuously differentiable solution trajectory $\boldsymbol{\chi}(\xi)$ exists, as established in [63, Theorem 5.2.1], which addressed an inexact version to approximate the Jacobian matrix. We obtain by differentiating (3.5) with respect to ξ:

$$D_{\boldsymbol{\chi}} \mathcal{F}(\xi; \boldsymbol{\chi}(\xi)) \cdot \frac{d\boldsymbol{\chi}(\xi)}{d\xi} = -\frac{\partial \mathcal{F}}{\partial \xi}(\xi; \boldsymbol{\chi}).$$

Hence, for any values of the parameter Re, we introduce a two-step iterative BDF2-Newton strategy. In the first step, a prediction is generated using a BDF2 scheme, treating the variable ξ as if it corresponds to a pseudo-time. The prediction is then followed by a correction step applied using Newton's method or any high-order variant. For simplicity, we assume a constant step size $\Delta\xi$. For ease of notation, the subscript m refers only to the continuation step. The strategy involves finding a sequence of solutions $\boldsymbol{\chi}_m \approx \boldsymbol{\chi}(\xi_m)$ for a discrete sequence of the continuation parameter $\xi_0 = 0 < \xi_1 < \cdots < \xi_M = \mathrm{Re}$, such that $\xi_{m+1} = \xi_m + \Delta\xi$. Let $< ., . >$ denote the duality product induced by the L^2 pivot space. The approach is outlined in Algorithm 1, employing the N3 variant, our preferred method, as will be demonstrated later through the numerical experiments. Note that an adaptive step $\delta\xi_m$ could also be considered, but was not addressed in this work; interested readers are referred to, for example, [64].

4. Numerical experiments

Computations were carried out on the Almesbar High-Performance Computing cluster, which consists of 204 CPU nodes. Unless specified otherwise, we used 52 computing cores, with each node equipped with 2x Intel Xeon Gold 6230R 26-Core 2.1GHz CPUs (Cascade Lake microarchitecture, Q1 2020). Parallel processing was implemented

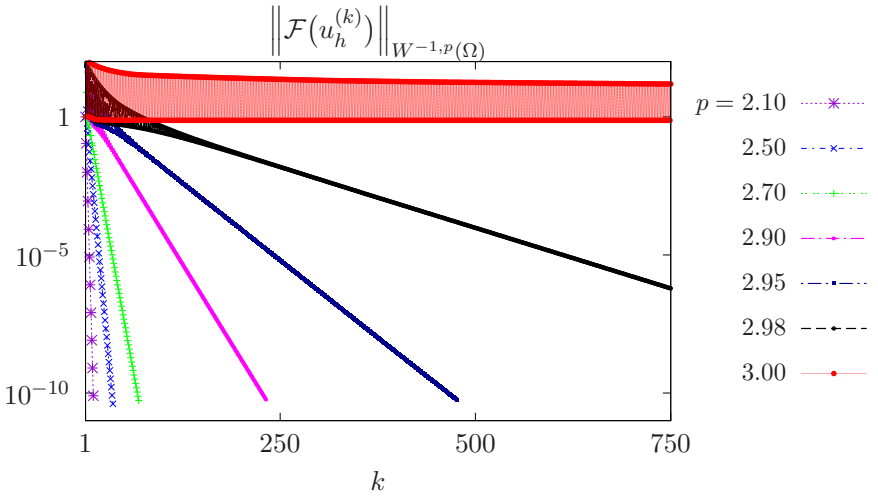

Figure 1: Convergence of the residual for fixed-point iteration as a function of iteration number, presented on a semi-logarithmic scale.

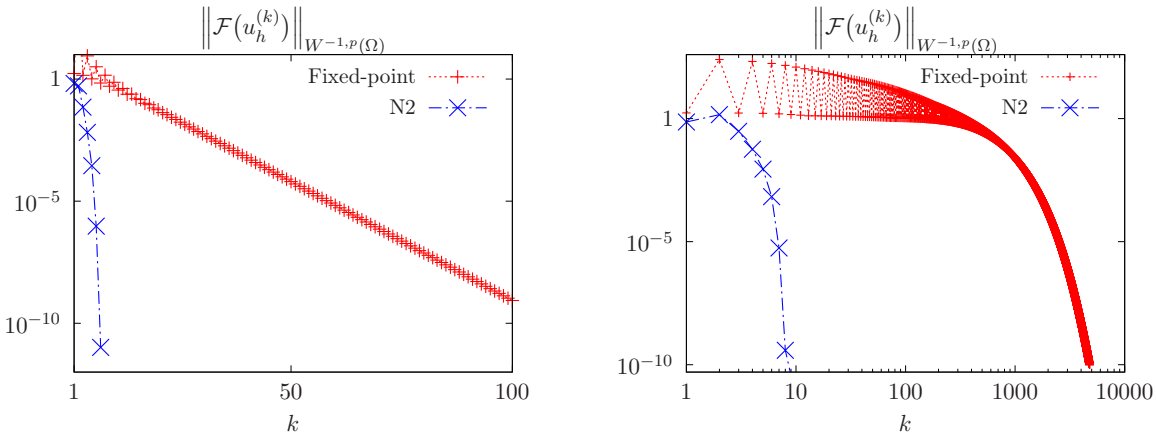

Figure 2: Convergence of the residual against the iteration number for both fixed-point and N2 methods. (Left) $p = 2.8$ and (Right) $p = 2.995$.

via the MPI Message Passing Interface[1], with MUMPS utilized for factorization and serving as the direct solver on distributed-memory systems. Numerical strategies were implemented using Freefem++[2] and Rheolef [3]. Non-uniform meshes were created with Gmsh[4]. The figures were produced using Paraview[5] and Gnuplot[6].

4.1. Example 1: Stationary *p*-Laplacian problem

This test case evaluates the performance of Newton's variants in comparison to the quadratic standard Newton's strategy. It aims to provide insights into the approach to be subsequently employed for solving the time-dependent Navier-Stokes problem in both 2D and 3D.

We first solve the p-Laplacian problem using the fixed-point algorithm with $p = 2.1$, analyzing the convergence while increasing the nonlinearity level. A structured mesh with a mesh size of $h = 2 \times 10^{-2}$ is used. The evolution of the residual $\left\|\mathcal{F}(u_h^{(k)})\right\|_{W^{-1,p}(\Omega)}$ with respect to the iteration number k is plotted, where the subscript h refers to the discretized solution. Results are shown in Fig. 1, displaying the expected linear convergence, which is rapid when the problem is close to the linear Poisson problem. However, the linear convergence becomes progressively slow to achieve as we increase the parameter p. For $p = 3$, computations result in an oscillatory solution that fails to converge. This behavior is consistent across different mesh sizes.

We now set $p = 2.8$ and $p = 2.995$ and use $\mathbb{P}_1$ Lagrange elements with a mesh size $h_1 = 2.5 \times 10^{-2}$. Fig. 2 shows the residual's convergence with respect to the iteration number. The N2 method outperforms the fixed-point algorithm as expected, especially for $p = 2.995$, where N2 converges in 10 iterations, while the fixed-point method requires 4780 iterations. This improved efficiency is evident in the computational time, with N2 using 0.62 seconds of CPU time, significantly less than the fixed-point algorithm's 1 minute and 55.75 seconds.

To assess the convergence with an increasing nonlinearity controlled by p, we calculate the rate of convergence as:

$$\mathrm{ROC} = \frac{\log\left(\dfrac{\|\mathcal{F}(u_h^{(k)})\|_{W^{-1,p}(\Omega)}}{\|\mathcal{F}(u_h^{(k-1)})\|_{W^{-1,p}(\Omega)}}\right)}{\log\left(\dfrac{\|\mathcal{F}(u_h^{(k-1)})\|_{W^{-1,p}(\Omega)}}{\|\mathcal{F}(u_h^{(k-2)})\|_{W^{-1,p}(\Omega)}}\right)}, \quad \text{for} \quad k \geqslant 2.$$

[1] http://www.mpich.org, version 4.2.2, 2024-07-03
[2] https://packages.debian.org/sid/freefem, version 3.5.8-7, accessed 2024-08-22
[3] https://packages.debian.org/unstable/rheolef, version 7.2-3, accessed 2024-08-24
[4] https://gmsh.info/, version 4.13.1, 2024-05-24
[5] https://www.paraview.org/, version 5.13, 2024-08-09
[6] http://www.gnuplot.info/, version 6.0.1, 2024-05-10

	$p = 2.3$	3.0	3.5	4.0	4.5	12.0
$k = 2$	2.50	0.37	0.13	0.07	0.04	0.01
3	**1.98**	1.03	1.00	1.00	1.00	11.91
4	**2.02**	1.13	1.00	1.00	1.00	0.08
5		1.49	1.00	1.00	1.00	1.00
6		**1.92**	1.00	1.00	1.00	1.00
7		**2.00**	1.02	1.00	1.00	1.00
11			**1.94**	1.00	1.00	1.00
12			**2.00**	1.00	1.00	1.00
20				**1.95**	1.00	1.00
21				**2.00**	1.00	1.00
27					**1.91**	1.00
28					**2.00**	1.00
2547						**1.86**
2548						**1.99**

Table 1: Rates of convergence of the residual relative to the iteration number k for different values of p using the N2 method.

Table 1 shows that achieving higher accuracy with increasing nonlinearity necessitates more iterations. The residual decreases gradually with increasing p, eventually reaching a threshold beyond which quadratic convergence is achieved, with ROC clearly approaching 2.

Hereafter, we assess the efficiency of high-order Newton variants and numerically analyze the convergence by comparing them with the fixed-point and N2 methods. We consider a mesh size of $h = 2.5 \times 10^{-2}$ and $\mathbb{P}_1$ Lagrange polynomials. Fig. 3 shows the convergence of residuals for $p = 2.995$, demonstrating rapid and comparable convergence for the Newton variants, while the fixed-point method requires more than $10,000$ iterations.

Given the fixed-point algorithm's convergence failure for $p \geqslant 3$, we conduct a comparative study among various Newton methods for nonlinearity coefficients $p = 4$ and $p = 5.5$. The convergence of residuals, shown in Fig. 4, highlights the superior performance of the N3 and N5 methods. During an initial phase, the residual's decrease is slow due to the computed solution being distant from the exact one. The standard Newton method exhibits quadratic convergence when the approximated solution is within a specific neighborhood. The N3 variant improves convergence with its higher order, with the N5 variant showing further enhancement. Indeed, the N5 variant converges in 20 iterations for $p = 5.5$, while the N2 method requires 42 iterations to achieve the desired accuracy.

To assess the computational cost, we solve the p-Laplacian problem by gradually increasing p. The computational CPU time is presented in Table 2, with "NC" indicating cases where the method fails to converge. Due to local convergence properties, we initially performed one fixed-point iteration to initialize the Newton methods. These methods

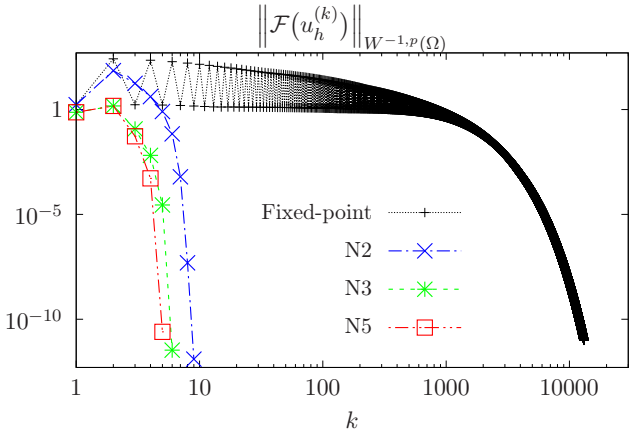

Figure 3: Residual convergence of the fixed-point and Newton's variants versus the iteration number for $p = 2.995$. The logarithmic scale is used.

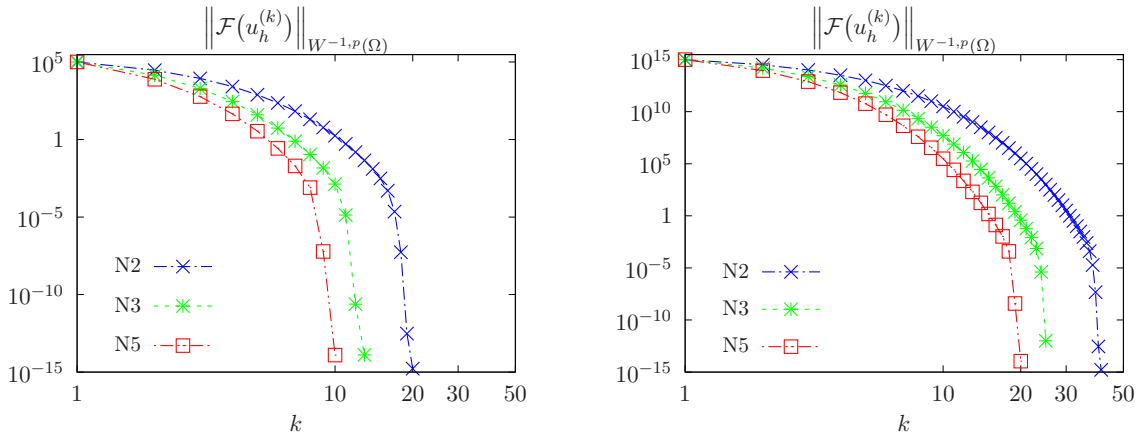

Figure 4: Convergence of the residuals with respect to iteration number in the logarithmic scale for $p = 4$ (Left) and $p = 5$ (Right).

can fail to converge if the initial approximation is significantly far from the solution. The N5 variant demonstrates slightly better performance in terms of computing time compared to both the N2 and N3 methods. When the nonlinearities are stronger, the additional computational cost associated with the extra Jacobian evaluation in the N5 method is effectively balanced by its fifth-order convergence. However, the N3 variant fails to converge for $p > 6$. This issue will be addressed more effectively in Example 2 using a globalized version. In the following, we will more thoroughly evaluate the performances of the N3 and N5 variants relative to the N2 method in both 2D and 3D cases for more complex problems.

CPU time [seconds]				
p	Fixed-point method	N2 method	N3 variant	N5 variant
2.998	379.92	0.75	0.78	0.67
4.000	NC	1.35	1.38	1.28
5.000	NC	2.52	2.40	2.33
5.500	NC	3.19	3.14	2.89
7.000	NC	6.15	NC	5.67
8.000	NC	9.29	NC	8.06

Table 2: Comparison of CPU computation times for the p-Laplacian problem for different values of p.

4.2. Example 2: Lid-driven cavity benchmark

In this test case, we examine the time-dependent lid-driven cavity problem, a benchmark for viscous incompressible flow. At lower Reynolds numbers, the resulting fluid flow shows nearly symmetric behavior near the centerline, forming two vortices in the corners. As Reynolds numbers increase, the nonlinearity in the system becomes more prominent. The central vortex begins moving towards the downstream corner before eventually returning to the center as the Reynolds number continues to rise. In high Reynolds regimes, multiple secondary and tertiary vortices emerge, significantly altering the overall flow structure. Several studies have explored 2D flow patterns at lower Reynolds numbers, serving as a reference for numerical validation. For example, comparisons of streamline topology in two-dimensional flow are found in [65, 66, 67, 68, 69, 70, 71].

We first focus on the steady-state solution of a 2D fluid confined within a cavity $\Omega = (0, 1)^2 \in \mathbb{R}^2$. A tangential velocity $\boldsymbol{u}_b = (1, 0)^T$ is prescribed on the upper boundary of the fluid, while homogeneous Dirichlet conditions are prescribed on the remaining boundaries. The streamline function and the vorticity field offer insights into the fluid flow patterns and rotational behavior. The vorticity, denoted by ω, quantifies local fluid rotation and is defined in 2D

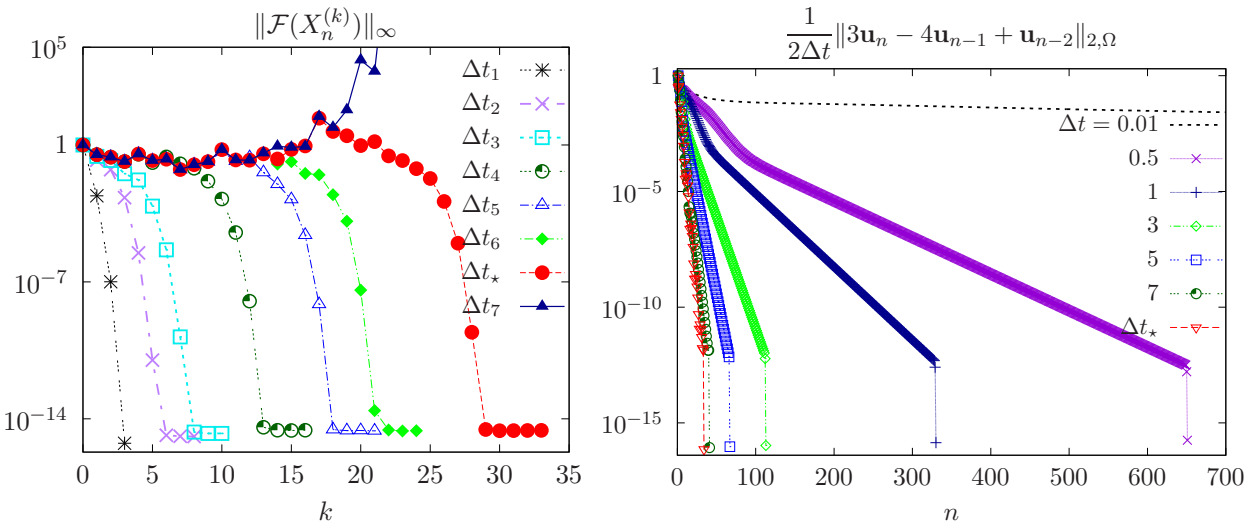

Figure 5: Convergence of the N2 method against the time step: (Left) Residual variaton versus the iteration number k for the first time iteration $n = 0$, with $\Delta t_1 = 10^{-2}$, $\Delta t_2 = 1$, $\Delta t_3 = 5$, $\Delta t_4 = 16.1$, $\Delta t_5 = 16.1084$, $\Delta t_6 = 16.1084103$, $\Delta t_7 = 16.1084105$ and $\Delta t_\star = 16.1084104$. (Right) Convergence of the global residual until the steady-state against the time iteration n. The y-semi-logarithmic scale is used.

Δt	0.5	1.0	3.0	5.0	7.0	$\Delta t_\star$
K_{tot}	650	329	112	66	40	33
CPU time	123 m 4 s	59 m 44 s	19 m 49 s	13 m 18 s	9 m 3 s	8 m 39 s
$-\psi_{\min}$	0.1100	0.1100	0.1100	0.1100	0.1100	0.1100
$-\omega$	1.9313	1.9313	1.9313	1.9313	1.9313	1.9313
(x, y)	(0.53, 0.57)	(0.53, 0.57)	(0.53, 0.57)	(0.53, 0.57)	(0.53, 0.57)	(0.53, 0.57)

Table 3: Computational time to reach steady-state for Re = 950 with various time steps Δt. K_{tot} denotes the total number of iterations using the standard Newton's method.

as the curl of the velocity $\omega = \nabla \wedge \boldsymbol{u} = \partial_x v - \partial_y u$, where $\mathbf{u} = (u, v)$. The streamline function, denoted by ψ, is a scalar field that describes fluid flow paths and satisfies $-\Delta\psi = \nabla \wedge \mathbf{u}$ in Ω, with $\psi = 0$ on the boundary $\partial\Omega$.

4.2.1. Assessment of the improved Newton variants

Set $h = 2 \times 10^{-2}$ and Re = 950, and examine the convergence behavior of the N2 method with respect to the time step Δt. First, we set $\Delta t = 10^{-2}$ and present the convergence of the residual with respect to Newton's iteration number k for the first time step. Convergence is achieved in 3 iterations, and quadratic convergence is observed. Subsequently, we systematically increase the time step and show the evolution of the residual in Fig. 5. Convergence is achieved for larger time steps until the residual falls below a certain threshold. Beyond a maximum threshold value $\Delta t_\star = 16.1084104$ (obtained by dichotomy with $6 \times 10^{-7}\%$ precision), the algorithm fails to converge. For time steps ranging from $\Delta t = 10^{-2}$ to $\Delta t_\star$, we assess convergence by evaluating

$$\frac{1}{2\Delta t} \|3\mathbf{u}_n - 4\mathbf{u}_{n-1} + \mathbf{u}_{n-2}\|_{2,\Omega} ,$$

which should converge to zero as the steady state is reached. This convergence is illustrated in Fig. 5 (right).

Table 3 shows the total number of Newton iterations for various time steps, along with the total CPU time required for sequential computation runs. Additionally, we assess convergence for different time steps to the same values of minimum streamline $\psi_{\min}$, the position of the primary vortex (x, y), as well as vorticity ω.

Similarly, we perform a numerical study of the convergence for the N5 and N3 variants with respect to Δt, keeping the parameters $h = 2 \times 10^{-2}$ and Re = 950. The convergence plots of the residuals are shown in Fig. 6 and Fig. 7 for

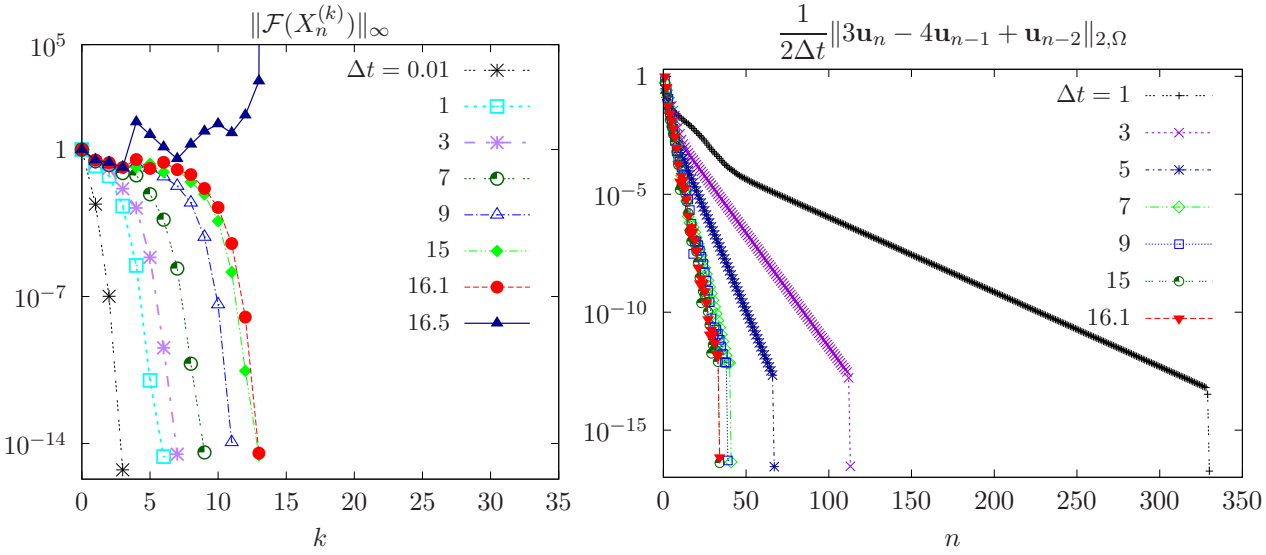

Figure 6: Convergence of the N5 Method with respect to the time step: (Left) Variation of residual versus iteration count k for $n = 0$. (Right) Convergence of the global residual to steady-state versus time iteration n. The y-axis is shown on a semi-logarithmic scale.

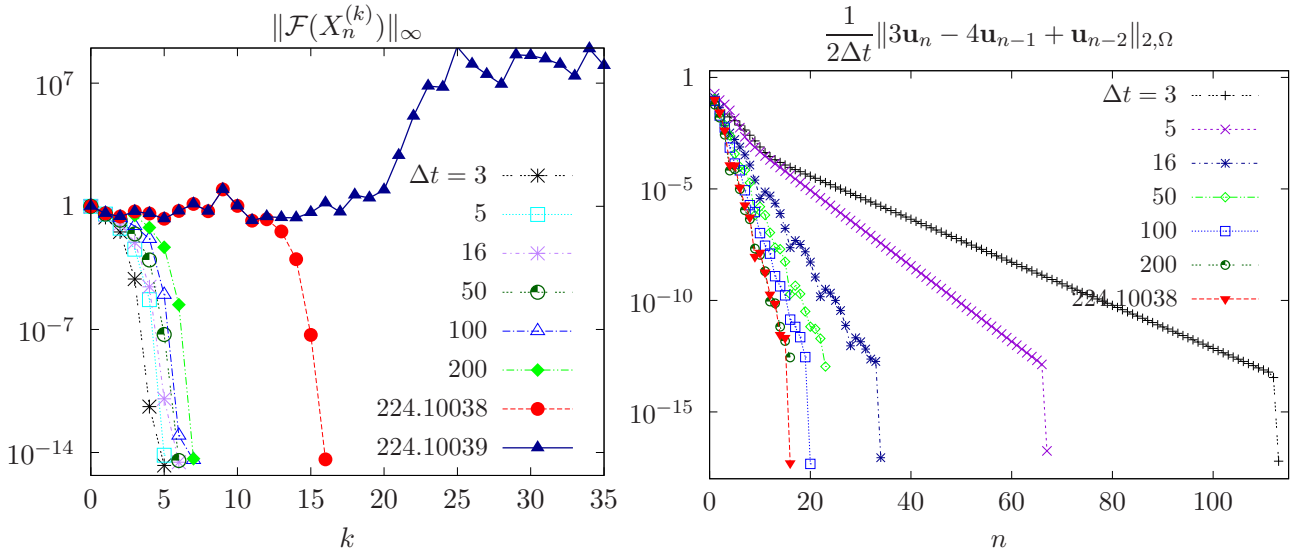

Figure 7: Convergence of the N3 method with respect to the time step: (Left) Variation of the residual as a function of the iteration count k for $n = 0$. (Right) Global residual convergence to steady-state as a function of time iteration n. The y-axis uses a semi-logarithmic scale.

the N5 and N3 variants, respectively. Notably, the maximal threshold value of Δt achieved with the N5 variant is close to $\Delta t_\star$ obtained by the standard N2 method. However, the N3 variant demonstrates significantly better efficiency, allowing for much larger time steps, reaching up to $\Delta t \approx 224.1$, as shown in Fig. 7 (right).

We proceed with a convergence study, paying particular attention to the computational cost. Steady-state solutions have been computed in distributed and parallel runs using 6 processors for various Newton variants. A sequence of successively refined mesh sizes was employed, and we present the values of the minimum streamline $\psi_{\min}$ corresponding to the primary main vortex. The results in Table 4 reveal convergence of $\psi_{\min}$ for the three variants. Most importantly, the N3 variant shows significant savings in terms of computational cost, achieving approximately a 50% reduction in CPU time compared to N2 and a 60% reduction compared to N5. In what follows, the N3 variant is our preferred strategy and will be employed unless specified otherwise.

	N2			N5			N3		
$1/h$	K_{tot}	$-\psi_{\min}$	CPU	K_{tot}	$-\psi_{\min}$	CPU	K_{tot}	$-\psi_{\min}$	CPU
25	34	1.7983	2 m 43 s	34	1.7983	3 m 4 s	19	1.7983	1 m 35 s
50	33	1.9313	1 m 37 s	33	1.9313	1 m 57 s	15	1.9313	0 m 49 s
100	35	2.0028	13 m 11 s	35	2.0028	11 m 3 s	16	2.0028	4 m 37 s
150	35	2.0270	19 m 57 s	33	2.0270	26 m 55 s	15	2.0270	11 m 34 s

Table 4: Comparison of various Newton variants in terms of computational time, total Newton iterations required to reach steady state, and the minimal stream values characterizing the primary vortex across different mesh sizes.

4.2.2. Strategy by continuation in case of large Reynolds numbers

Consider a mesh size of $h = 5 \times 10^{-3}$. In the case of a transient problem, the solution at a particular time step can provide a suitable initial guess for the subsequent time step. However, for large values of Re, this is not generally fulfilled at the initial time step or when solving the steady-state problem. We solve the steady-state problem for Re = 500 and gradually increase Re. As a result, the implicit algorithm becomes progressively less well-initialized, leading to difficulties in achieving convergence, and the algorithm fails to converge when it reaches a threshold value of Re. The variations in the global residual are shown in Fig. 8, revealing a threshold value around $\mathrm{Re}_\star = 852.27162$ within a margin of 10^{-5} obtained by dichotomy.

We employ the aforementioned continuation strategy to address the local convergence issues arising from insufficient initialization for large Re, as detailed in Section 3.3 and outlined in Algorithm 1. We choose a significantly large Reynolds number, Re = 50′000, which surpasses the numerically determined stability threshold value $\mathrm{Re}_\star$. We provide in Fig. 9 the convergence of residuals while gradually increasing ξ towards the target value Re. The results demonstrate the stability of computations, showing the convergence to the steady-state solution at this high Reynolds number.

4.2.3. Grid convergence - local and global quantities

We proceed to validate the N3 variant in terms of spatial convergence. Following [67], we search for the steady-state solution for Re = 1000. We analyze the characteristics of both the primary and secondary vortices: the maximum and minimum stream function $\psi_{\min}$ and $\psi_{\max}$, as well as the corresponding vorticity ω and position (x, y). Computed

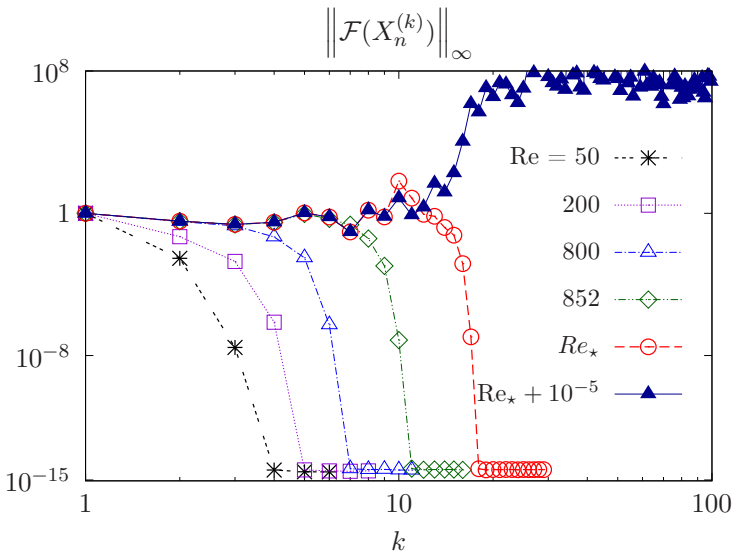

Figure 8: Residual convergence with respect to the Reynolds number of the N3 variant for the steady-state problem. The logarithmic scale is used.

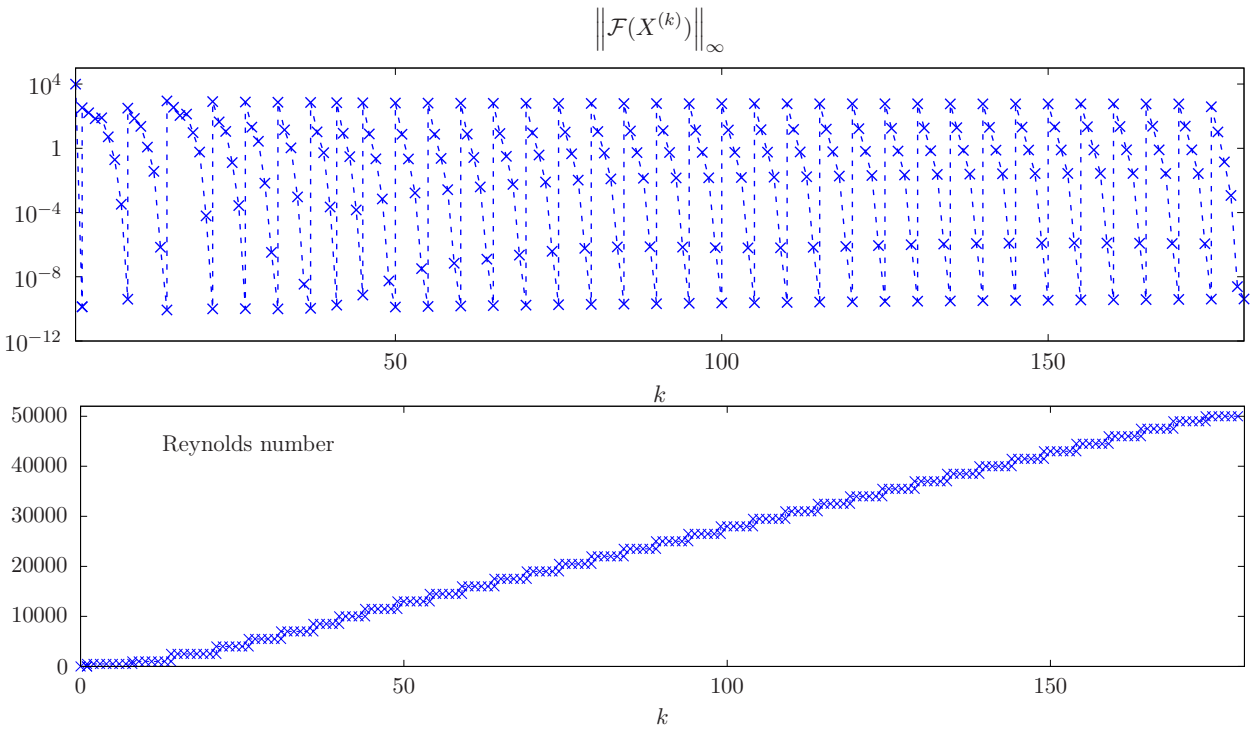

Figure 9: Steady-state simulation for Re = 50′000 using $h = 5 \times 10^{-3}$ and $\mathbb{P}_2/\mathbb{P}_1$ Lagrange elements for $(\boldsymbol{u}, p)$. (Top) Residual convergence plotted against the number of iterations on a semi-logarithmic scale in the y-axis. (Bottom) Efficient setting of Re via continuation.

values are reported for successively refined meshes in Table 5, with reference values in the last row obtained by numerical continuation. A generally satisfactory agreement is observed as h decreases. In the following, we choose $h = 5 \times 10^{-3}$ unless otherwise specified.

In addition to these local quantities, we also consider some global quantities studied in the literature [67]. We define the total kinetic energy $\mathscr{E}$, enstrophy $\mathscr{Z}$, and palinstrophy $\mathscr{P}$ as follows:

$$\mathscr{E} = \frac{1}{2}\int_\Omega \|\boldsymbol{u}\|^2 \,\mathrm{d}\boldsymbol{x}, \quad \mathscr{Z} = \frac{1}{2}\int_\Omega \|\omega\|^2 \,\mathrm{d}\boldsymbol{x}, \quad \mathscr{P} = \frac{1}{2}\int_\Omega \|\boldsymbol{\nabla}\omega\|^2 \,\mathrm{d}\boldsymbol{x}.$$

These quantities result in constant values as the solution is stationary. However, achieving spatial convergence for enstrophy and palinstrophy is impossible due to the singularity at the corner [72]. Indeed, enstrophy exhibits behavior

	Primary vortex				Secondary vortex			
h	$-\psi_{\min}$	$-\omega$	x	y	$\psi_{\max}$	ω	x	y
1/25	0.1016	1.7838	0.5400	0.5800	1.1615×10^{-3}	0.7903	0.8800	0.1200
1/50	0.1099	1.9198	0.5300	0.5700	1.4276×10^{-3}	0.8946	0.8700	0.1100
1/100	0.1144	1.9930	0.5300	0.5650	1.5771×10^{-3}	1.0198	0.8650	0.1100
1/200	0.1166	2.0303	0.5325	0.5650	1.6531×10^{-3}	1.0760	0.8650	0.1125
1/300	0.1174	2.0428	0.5317	0.5650	1.6785×10^{-3}	1.0745	0.8650	0.1117
1/400	0.1178	2.0489	0.5313	0.5650	1.6912×10^{-3}	1.0896	0.8650	0.1120
Numerical continuation:	**0.1189**	**2.0676**	**0.5308**	**0.5650**	$\mathbf{1.7290 \times 10^{-3}}$	**1.1037**	**0.8650**	**0.1129**

Table 5: Steady-state flow for Re = 1000. Spatial convergence analysis of the primary and secondary bottom-right vortices, including the minimum of the stream function, corresponding vorticity ω, and the location $\boldsymbol{x}(x, y)$.

Method	h	$\mathscr{E}$	$\mathscr{Z}$	$\mathscr{P}$
Present	$1/25$	2.3088×10^{-2}	4.7568	6.5512×10^{3}
Present	$1/50$	2.2796×10^{-2}	4.8246	8.2175×10^{3}
Present	$1/100$	2.2769×10^{-2}	4.8301	8.6335×10^{3}
Present	$1/200$	2.2767×10^{-2}	4.8304	8.7155×10^{3}
Present	$1/400$	2.2607×10^{-2}	4.8304	8.7618×10^{3}
Numerical continuation	$--$	$\mathbf{2.2767 \times 10^{-2}}$	**4.8305**	$\mathbf{8.7811 \times 10^{3}}$
Reference [67]	$1/512$	2.2767×10^{-2}	4.82430	8.2699×10^{3}

Table 6: Convergence history of global output values for the regularized cavity problem with Re = 1000, alongside a comparison with the finite differences method using a multigrid solver in [67].

like $1/r$ near the singular corners, causing enstrophy and palinstrophy to diverge as the mesh size approaches zero. Therefore, a smooth tangential velocity $\boldsymbol{u}(t,(x,y)) = \left(16x^2(1-x)^2, 0\right)^T$ is prescribed on the upper boundary of Ω. The spatial convergence history of the global quantities at the steady state with Re = 1000 is provided in Table 6 for successively refined meshes, showing that convergence is achieved. The results align with those obtained in [67] with fine mesh resolution.

4.2.4. *Qualitative validation*

We begin by setting Re = 1000 and run simulations until reaching a steady state. The isoline values of the stream function and vorticity contours, closely resembling those in [67] (including the maximum and minimum values), are

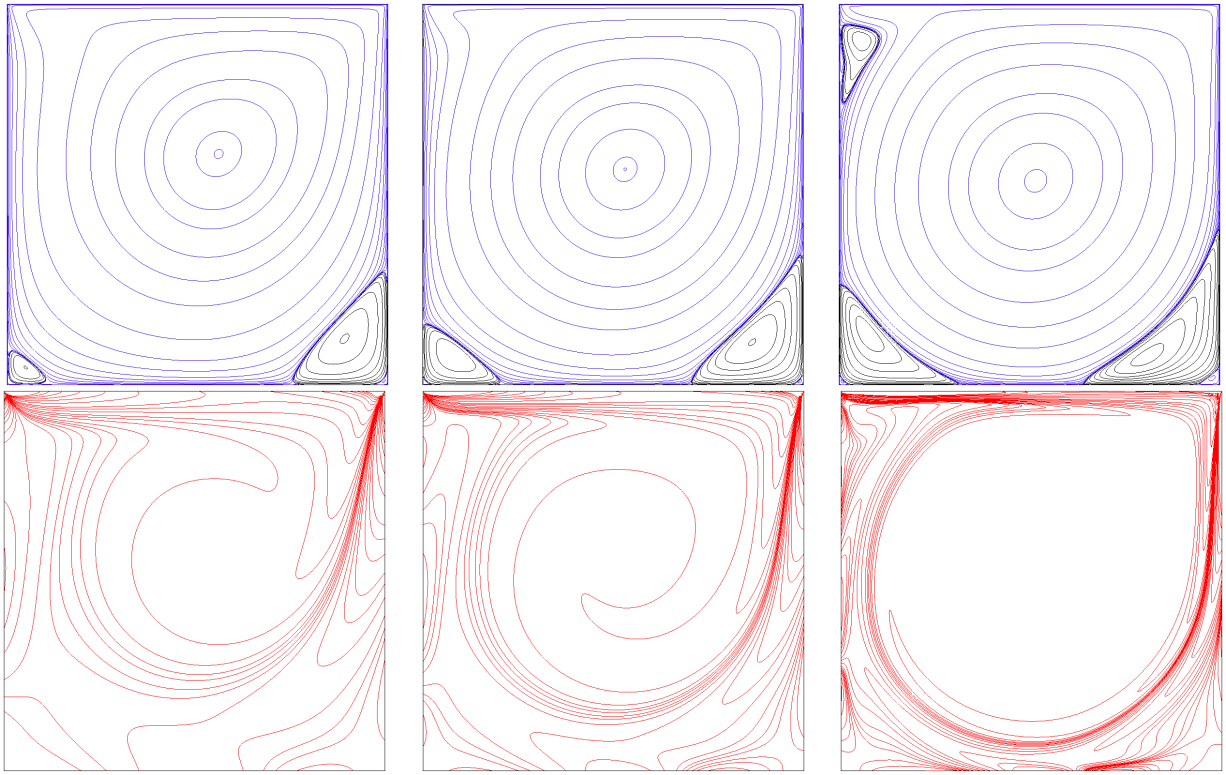

Figure 10: Steady-state solutions for Re = 400, 1000, and 5000, from left to right. Top: Streamfunction (black for $\psi > 0$, blue for $\psi \leqslant 0$). Bottom: Vorticity isocontours (red). [Color figure can be viewed online.]

Stream-function ψ					
1.6×10^{-3}	1×10^{-3}	6×10^{-4}	3×10^{-4}	1×10^{-4}	3×10^{-5}
1×10^{-5}	3×10^{-6}	1×10^{-6}	1×10^{-7}	1×10^{-8}	1×10^{-9}
1×10^{-10}	0.0	-1×10^{-10}	-1×10^{-9}	-1×10^{-8}	-1×10^{-7}
-1×10^{-6}	-3×10^{-6}	-1×10^{-5}	-3×10^{-5}	-1×10^{-4}	-3×10^{-4}
-1×10^{-3}	-3×10^{-3}	-0.01	-0.03	-0.05	-0.07
-0.09	-0.1	-0.11	-0.115	-0.1155	
Vorticity ω					
-40.0	-35.0	-30.0	-25.0	-20.0	-15.0
-10.0	-8.0	-6.0	-4.0	-3.0	-2.0
-1.0	-0.5	-0.2	0.2	0.5	1.0
2.0	3.0	4.0	6.0	8.0	10.0
15.0	20.0	25.0	30.0	35.0	40.0

Table 7: Isoline values for the streamfunction and vorticity contours in the steady-state obtained with Re = 1000, as depicted in Fig. 10.

provided in Table 7. Snapshots are presented in Fig. 10, where the solid black isolines denote positive values, and the solid blue isolines represent non-positive values. For higher Reynolds numbers, a few extra isolines will be addeded near the numerical values of $\psi_{\max}$ and $\psi_{\min}$.

Indeed, the steady-state results for Re = 1000 reveal a primary vortex accompanied by two secondary vortices in the lower corners. The primary vortex has a stream function value of 0.1166, with its center located at coordinates $(0.5325, 0.565)$, which aligns with the ranges available in existing literature, as shown in Table 8. Therein, the corre-

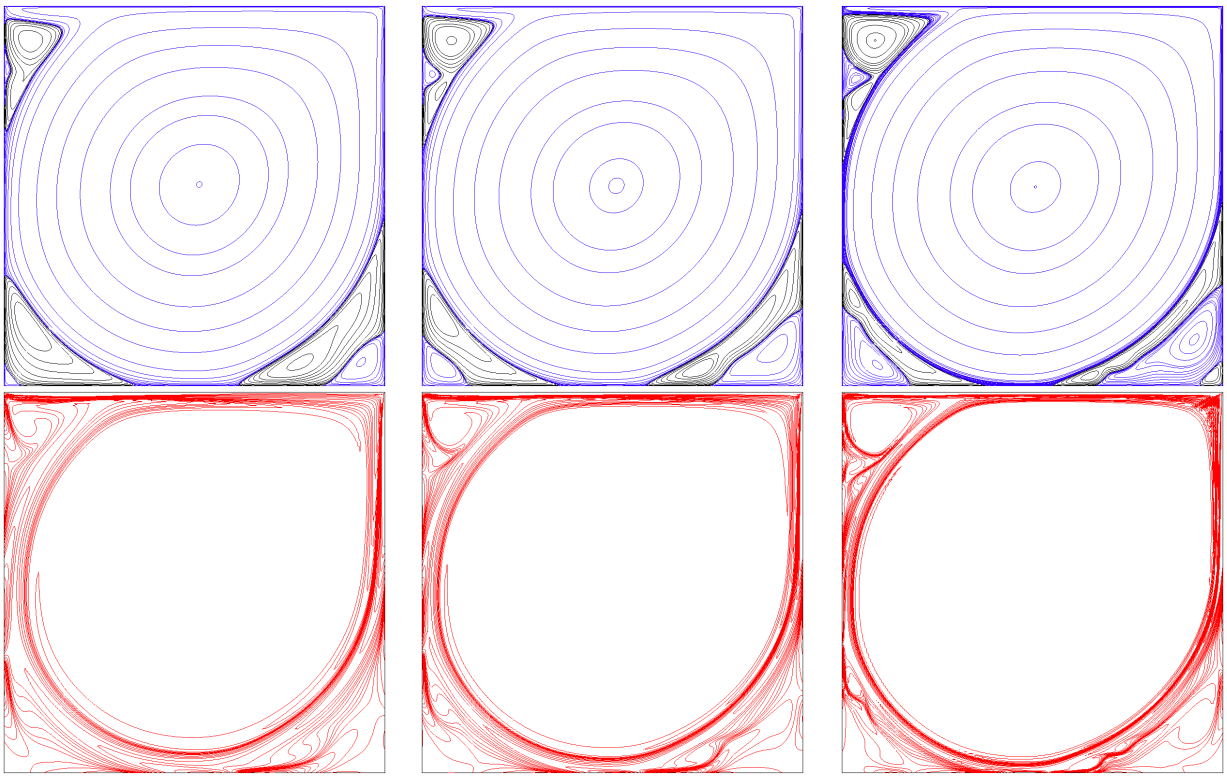

Figure 11: Steady-state solutions for Re = $10'000$, $20'000$, and $50'000$, from left to right. Top: Streamfunction (black for $\psi > 0$, blue for $\psi \leqslant 0$). Bottom: Vorticity isocontours (red). [Color figure can be viewed online.]

Reference	Discretization method	$1/h$	Primary vortex			
			$-\psi_{\min}$	$-\omega_P$	x_P	y_P
Ref. [65]	FDM	128	0.1179	2.0497	0.4687	0.5625
Ref. [66]	FDM	140	0.1160	2.0260	0.4714	0.5642
Ref. [73]	FDM	320	0.1173	–	0.4562	0.5625
Ref. [74]	CCM	160	0.1189	2.0677	0.4692	0.5652
Ref. [75]	FEM	–	0.1100	–	0.5400	0.5730
Ref. [67]	FDM	1024	0.1189	2.0674	0.4688	0.5654
Ref. [76]	FEM, DG	64	0.1210	–	0.5240	0.5600
Ref. [68]	SPH	–	0.1143	2.0569	0.5307	0.5646
Ref. [69]	FVM, GPU accelerated	601	0.1189	2.0666	0.5308	0.5657
Present	FEM	200	0.1166	2.0303	0.5325	0.5650
			Secondary vortex			
			$\psi_{\max}$	ω_{RB}	x_{RB}	y_{RB}
Ref. [65]	FDM	128	1.7510×10^{-3}	1.1547	0.1406	0.1094
Ref. [66]	FDM	140	1.7000×10^{-3}	0.9990	0.1357	0.1071
Ref. [73]	FDM	320	1.7400×10^{-3}	–	0.1375	0.1063
Ref. [74]	CCM	160	1.7297×10^{-3}	1.1098	0.1360	0.1118
Ref. [67]	FDM	1024	1.7292×10^{-3}	1.1120	0.1367	0.1123
Ref. [69]	FVM, GPU accelerated	601	1.7320×10^{-3}	1.1140	0.8636	0.1115
Present	FEM	200	1.6531×10^{-3}	1.0759	0.8650	0.1125

Table 8: Flow characteristics for Re = 1000, comparing primary and lower-right secondary vortices with reference results from the literature.

sponding discretization techniques are listed as well, using the abbreviations: (FDM) Finite Difference, (FEM) Finite Element, (FVM) Finite Volume, (SPH) Smoothed Particle Hydrodynamic, (DG) Discontinuous Galerkin, and (CCM) Chebyshev Collocation methods.

Reference	Discretization method	$1/h$	Primary vortex			
			$-\psi_{\min}$	ω_P	x_P	y_P
Ref. [65]	FDM	256	0.1190	1.8602	0.4883	0.5352
Ref. [73]	FDM	160	0.0920	–	0.4875	0.5313
Ref. [77]	FEM	256	0.1212	–	0.4844	0.5352
Ref. [67]	FDM	2048	0.1220	1.9327	0.4854	0.5352
Ref. [76]	FEM, DG	64	0.1310	–	0.5150	0.5460
Ref. [69]	FVM, GPU accelerated	601	0.1221	1.9381	0.5155	0.5355
Present	FEM	200	0.1170	1.8578	0.5150	0.5350
			Secondary vortex			
			$\psi_{\max}$	ω_{RB}	x_{RB}	y_{RB}
Ref. [65]	FDM	256	3.0348×10^{-3}	2.6635	0.1914	0.0742
Ref. [73]	FDM	160	3.1300×10^{-3}	–	0.1500	0.0813
Ref. [67]	FDM	2048	3.0706×10^{-3}	2.7244	0.1943	0.0732
Ref. [69]	FVM, GPU accelerated	601	3.0780×10^{-3}	2.7504	0.8052	0.0729
Present	FEM	200	2.9015×10^{-3}	2.6360	0.8050	0.0725

Table 9: Comparison of flow characteristics at Re = 5000 for primary and lower-right secondary vortices, against existing literature results.

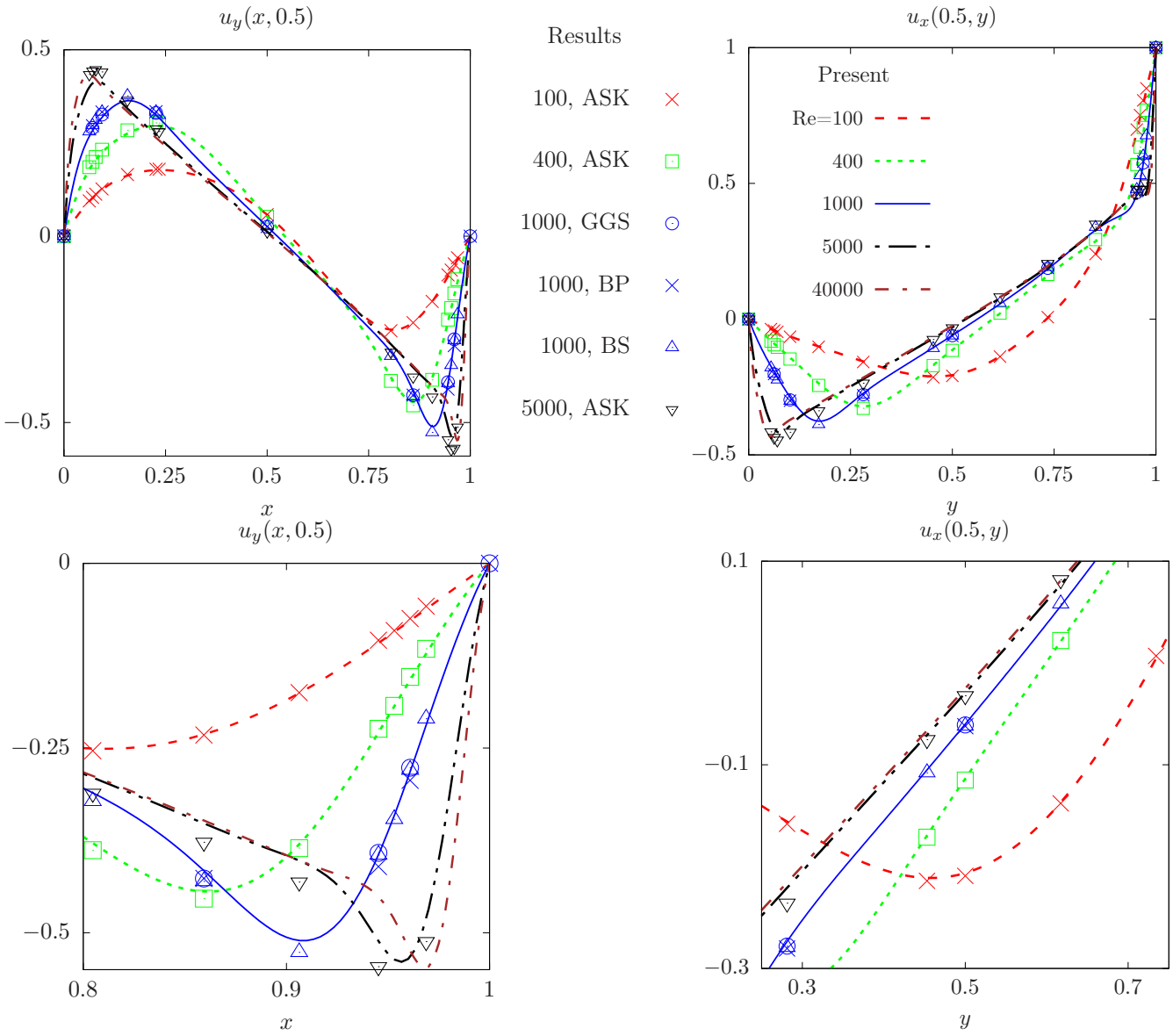

Figure 12: Velocity centerline slices: (Left) Vertical velocity along $y = 0.5$ (y-component), and (Right) Horizontal velocity along $x = 0.5$ (x-component). (Top) General view. (Bottom) Zoomed-in views near the points $(0.9, -0.25)$ and $(0.5, -0.1)$, respectively. The color coding corresponds to different Reynolds numbers (Re). Line plots represent the current calculations, while point plots display reference data from the literature: GGS [65], BS [67], ASK [69], and BP [74]. [Colour figure can be viewed online]

Similarly, Table 9 presents a comparative analysis of the local quantities for the steady flow at a higher Reynolds number Re = 5000. We include computed values from the literature, demonstrating good agreement.

Furthermore, we showcase results at high Reynolds numbers, highlighting the stability of the enhanced globalized N3 strategy up to Re = 50′000. To our knowledge, this Reynolds number has not been examined in the existing literature with similar numerical approaches. Figures 10 and 11 depict the emergence of additional vortices and patterns with increasing Re.

Lastly, to provide a qualitative and more accurate comparison of the solution profiles shown in Fig. 10 and Fig. 11, we analyze steady-state solutions across various Reynolds numbers, from Re = 100 up to Re = 5000. We overlay the profiles of the vertical and horizontal velocity fields obtained along the horizontal and vertical centerlines through

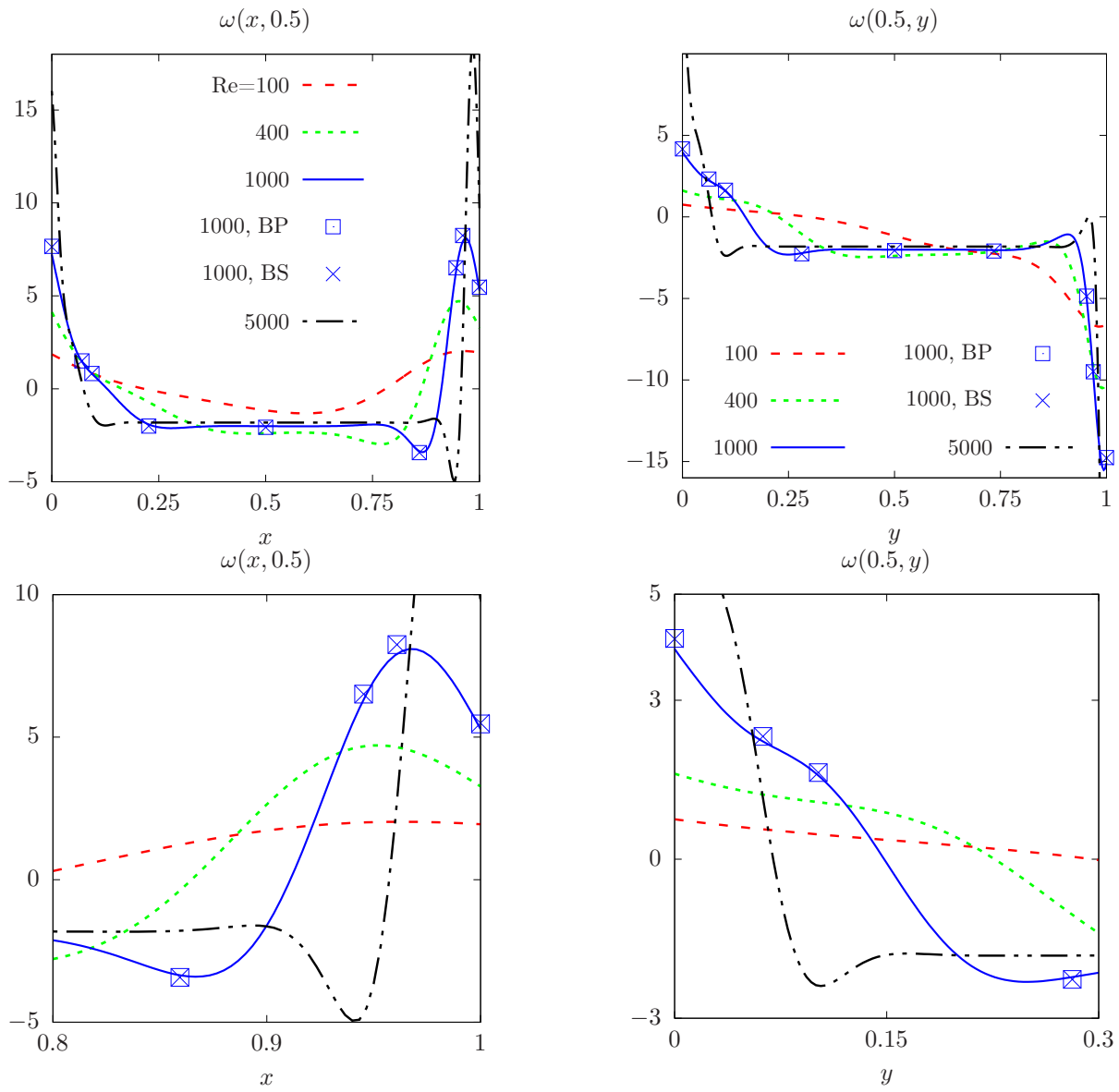

Figure 13: Vorticity centerline slices compared with the reference results from the literature: BS [67] and BP [74]. (Left) Vorticity along $y = 0.5$ (y-component), (Right) Vorticity along $x = 0.5$ (x-component). (Top) General view, (Bottom) Zoomed-in views. [Colour figure can be viewed online]

the geometrical center of the cavity at $y = 0.5$ and $x = 0.5$. These profiles are compared with numerical results available in the literature. The results from our current simulations are represented using colored lines, with each color corresponding to the same Re value. Additionally, in Fig. 13, we juxtapose the vorticity profiles along the horizontal and vertical centerlines of the cavity with other available results, depicting very good agreement.

4.3. Example 3: Three-dimensional lid-driven cavity

The flows within a cavity inherently exhibit three-dimensional vortical structures, and their complexity is influenced by the Reynolds number, as initially demonstrated in experimental studies [78, 79].

We conduct simulations of cavity flow until a steady state is achieved for various values of Re ranging from 100 to 2900. The steady-state flow patterns for six Reynolds numbers are illustrated in Figures 14, 15, and 16, showing

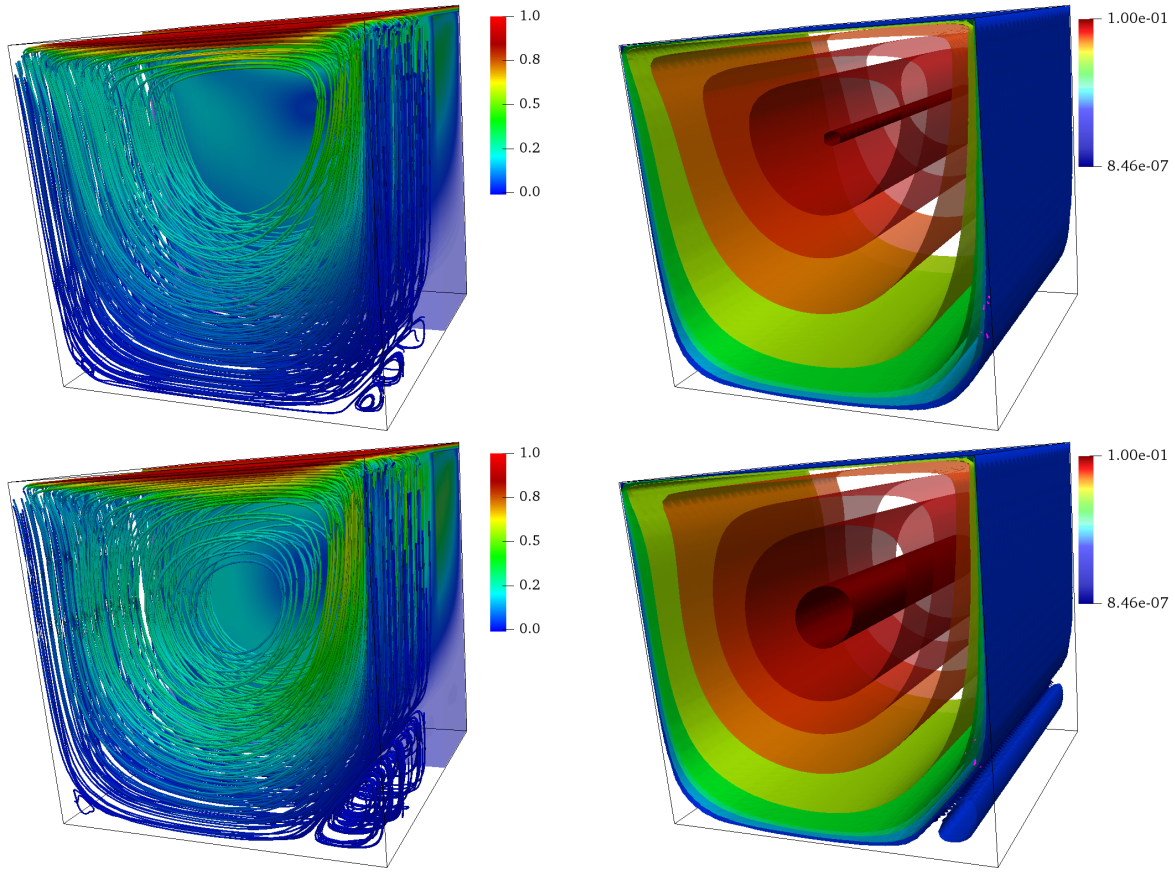

Figure 14: 3D lid driven cavity flow at the Steady-state with Re = 100 (top) and Re = 400. The figure displays the streamfunction (left) and corresponding isosurfaces (right). [Colour figure can be viewed online]

Stream-function $\|\boldsymbol{\psi}\|$				
1×10^{-4}	2.16×10^{-4}	6×10^{-4}	1×10^{-3}	4.65×10^{-3}
2.16×10^{-2}	5×10^{-2}	7.5×10^{-2}	1×10^{-1}	

Table 10: Isosurface values for the streamfunction contours in the steady-state displayed in Fig. 14, Fig. 15 and Fig. 16.

streamlines and corresponding isosurfaces. The isosurface values are provided in Table 10. The flow field profile shows asymmetry around the vertical centerline and primary vortex. As Re increases, we observe the formation of a new secondary vortex in the bottom-right corner, followed by a tertiary vortex in the bottom-left corner.

The primary vortex is defined by the peak value of the stream function $\|\boldsymbol{\psi}\|_{\max}$. This peak is reached at the spatial location $\boldsymbol{x}_P = (x_P, y_P, z_P)$, which features a stream vector $\boldsymbol{\psi}_P \equiv (\boldsymbol{\psi}_{P,1}, \boldsymbol{\psi}_{P,2}, \boldsymbol{\psi}_{P,3})$ and a vorticity vector $\boldsymbol{\omega}_P = (\boldsymbol{\omega}_{P,1}, \boldsymbol{\omega}_{P,2}, \boldsymbol{\omega}_{P,3})$. For future quantitative analysis in relation to our work, we compute the characteristics of the primary vortices for the previous Re numbers at steady state and present them in Table 11.

Similar to the two-dimensional case, we perform a quantitative analysis of the computational cost comparison between the N3 strategy and the classical N2 method. Let #K denote the number of mesh tetrahedra and #DOFs the total number of degrees of freedom. For two distinct mesh sizes, we set Re = 1000 and execute parallel simulations on a cluster with 52 CPU cores (2x Intel Xeon Gold 6230R). Table 12 presents the total CPU times for these computations, demonstrating that the N3 approach is more cost-effective than the N2 method due to its third-order convergence behavior.

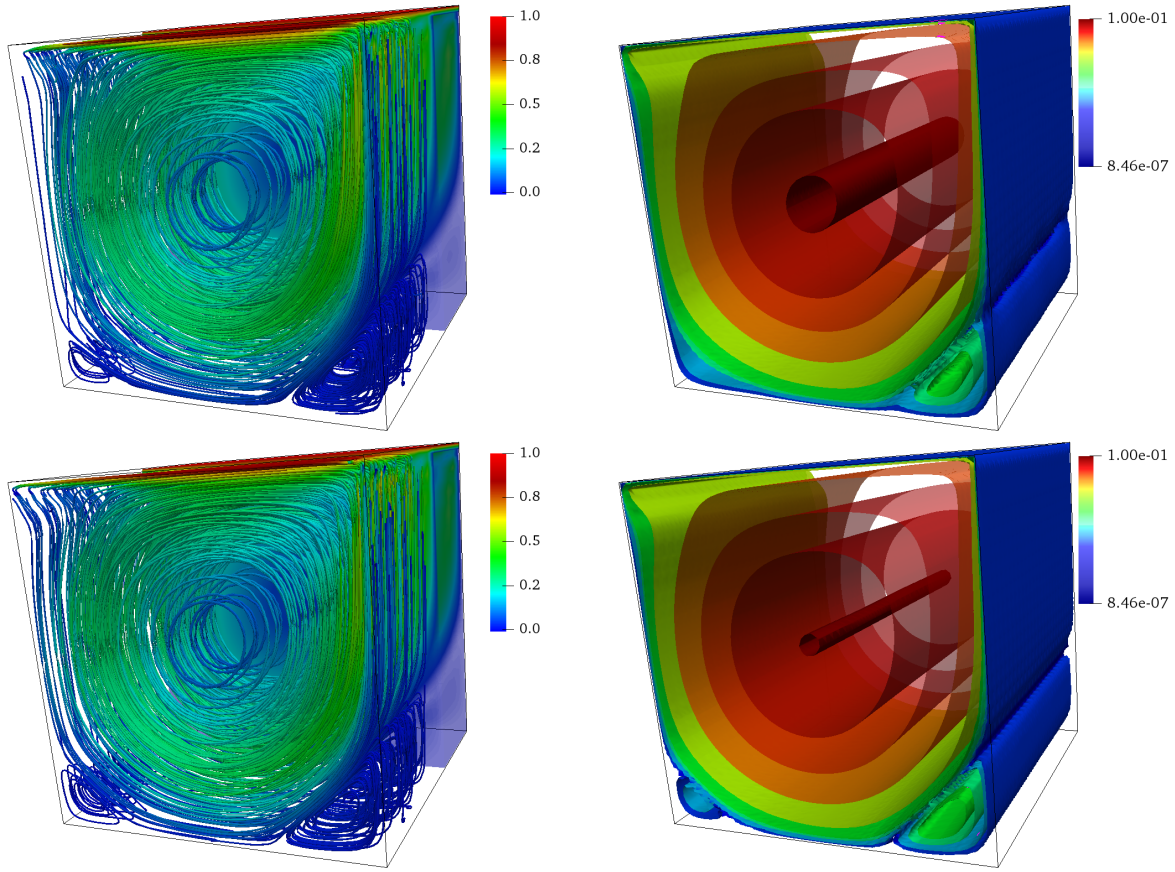

Figure 15: 3D lid-driven cavity flow at the Steady-state flow with Re = 1000 (top) and Re = 1500. The figure displays the streamfunction (left) and corresponding isosurfaces (right). [Colour figure can be viewed online]

We now focus on the 3D steady-state behavior using successively refined mesh sizes. Specifically, we examine key flow characteristics, including the streamline vector $\boldsymbol{\psi}_{\max}$ and the corresponding vorticity vector $\boldsymbol{\omega}_P$ that define

	Primary vortex					
	Re = 100	Re = 400	Re = 1000	Re = 1500	Re = 2000	Re = 2900
$\|\|\boldsymbol{\psi}\|\|_{\max}$	1.0035×10^{-1}	1.0037×10^{-1}	1.0255×10^{-1}	1.00376×10^{-1}	9.7417×10^{-2}	9.3817×10^{-2}
$\boldsymbol{\psi}_{P,1}$	-3.7426×10^{-6}	1.4174×10^{-5}	3.3005×10^{-5}	-2.4320×10^{-5}	-8.9683×10^{-5}	-4.6503×10^{-5}
$\boldsymbol{\psi}_{P,2}$	1.0035×10^{-1}	1.0373×10^{-1}	1.0255×10^{-1}	1.0038×10^{-1}	9.7417×10^{-2}	9.3817×10^{-2}
$\boldsymbol{\psi}_{P,3}$	-6.1334×10^{-6}	-1.9796×10^{-5}	-8.3715×10^{-5}	3.2170×10^{-5}	5.6839×10^{-5}	-2.2647×10^{-4}
x_P	0.6154	0.5769	0.5385	0.5385	0.5200	0.5200
y_P	1	1	0.1026	0.6538	0.6800	0.6800
z_P	0.7308	0.6346	0.5769	0.5577	0.5600	0.5600
$\boldsymbol{\omega}_{P,1}$	-8.6724×10^{-3}	1.6674×10^{-3}	2.6509×10^{-3}	2.0500×10^{-3}	-3.3695×10^{-3}	4.2667×10^{-2}
$\boldsymbol{\omega}_{P,2}$	3.0039	2.2579	1.7983	1.6986	1.6205	1.5368
$\boldsymbol{\omega}_{P,3}$	-6.4416×10^{-4}	-4.5523×10^{-3}	3.7155×10^{-3}	2.4529×10^{-3}	2.1303×10^{-3}	-4.8984×10^{-3}

Table 11: Comparison of CPU computation times to reach the steady-state solution for Re = 1000. The computations were performed in parallel using 52 CPU cores (2x Intel Xeon Gold 6230R).

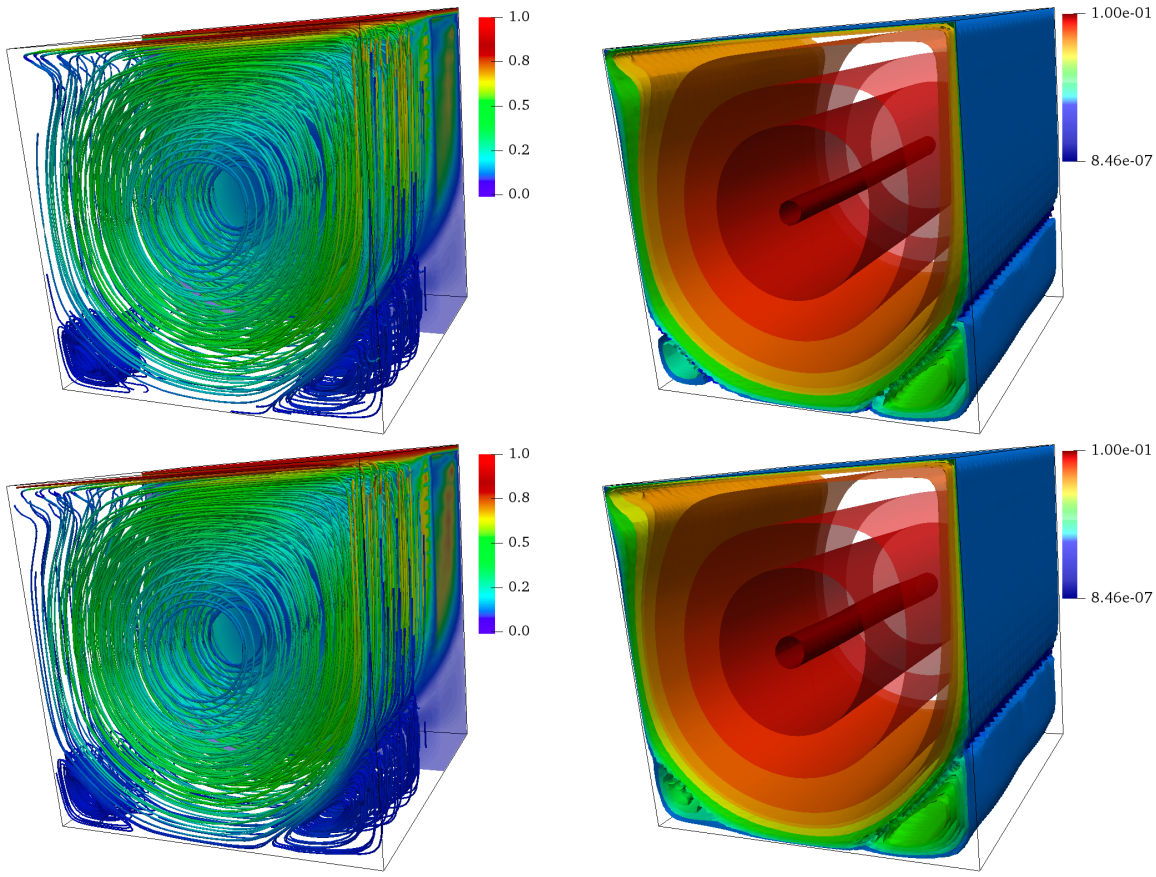

Figure 16: 3D lid driven flow at the steady-state with Re = 2000 (top) and Re = 2900. The figure displays the streamfunction (left) and corresponding isosurfaces (right). The isosurfaces with $\|\psi\| = 0.093$ are also displayed. [Colour figure can be viewed online]

#K	#DOFs	N2 method	N3 variant
93′750	415′529	84 min 2.07 s	74 min 48.18 s
48′000	216′024	21 min 28.61 s	18 min 38.23 s

Table 12: Comparison of CPU times to reach the steady-state solution for Re = 1000 using 52 CPU cores (2x Intel Xeon Gold 6230R).

the primary vortex, along with their components once the steady state is achieved. Table 13 presents the computed values for increasingly refined meshes, demonstrating convergence.

For qualitative comparisons, we also present in Fig. 17 a cross-sectional view through the midpoint plane intersecting at coordinates $(0.5, 0.5, 0.5)$ with a normal vector of $(0, 1, 0)^T$. The contour lines illustrate streamlines with isovalues of 1.1×10^{-4}, 7×10^{-2}, and 2.5×10^{-2} (from outermost to innermost). The results exhibit good agreement.

Additionally, we extract surfaces corresponding to specific isovalues of the velocity magnitude: 0.2 in Fig. 17 (Middle) and 0.25 in Fig. 17 (Right), depicting a qualitative convergence.

5. Concluding remarks

We have presented a stable numerical approach for solving highly non-linear problems, employing enhanced high-order Newton variants with third-order and fifth-order convergence. The numerical results of the two-dimensional and three-dimensional Newtonian flow in a lid-driven cavity highlight the N3 strategy's superior performance over

#K :	48′000	93′750	105′456	131′712
#DOFs :	216′024	415′529	466′314	579′968
$\|\|\psi\|\|_{\max}$	1.0136×10^{-1}	1.0375×10^{-1}	1.0414×10^{-1}	1.0478×10^{-1}
$\psi_{P,1}$	4.2785×10^{-5}	2.2472×10^{-5}	1.9831×10^{-5}	1.6448×10^{-5}
$\psi_{P,2}$	1.0136×10^{-1}	1.0375×10^{-1}	1.0414×10^{-1}	1.0478×10^{-1}
$\psi_{P,3}$	-1.5139×10^{-5}	-1.7729×10^{-5}	-1.7438×10^{-5}	-1.6593×10^{-5}
$\|\|\omega_P\|\|$	2.0505	2.1118	2.1230	2.1430
$\omega_{P,1}$	2.8918×10^{-4}	1.7645×10^{-3}	-2.8886×10^{-4}	-1.4086×10^{-3}
$\omega_{P,2}$	2.0505	2.2579	2.1230	2.1430
$\omega_{P,3}$	2.5923×10^{-3}	-5.9303×10^{-3}	1.6952×10^{-3}	1.4486×10^{-3}

Table 13: Comparison of flow characteristics for the primary vortex in a 3D lid-driven cavity at steady-state across successively refined meshes.

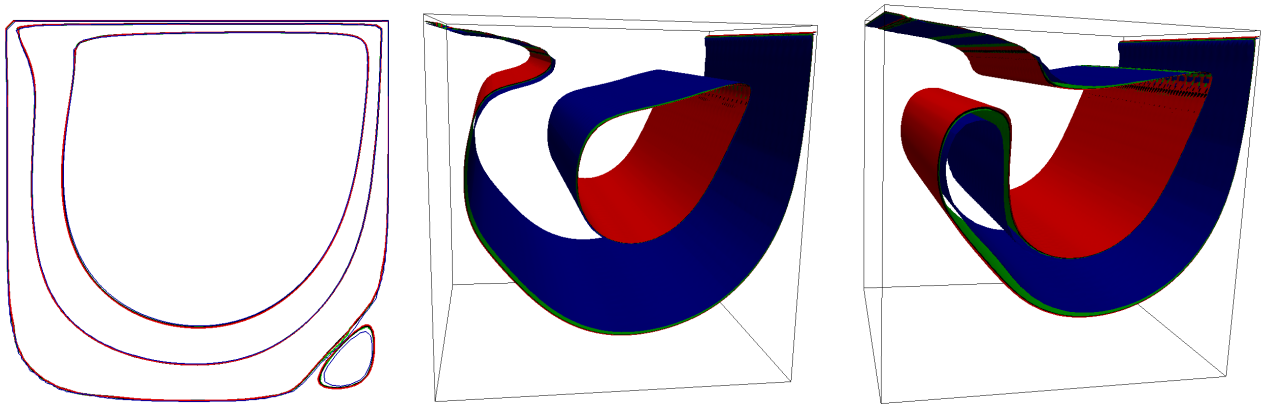

Figure 17: (Left) Cross-sectional view through the midpoint plane: Contour lines representing streamlines with isovalues 1.1×10^{-4}, 7×10^{-2}, and 2.5×10^{-2} (from outermost to innermost). Surfaces displaying isovalues of the velocity magnitude: 0.2 (Middle) and 0.25 (Right). Color legend: Blue (#DOFs: 216′024), Green (#DOFs: 415′529), Black (#DOFs: 466′314) and Red (# DOFs: 579′968). [Colour figure can be viewed online]

other approaches, especially in terms of computational cost. It exhibits faster convergence rates and the potential to significantly improve stability, particularly for larger Reynolds numbers. The introduction of a continuation strategy has further enhanced stability, allowing us to achieve much higher Reynolds numbers, up to 50′000 in the two-dimensional case.

This work is part of an ongoing effort to model highly nonlinear problems in bioengineering. Currently, we are exploring extensions of the developments in this paper, particularly focusing on multiphysics and fluid-structure interaction problems [2, 80, 81, 82]. We anticipate that the proposed framework could be effectively applied to simulate the hydrodynamics of biological membranes in confined geometries, which are known for their high nonlinearity and severe numerical stability constraints. Additionally, we see potential in applying this framework to model the coupled macroscopic cardiac mechanics in incompressible flow, where large structural deformations result in stiff problems, and the fluid dynamics induce high Reynolds regimes during the systole phase. Future work will also focus on developing suitable preconditioners for three-dimensional simulations, which could significantly reduce computational effort. Finally, we foresee the applicability of high order schemes for solving nonlinear partial-integro differential equations.

Acknowledgements

The authors gratefully acknowledge the financial support by Khalifa University through the grant FSU-2021-027 (#8474000367) and the Kuwait Foundation for the Advancement of Sciences (Project Code: PN22-16QE-1643). The

authors also recognize the contribution of Khalifa University's research computing facilities to the outcomes of this research.

Conflict of interest

The authors declare no potential conflict of interest.

References

[1] R. L. Burden, J. D. Faires, A. C. Reynolds, Numerical Analysis, PWS Publishing Company, Boston, 2001.
[2] P. Crosetto, Fluid-structure interaction problems in hemodynamics: Parallel solvers, preconditioners, and applications, Ph.D. thesis, EPFL, Lausanne (2011). doi:10.5075/epfl-thesis-5109.
[3] A. Laadhari, Implicit finite element methodology for the numerical modeling of incompressible two-fluid flows with moving hyperelastic interface, Applied Mathematics and Computation 333 (2018) 376–400.
[4] O. Duran, P. R. Devloo, S. M. Gomes, J. Villegas, A multiscale mixed finite element method applied to the simulation of two-phase flows, Computer Methods in Applied Mechanics and Engineering 383 (2021) 113870.
[5] H. Temimi, M. Ben-Romdhane, Numerical solution of falkner-skan equation by iterative transformation method, Mathematical Modelling and Analysis 23 (1) (2018) 139–151.
[6] M. Ben-Romdhane, H. Temimi, An iterative numerical method for solving the lane-emden initial and boundary value problems, International Journal of Computational Methods 15 (04) (2018) 1850020.
[7] C. Cai, J. Qiu, K. Wu, Provably convergent Newton-Raphson methods for recovering primitive variables with applications to physical-constraint-preserving Hermite WENO schemes for relativistic hydrodynamics, Journal of Computational Physics 498 (2024) 112669.
[8] M. Ben-Romdhane, H. Temimi, M. Baccouch, An iterative finite difference method for approximating the two-branched solution of bratu's problem, Applied Numerical Mathematics 139 (2019) 62–76.
[9] H. Temimi, M. Ben-Romdhane, M. Baccouch, M. Musa, A two-branched numerical solution of the two-dimensional bratu's problem, Applied Numerical Mathematics 153 (2020) 202–216.
[10] M. Mohammadi, S. Vakilipour, S. Ormiston, Newton linearization of the Navier-Stokes equations for flow computations using a fully coupled finite volume method, Applied Mathematics and Computation 397 (2021) 125916.
[11] N. Favrie, A. Renaud, D. Kondo, Hyperbolic modeling of gradient damage and one-dimensional finite volume simulations, Computer Methods in Applied Mechanics and Engineering 419 (2024) 116643.
[12] M. G. Carlino, W. Boscheri, Arbitrary-Lagrangian-Eulerian finite volume IMEX schemes for the incompressible Navier-Stokes equations on evolving Chimera meshes, Journal of Computational Physics 501 (2024) 112764.
[13] H. Homeier, A modified newton method with cubic convergence: The multivariate case, Journal of Computational and Applied Mathematics 169 (1) (2004) 161 – 169.
[14] J. Kou, Y. Li, X. Wang, A modification of newton method with third-order convergence, Applied Mathematics and Computation 181 (2) (2006) 1106–1111.
[15] C. Chun, A simply constructed third-order modification of newton's method, Journal of Computational and Applied Mathematics 219 (2008) 81–89.
[16] S. Li, H. Li, L. Cheng, Some second-derivative-free variants of halley's method for multiple roots, Applied Mathematics and Computation 215 (2009) 2192–2198.
[17] T. J. McDougall, S. J. Wotherspoon, A simple modification of newton's method to achieve convergence of order $1+\sqrt{2}$, Applied Mathematics Letters 29 (2014) 20–25.
[18] A. K. Maheshwari, A fourth-order iterative method for solving nonlinear equations, Applied Mathematics and Computation 211 (2009) 383–391.
[19] M. A. Noor, W. A. Khan, Fourth-order iterative method free from second derivative for solving nonlinear equations, Applied Mathematics and Sciences 6 (93) (2012) 4617–4625.
[20] C. Chun, Y. Ham, Some fourth-order modifications of newton's method, Applied Mathematics and Computation 197 (2008) 654–658.
[21] J. Kou, Y. Li, X. Wang, A composite fourth-order iterative method for solving non-linear equations, Applied Mathematics and Computation 184 (2) (2007) 471–475.
[22] R. Behl, A. Cordero, S. Motsa, J. R. Torregrosa, On developing fourth-order optimal families of methods for multiple roots and their dynamics, Applied Mathematics and Computation 265 (2015) 520–532.
[23] F. Zafar, A. Cordero, J. R. Torregrosa, Stability analysis of a family of optimal fourth-order methods for multiple roots, Numerical Algorithms 81 (3) (2019) 947 – 981.
[24] J. R. Sharma, H. Arora, M. S. Petkovic, An efficient derivative free family of fourth order methods for solving systems of nonlinear equations, Applied Mathematics and Computation 235 (2014) 383–393.
[25] F. Ahmad, F. Soleymani, F. Khaksar Haghani, S. Serra-Capizzano, Higher order derivative-free iterative methods with and without memory for systems of nonlinear equations, Applied Mathematics and Computation 314 (2017) 199–211.
[26] H. Singh, J. R. Sharma, S. Kumar, A simple yet efficient two-step fifth-order weighted-newton method for nonlinear models, Numerical Algorithms 93 (1) (2023) 203 – 225.
[27] M. Rafiullah, A fifth-order iterative method for solving nonlinear equations, Numerical Analysis and Applications 14 (2011) 297–302.
[28] N. Y. Abdul-Hassan, A. H. Ali, C. Park, A new fifth-order iterative method free from second derivative for solving nonlinear equations, Journal of Applied Mathematics and Computation 68 (2022) 2877–2886.

[29] J. R. Sharma, P. Gupta, An efficient fifth order method for solving systems of nonlinear equations, Computers & Mathematics with Applications 67 (3) (2014) 591–601.

[30] O. Said Solaiman, I. Hashim, Two new efficient sixth order iterative methods for solving nonlinear equations, Journal of King Saud University - Science 31 (4) (2019) 701–705.

[31] M. Grau-Sanchez, A. Grau, M. Noguera, On the computational efficiency index and some iterative methods for solving systems of nonlinear equations, Journal of Computational and Applied Mathematics 236 (6) (2011) 1259–1266.

[32] A. Cordero, E. Gmez, J. R. Torregrosa, Efficient high-order iterative methods for solving nonlinear systems and their application on heat conduction problems, Complexity 2017 (2017).

[33] R. Behl, H. Arora, CMMSE: A novel scheme having seventh-order convergence for nonlinear systems, Journal of Computational and Applied Mathematics 404 (2022) 113301.

[34] X.-Y. Xiao, H.-W. Yin, Accelerating the convergence speed of iterative methods for solving nonlinear systems, Applied Mathematics and Computation 333 (2018) 8–19.

[35] J. R. Sharma, S. Kumar, I. K. Argyros, A class of higher-order newton-like methods for systems of nonlinear equations, International Journal of Computational Methods 19 (2) (2022).

[36] J. R. Sharma, S. Kumar, A class of accurate newton–jarratt-like methods with applications to nonlinear models, Computational and Applied Mathematics 41 (1) (2022).

[37] J. R. Sharma, H. Singh, A computationally efficient sixth-order method for nonlinear models, in: R. K. Sharma, L. Pareschi, A. Atangana, B. Sahoo, V. K. Kukreja (Eds.), Frontiers in Industrial and Applied Mathematics, Springer Nature Singapore, Singapore, 2023, pp. 567–585.

[38] R. Behl, A. Cordero, J. R. Torregrosa, A new higher-order optimal derivative free scheme for multiple roots, Journal of Computational and Applied Mathematics 404 (2022) 113773.

[39] S. Panday, A. Sharma, G. Thangkhenpau, Optimal fourth and eighth-order iterative methods for non-linear equations, Journal of Applied Mathematics and Computing 69 (1) (2023) 953 – 971.

[40] S. Kumar, J. Bhagwan, L. Jntschi, Numerical simulation of multiple roots of van der waals and cstr problems with a derivative-free technique, AIMS Mathematics 8 (6) (2023) 14288–14299.

[41] T. Zhanlav, R. Mijiddorj, O. Khuder, A family of newton-type methods with seventh and eighth-order of convergence for solving systems of nonlinear equations, Hacettepe Journal of Mathematics and Statistics 52 (4) (2023) 1006 – 1021.

[42] B. Campos, J. Canela, P. Vindel, Dynamics of newton-like root finding methods, Numerical Algorithms 93 (4) (2023) 1453 – 1480.

[43] X. Zhou, B. Liu, Iterative methods for multiple roots with memory using self-accelerating technique, Journal of Computational and Applied Mathematics 428 (2023).

[44] J. M. Fernndez-Daz, C. O. Menndez-Prez, A superlinear scaling factor regula falsi root finder that detects the simple or multiple character of the root, Mathematics and Computers in Simulation 215 (2024) 1–20.

[45] A. Duarte Gomez, N. Deak, F. Bisetti, Jacobian-free Newton-Krylov method for the simulation of non-thermal plasma discharges with high-order time integration and physics-based preconditioning, Journal of Computational Physics 480 (2023) 112007.

[46] X. Zhou, Jacobian-free Newton Krylov coarse mesh finite difference algorithm based on high-order nodal expansion method for three-dimensional nuclear reactor pin-by-pin multiphysics coupled models, Computer Physics Communications 282 (2023) 108509.

[47] J. Chouchoulis, J. Schütz, Jacobian-free implicit MDRK methods for stiff systems of ODEs, Applied Numerical Mathematics 196 (2024) 45–61.

[48] A. M. Ostrowski, Solution of equations and systems of equations: Pure and applied mathematics: A series of monographs and textbooks, vol. 9, Vol. 9, Elsevier, 2016.

[49] R. Suparatulatorn, S. Suantai, Stability and convergence analysis of hybrid algorithms for berinde contraction mappings and its applications, Results in Nonlinear Analysis 4 (2021) 159–168.

[50] N. H. Tuan, H. S. Mohammadi, A mathematical model for covid-19 transmission by using the caputo fractional derivative, Chaos, Solitons & Fractals 140 (2020) 110107.

[51] N. H. Tuan, Y. Zhou, Well-posedness of an initial value problem for fractional diffusion equation with caputo-fabrizio derivative, Journal of Computational and Applied Mathematics 375 (2020) 112811.

[52] A. Cordero, C. Jordn, E. Sanabria-Codesal, J. R. Torregrosa, Highly efficient iterative algorithms for solving nonlinear systems with arbitrary order of convergence p+3, p5, Journal of Computational and Applied Mathematics 330 (2018) 748–758.

[53] A. Laadhari, G. Székely, Fully implicit finite element method for the modeling of free surface flows with surface tension effect, International Journal for Numerical Methods in Engineering 111 (11) (2017) 1047–1074.

[54] A. Laadhari, Exact Newton method with third-order convergence to model the dynamics of bubbles in incompressible flow, Applied Mathematics Letters 69 (2017) 138–145.

[55] A. Laadhari, P. Saramito, C. Misbah, G. Szekely, Fully implicit methodology for the dynamics of biomembranes and capillary interfaces by combining the Level Set and Newton methods, Journal of Computational Physics 343 (2017) 271–299.

[56] A. Laadhari, An operator splitting strategy for fluid-structure interaction problems with thin elastic structures in an incompressible Newtonian flow, Applied Mathematics Letters 81 (2018) 35–43.

[57] A. Axelsson, Globally convergent continuation methods for non-linear equations, Nijmegen: Subfaculty Mathematics, 1997.

[58] C. B. Garcia, F. J. Gould, Relations between several path following algorithms and local and global newton methods, SIAM Review 22 (3) (1980) 263–274.

[59] L. C. Evans, A new proof of local $C^1\alpha$ regularity for solutions of certain degenerate elliptic P.D.E., Journal of Differential Equations 45 (3) (1982) 356–373.

[60] A. Aristov, Exact solutions of a nonlinear equation with p-Laplacian, Lobachevskii J Math 42 (2021) 1768–1775.

[61] R. Glowinski, A. Marroco, Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de Dirichlet non linéaires, Revue française d'automatique, informatique, recherche opérationnelle. Analyse numérique 9 (R2) (1975) 41–76.

[62] P. G. Ciarlet, The Finite Element Method for Elliptic Problems, North-Holland, Amsterdam, New York, Oxford, 1978.

[63] E. L. Allgower, K. Georg, Springer Series in Computational Mathematics, Vol. 13, Springer Berlin Heidelberg, Berlin, Heidelberg, 1990, Ch. Numerical Continuation Methods: An Introduction, pp. XIV, 388.

[64] X. long Luo, H. Xiao, S. Zhang, Continuation newton methods with deflation techniques and quasi-genetic evolution for global optimization problems (2022). arXiv:2107.13864.

[65] U. Ghia, K. Ghia, C. Shin, High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method, Journal of Computational Physics 48 (3) (1982) 387–411.

[66] R. Schreiber, H. Keller, Driven cavity flows by efficient numerical techniques, Journal of Computational Physics 49 (2) (1983) 310–333.

[67] C.-H. Bruneau, M. Saad, The 2d lid-driven cavity problem revisited, Computers & Fluids 35 (3) (2006) 326–348.

[68] S. Khorasanizade, J. M. M. Sousa, A detailed study of lid-driven cavity flow at moderate reynolds numbers using incompressible sph, International Journal for Numerical Methods in Fluids 76 (10) (2014) 653–668.

[69] T. A. AbdelMigid, K. M. Saqr, M. A. Kotb, A. A. Aboelfarag, Revisiting the lid-driven cavity flow problem: Review and new steady state benchmarking results using gpu accelerated code, Alexandria Engineering Journal 56 (1) (2017) 123–135.

[70] R. Mahmood, N. Kousar, K. U. Rehman, M. Mohasan, Lid driven flow field statistics: A non-conforming finite element simulation, Physica A: Statistical Mechanics and its Applications 528 (2019) 121198.

[71] H. C. Kuhlmann, F. Roman, The lid-driven cavity, Computational Methods in Applied Sciences 50 (2019) 233 – 309.

[72] F. Auteri, N. Parolini, L. Quartapelle, Numerical investigation on the stability of singular driven cavity flow, Journal of Computational Physics 183 (1) (2002) 1–25.

[73] S. Vanka, Block-implicit multigrid solution of navier-stokes equations in primitive variables, Journal of Computational Physics 65 (1) (1986) 138–158.

[74] O. Botella, R. Peyret, Benchmark spectral results on the lid-driven cavity flow, Computers & Fluids 27 (4) (1998) 421–433.

[75] J. Donea, A. Huerta, Viscous Incompressible Flows, John Wiley & Sons, Ltd, 2003, Ch. 6, pp. 265–322.

[76] R. J. Labeur, G. N. Wells, A galerkin interface stabilisation method for the advection-diffusion and incompressible navier-stokes equations, Computer Methods in Applied Mechanics and Engineering 196 (49) (2007) 4985–5000.

[77] T. Pan, R. Glowinski, A projection/wave-like equation method for the numerical simulation of incompressible viscous fluid flow modeled by the navier-stokes equations, Comp. Fluid Dyn. J. 9 (2) (200) –.

[78] H. Rhee, J. Koseff, R. Street, Flow visualization of a recirculating flow by rheoscopic liquid and liquid crystal techniques, Experiments in Fluids 2 (2) (1984) 57 – 64.

[79] A. K. Prasad, J. R. Koseff, Reynolds number and endwall effects on a liddriven cavity flow, Physics of Fluids A: Fluid Dynamics 1 (2) (1989) 208–218.

[80] A. Laadhari, Y. Barral, G. Székely, A data-driven optimal control method for endoplasmic reticulum membrane compartmentalization in budding yeast cells, Mathematical Methods in the Applied Sciences 46 (2023) 8855–8876.

[81] J. W. Barrett, H. Garcke, R. Nurnberg, Numerical computations of the dynamics of fluidic membranes and vesicles, Physical Review E 92 (2015) 052704.

[82] N. Valizadeh, T. Rabczuk, Isogeometric analysis of hydrodynamics of vesicles using a monolithic phase-field approach, Computer Methods in Applied Mechanics and Engineering 388 (2022) 114191.