Implementing Newton Method to Develop 3D Fully Coupled Solvers using foam-extend

Behnam Cheraghi[1], Shidvash Vakilipour[2],
Faculty of New Sciences and Technologies, University of Tehran, Tehran, Iran

Abstract
Linearization of nonlinear terms in momentum and energy equations is one of the challenging and important issues besides pressure-velocity coupling. To date, Picard method has been successfully applied to large range of problems of computational fluid dynamics for linearizing the advection terms. In many problems, one can use other methods such as Newton Raphson method to increase the rate of convergence and improve the convergence and stability behavior. This paper describes the development of two prior prominent finite volume Newton-Picard linearized solvers from 2D to 3D and in the foam-extend platform. These two solvers were developed and well matured and tested in literature for incompressible flows. The first solver uses Newton method for linearize the advection term of energy equation and the remaining terms of other equations were linearized using Picard method. The second solver linearize the convection term of momentum equation using Newton method and the remaining terms of other equations with Picard. The solvers are based on finite volume (FV) and uses the foam-extend libraries and some developed functions. The performance of solvers were evaluated and validated in 2D test cases and their capabilities and performances were shown in some 3D cases. In 3D cases, similar results with 2D were observed. Reducing the number of iterations and capability to increasing the courant number are of the main achievement and advantages of these solvers.
Keywords: *Newton method - FVM solvers - Advection convection linearization – rate of convergence.*

1. Introduction

Linearization process for nonlinear system of equations is a common part in numerical simulation of most physical phenomena and engineering problems. In solving a fluid flow problem, it has always been a challenging issue in treating nonlinear advection terms in the NavierStokes or any other transport equations. It can be asserted that the main difference between various approaches of solving these equations is both the pressure-velocity coupling and the linearization of this particular nonlinear terms. Pressure-velocity coupling techniques, first developed on staggered grids by Patankar [1] and thereafter, successful attempts were made for its implementation on collocated grids, and to solve a variety of problems, [2]–[4]. In addition, implicit numerical algorithms and coupling procedures are well studied and developed to accelerate solution convergence and maintain its stability, [5]–[9]. Newton method has been employed for linearization of advection terms in several studies, [9]–[11]. Increasing the rate of solution convergence and maintaining numerical stability are the main objectives of these works. Vakilipour and Ormiston [9], implemented Newton method to linearize energy convection terms in a fully coupled finite volume algorithm. In comparison with Picard method, their results showed a significant increase of solution convergence rate for Newton method. Moreover, other studies confirmed that the Newton method for linearization of advection terms accelerates the solution convergence, [12, 13]. Mohammadi et al. [11], implemented a fully coupled algorithm together with fully implicit approach with Newton method on only momentum equation in two-dimensional and evaluated it on many test cases. A large increase in the rate of convergence and appropriate solution stability behavior were the results of this study.

In present study, two mentioned and old 2D fully coupled solvers, presented in [9] and [11], were implemented and developed to 3D solvers, using foam-extend, an open source object oriented C++ library for computational continuum mechanics. In this regard, the continuity, momentum, and temperature equations are implicitly discretized and results in a fully coupled system of linear equation. In all solvers, the buoyancy force term is modeled by the Boussinesq approximation and also, the associating temperature field is implicitly encountered in the momentum equations. Moreover, the full Picard solver encounters coupled implicit temperature field in the momentum equations, and is an extension of `transientFoam` solver of foam-extend and were used to compare two developed solvers. In full Picard solver, nonlinear advection terms in momentum and energy equations are linearized using the Picard method. In energy Newton solver, Newton method is employed to linearize advection term in energy equation, which its algorithm was developed, studied, and presented by Vakilipour and Ormiston [9]. The momentum Newton solver utilizes Newton method to linearize the momentum advection terms and its algorithm was developed and presented by Mohammadi et al..[11]. The computational performance and

1. PhD student, Aerospace engineering, 098-2186093249
2. Supervisor, 098-2161115754, 098-2186023297
vakilipour@ut.ac.ir (corresponding author)

behavior of the previously developed solvers are evaluated by solving some test cases.

2. Governing Equations

The governing equations for continuity, momentum and energy are given by:

$$\nabla.(\rho\mathbf{u}) = 0 \quad (1)$$

$$\frac{\partial(\rho\mathbf{u})}{\partial t} + \nabla.(\rho\mathbf{u}\mathbf{u}) = -\nabla p + \nabla.(\mu\nabla\mathbf{u}) + \mathbf{g}\beta(T - T_{ref}) \quad (2)$$

$$\frac{\partial(\rho T)}{\partial t} + \nabla.(\rho\mathbf{u}T) = \nabla.(\alpha\nabla T) \quad (3)$$

Where ρ, $\mathbf{u}$, p, T, μ, α, β, $\mathbf{g}$, T_{ref} are the fluid density, velocity vector, pressure, temperature, fluid viscosity, thermal diffusivity, thermal expansion coefficient, gravitational acceleration vector, and a reference temperature, respectively. In Eq. (2), Boussinesq approximation is employed to model the buoyancy force term.

2.1. Buoyancy term

The buoyancy term in Eq. (2), is a linear function of temperature and considered as a source for coupling the momentum and energy equations. The buoyancy term can be divided in two different parts; variable and constant.

$$\mathbf{g}\beta(T - T_{ref}) = \mathbf{g}\beta T - \mathbf{g}\beta T_{ref} \quad (4)$$

The first part is treated implicitly and changes diagonal elements of the coefficient matrix. The second part is incorporated into the RHS of the system of linear equations.

2.2. Newton Method

Newton method is categorized into the high order linearization methods while the Picard is a first order method to linearize nonlinear terms. In general, advection terms in a conservation equation are candidates for linearization either by Picard or Newton method [9]–[11]. To employ Newton method, an advection term is approximated using values calculated at previous iteration given by.

$$\dot{m}_f\phi_f \approx \dot{m}_f^k\phi_f + \dot{m}_f\phi_f^k - \dot{m}_f^k\phi_f^k \quad (5)$$

Where superscript k denotes previous iteration and an implicit variable is not denoted with a superscript. On the other hand, the Picard linearization method is implemented by.

$$\dot{m}_f\phi_f \approx \dot{m}_f^k\phi_f \quad (6)$$

2.3. The Coupling

Coupled solvers have been developed in foam-extend using block matrix manipulation and presented in several studies, [5]–[9, 11], And in pressure-based finite volume methods, Rhie-Chow scheme has been widely employed to link pressure and velocity fields [2,3,5,7]–[9,11,14]. For the developed solvers, the coefficient block matrix system and solution vector can be presented as follows.

$$\begin{bmatrix} [\text{Block}]_{Pi} & [\text{Block}]_{Ni} & \cdots \\ \vdots & [\text{Block}]_{Pj} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} [\text{Solution Vector}]_i \\ [\text{Solution Vector}]_j \\ \vdots \end{bmatrix} = \begin{bmatrix} [\text{Source}]_i \\ [\text{Source}]_j \\ \vdots \end{bmatrix} \quad (7)$$

Where P and N denotes for any present and neighbor cells, respectively. The Solution Vector contains five variables and Source stands for RHS vectors in the discretized equations of a present cell. Subscripts i and j are the dummy indices over all domain cells and their neighbors, respectively.

3. Implementation using foam-extend

In order to implement the previous steps using foam-extend, various functions have been used. Some of prominent and created functions are described as follows.

For assembling the diagonal contributions of buoyant term, the following function has been used:

```
insertEquationCoupling()
```

And for assembling the constant part of the buoyant term, the following syntaxes were used:

```
fvVectorMatrix UEqn
(
  …
  ==
  g*beta*Tref
)
```

Where the `UEqn` is an object of `fvVectorMatrix` class which contains the discretization operators and functions for performing momentum equation discretization and matrix manipulations.

The following syntax is used for generating coefficients of first implicit term of Newton linearization method:

```
fvm::div(phi,ϕ)
```

Where the `fvm` is a name space in foam-extend for implicit functions and operators and the `div` is function name for divergence terms.

Coefficients of second implicit part of Newton method were generated by multiplying the continuity equation coefficients and the desired face value. The face values were calculated by identical upwind scheme employed for the dependent variable in associating equation of momentum and energy. To perform this face value, one function operator is added using foam-extend as follow:

```
fvc::fluxUnit()
```

This `fluxUnit()` function is an extension of `flux()` function in original foam-extend-5.0, for calculating the face values.

The explicit (third term) part of Newton method were handled using the following syntax:

```
fvc::div(phi,ϕ)
```

Where `fvc` is a name space for explicit functions and operators.

Finally to creating the block matrix system for system of desired equations, a solution vector with 5 variables (3 velocity components, pressure and temperature) were used. In foam-extend the `vector2`, `vector3`, `vector4`, `vector6`, and `vector8` are available for solution vectors with 2, 3, 4, 6, and 8 variables. In present work, the `vector5` is created and used in all the intended solvers for managing 5 mentioned variables.

3.1. Momentum Newton Solver

In momentum Newton solvers linearized advection terms of momentum and energy equations is presented as:

$$\dot{m}^k \mathbf{u} + \dot{m}\mathbf{u}^k - \dot{m}^k \mathbf{u}^k \tag{6}$$

$$\dot{m}^k T \tag{7}$$

The associated procedure and included headers were for the first mentioned solver with momentum Newton linearized and energy Picard linearized is as follows:

```
fvBlockMatrix<vector5> UpTEqn(UpT);
#include "UEqn.H"
#include "pEqn.H"
#include "TEqn.H"
#include "UEqnNl.H"
#include "CouplingTerms.H"
```

Where `UpTEqn` is an object of `fvBlockMatrix` class for creating a block matrix system. `UEqn`, `pEqn`, `TEqn`, `UEqnNl` and `CouplingTerms`, are headers for managing momentum, continuity, energy, momentum Newton terms, and coupling terms respectively.

3.2. Energy Newton Solver

Also, in momentum Newton solvers linearized advection terms of momentum and energy equations is presented as:

$$\dot{m}^k \mathbf{u} \tag{8}$$

$$\dot{m}^k T + \dot{m}T^k - \dot{m}^k T^k \tag{9}$$

And the procedures or headers of second solver with momentum Picard linearized and energy Newton linearized is presented below:

```
fvBlockMatrix<vector5> UpTEqn(UpT);
#include "UEqn.H"
#include "pEqn.H"
#include "TEqn.H"
#include "TEqnNl.H"
#include "CouplingTerms.H"
```

Where `TEqnNl` is header file for Newton terms of energy equation.

4. Results and Discussion

The developed solvers are employed to solve a number of benchmark flow problems and obtained numerical solutions are assessed from the accuracy and convergence viewpoints. The test flow problems are solved on non-orthogonal grids at selected Reynolds and Richardson numbers which provides reasonable comparisons between computational performance of developed solvers in numerical simulation of test cases with acceptable physical and geometrical complexity. The Reynolds and Richardson numbers are selected in the ranges of up to 10000 and 0.1 to 10 respectively. A relative difference (RD) is defined to compare the numerical solutions given by:

$$RD(\phi) = \left|\frac{\phi_j - \phi_{j-1}}{\phi_{j-1}}\right| \times 100 \tag{10}$$

Where, ϕ_j and ϕ_{j-1} are the flow variables calculated on a grid and one level coarser, respectively.

4.1. Cavity with Buoyant Force

The first case, is a cavity with buoyant force to examine the solvers capabilities in solving the simple natural convection problem. The flow domain is a 2D cavity with a grid with 80×80 non-uniform cells and with vertical isothermal wall and horizontal adiabatic walls, described by [15], as shown in Fig. 1.

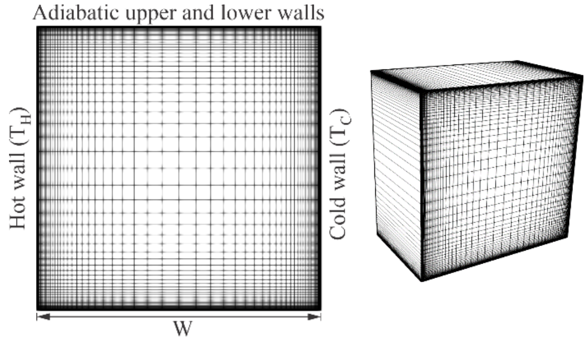

Fig. 1: Representation of non-uniform grid arrangement with boundary conditions

The Prandtl and Rayleigh numbers are defined as.

$$Pr = (\upsilon/\alpha) \tag{11}$$

$$Ra = \left(\frac{\mathbf{g}\,\beta\,\Delta T\,H^3}{\upsilon^2}\right)Pr \tag{12}$$

Where, υ, α and H are momentum diffusivity, thermal diffusivity and length scale, respectively. The interior of the cavity is filled with air, temperature difference is set to be 20 K between vertical walls, and the reference temperature has been set to 293 K.

Temperature distribution is consistent with the literature results and presented in Fig. 2.

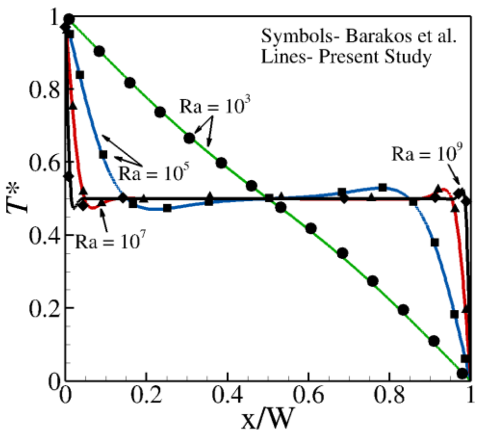

Fig. 2: Temperature distribution for cavity.

Also the similarity between the main structures is clear and presented in Fig. 3.

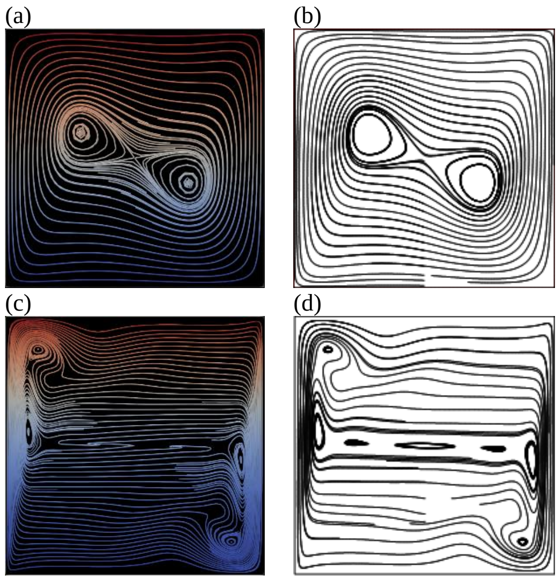

Fig. 3: The main structures a) Present study, Ra=10^5 b) Barakos et al. Ra=10^5. c) Present study, Ra=10^7 d) Barakos et al. Ra=10^7 .

Figure 4 shows the rate of convergence of mentioned solvers.

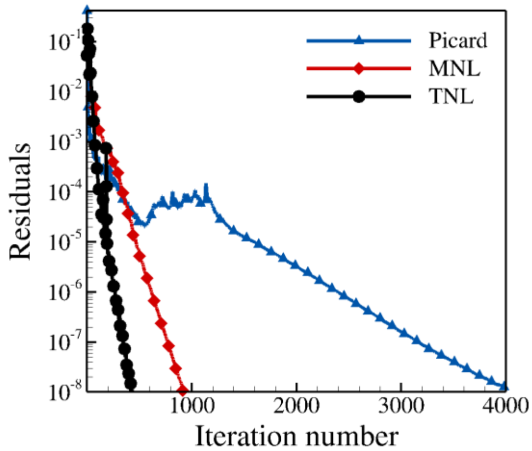

Fig. 4: Rate of convergence for 2D cavity at $Ra = 10^9$.

As can be seen from the Fig. 4, the solvers have better convergence rate than the full Picard solver.

4.2. Annulus

The annulus test case, is another case for examining the solvers to solve the natural convection problem in non-orthogonal grids, described in [9]. The domain and the discretization grid is presented at Fig. 5.

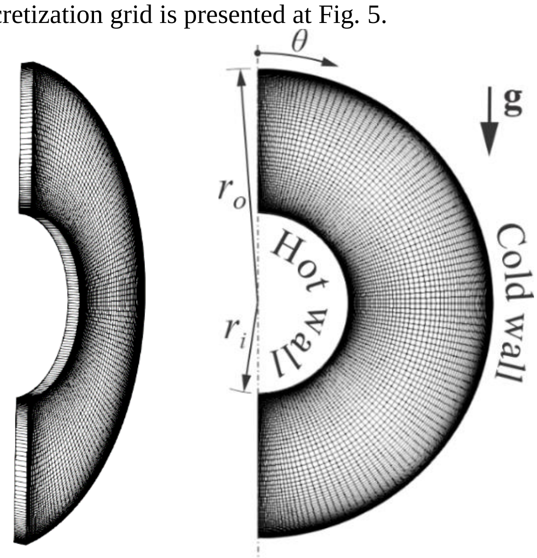

Fig. 5: a) Representation of annulus with boundary conditions and non-uniform, non-orthogonal grid.

Temperature difference, Prandtl number, Rayleigh number, and (r_o/r_i) ratio have been set to 20 K, 0.71, 4.7×10^4, and 2.6 respectively. The temperature profiles at various angles are depicted in Fig. 6.

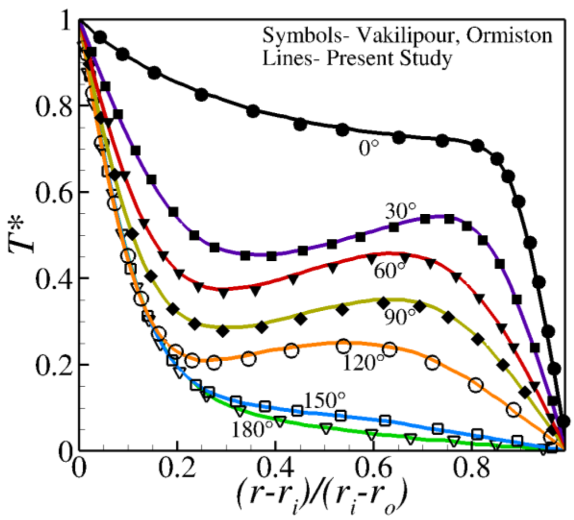

Fig. 6: Temperature profiles at various angles in comparison with Vakilipour,Ormiston [9].

A perfect match have been observed between the temperature profiles at Fig. 6, and also the resulted stream lines, presented in Fig. 7.

(a) (b)

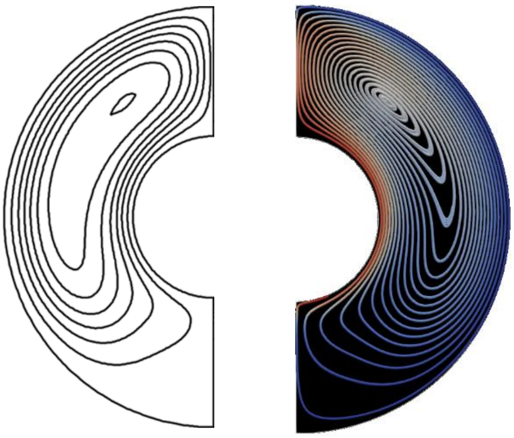

Fig. 7: Stream lines, left) Vakilipour, Ormiston right) Present study.

The rate of convergence in comparison with the full Picard solver is demonstrated at Fig. 8.

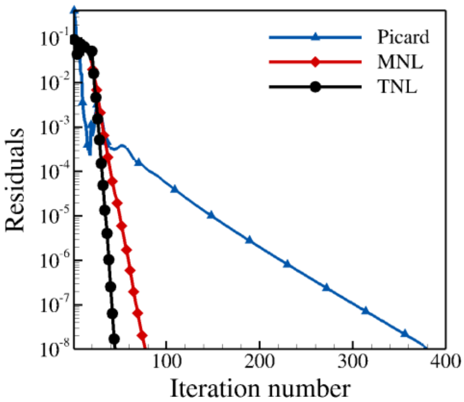

Fig. 8: Rate of convergence of mentioned solvers for annulus test case.

4.3. Rotating Cylinder in Cubical Cavity

The capability of momentum and energy Newton linearized solvers in problems are also clearly can be seen. The rotating cylinder in cubical cavity is the test case with non-orthogonal non-uniform grids. The domain is a cube with two isothermal (with top moving wall) and other full adiabatic outer walls and a rotating cylinder inside, which shown in Fig. 9.

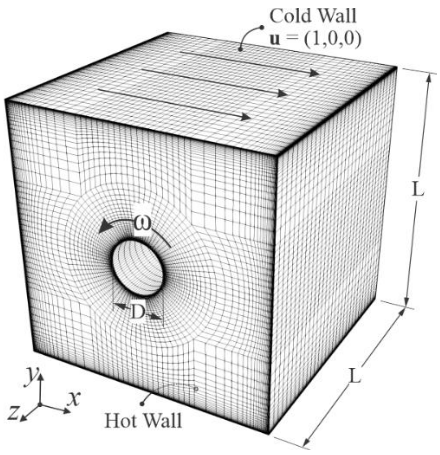

Fig. 9: Rotating cylinder in cubical cavity domain and grid.

The temperature difference and desired properties were set like the 2D cavity case and for simplicity have not been repeated. In present work, the flow computations is carried out at Reynolds numbers of 100 and Richardson numbers of 1. The rotating dimensionless velocity is defined as $\Omega = (\omega D)/(2\mathbf{u})$ and set to be 1, where ω is the angular velocity. The resulted velocity profile is consistent with literature [16], depicted at Fig. 10.

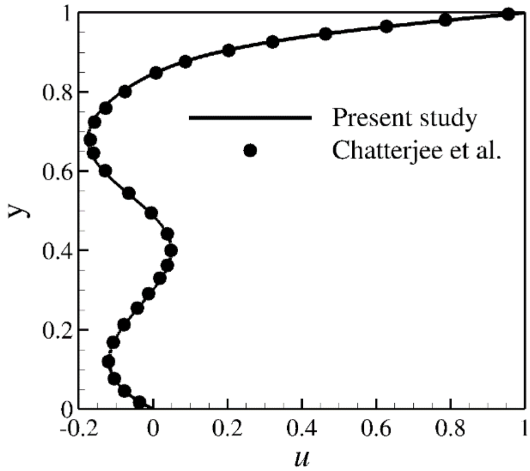

Fig. 10: Velocity along vertical line at $x = 0.25$ in comparison with Chatterjee et al. [16].

Major structures with temperature distribution is shown in Fig. 11.

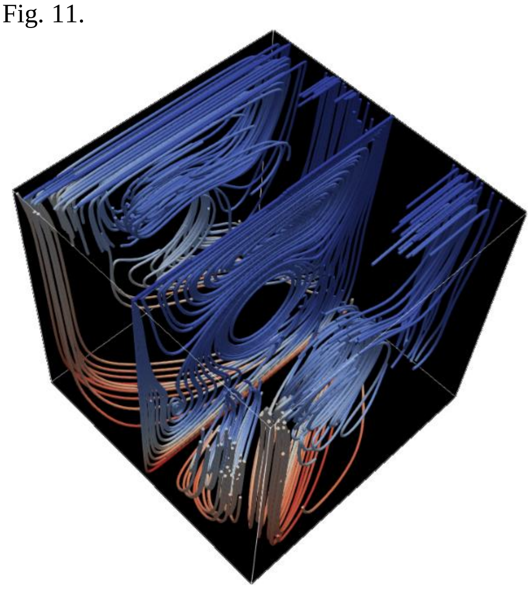

Fig. 11: Major flow structures with temperature distribution for Re, Ri, and Ω equal to 100, 1 and 1, respectively.

Finally, their capabilities in convergence behavior can be presented as Fig. 12.

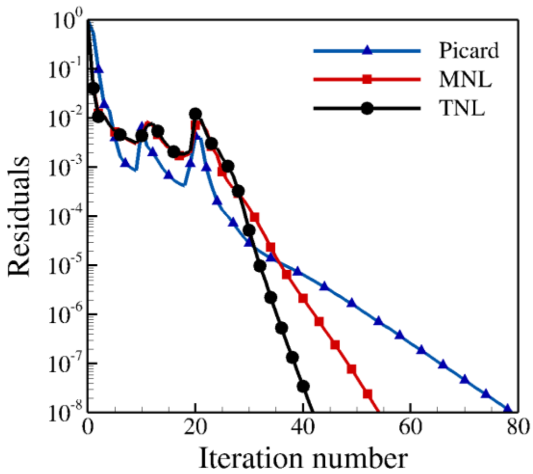

Fig. 12: Convergence rate behavior of the solvers.

As can be seen from the Fig. 12, the convergence rate of two developed Newton solvers are better than the Picard one, and the energy Newton solver, has better convergence rate in mixed convection flows.

5. Concluding Remarks

In present study, two coupled solvers previously developed using Newton method and introduced by Vakilipour and Ormiston [9] and Mohammadi et al. [11] were extended for 3D mixed convection flow calculations. The foam-extend libraries were employed to developed the Newton coupled solvers. The computational performance and capability of two developed solvers were investigated by numerical experiments performed for simulation of 2D and 3D natural and mixed convection flows. The following conclusions are given demonstrating the computational performance of developed solvers provided by the Newton method:

1. High convergence rate for natural and mixed convection flows is the first clear result in comparison with full Picard method.

2. Energy Newton solver has better convergence rate than the momentum Newton solver in flows with high Richardson numbers.

3. Newton method has its advantages either in uniform or non-uniform, orthogonal or non-orthogonal, and 2D or 3D grids.

4. Depending on Re, Ri or other flow and physical specifications, Newton method can reduce the number of iterations, even up to 4 to times.

6. References

[1] S. Patankar, Numerical Heat Transfer and Fluid Flow: CRC Press, Taylor & Francis Group, LLC. Florida, United States, 1980.

[2] C. Rhie, W. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, AIAA Journal 21 (1983) 1525–1532.

[3] T. Miller, F. Schmidt, Use of a pressure-weighted interpolation method for the solution of the incompressible Navier-Stokes equations on a nonstaggered grid system, Numerical Heat Transfer, Part A: Applications 14 (1988) 213–233.

[4] S. Karimian, G. Schneider, Pressure-based computational method for compressible and incompressible flows, Journal of thermophysics and heat transfer 8 (1994) 267–274.

[5] M. Darwish, I. Sraj, F. Moukalled, A coupled finite volume solver for the solution of incompressible flows on unstructured grids, Journal of Computational Physics 228 (2009) 180–201.

[6] H. Rusche, H. Jasak, Implicit solution techniques for coupled multifield problems block solution, coupled matrices, in: Advanced Training at the OpenFOAM Workshop, volume 23, 2010.

[7] L. Mangani, M. Buchmayr, M. Darwish, F. Moukalled, A fully coupled openfoam® solver for transient incompressible turbulent flows in ale formulation, Numerical Heat Transfer, Part B: Fundamentals 71 (2017) 313–326.

[8] T. Uroić, Implicitly coupled finite volume algorithms, Ph.D. thesis, University of Zagreb. Faculty of Mechanical Engineering and Naval Architecture, 2019.

[9] S. Vakilipour, S. Ormiston, A coupled pressure-based co-located finite-volume solution method for natural-convection flows, Numerical Heat Transfer, Part B: Fundamentals 61 (2012) 91–115.

[10] P. Galpin, G. Raithby, Treatment of non-linearities in the numerical solution of the incompressible navier–stokes equations, International journal for numerical methods in fluids 6 (1986) 409–426.

[11] 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.

[12] M. Pernice, H. F. Walker, Nitsol: A newton iterative solver for nonlinear systems, SIAM Journal on Scientific Computing 19 (1998) 302–318.

[13] K. Mitra, I. S. Pop, A modified l-scheme to solve nonlinear diffusion problems, Computers & Mathematics with Applications 77 (2019) 1722–1738.

[14] D. Anderson, J. C. Tannehill, R. H. Pletcher, Computational fluid mechanics and heat transfer, Taylor & Francis, 2016.

[15] Barakos, G., Ee Mitsoulis, and D. O. Assimacopoulos. "Natural convection flow in a square cavity revisited: laminar and turbulent models with wall functions." International journal for numerical methods in fluids 18.7 (1994): 695-719.

[16] D. Chatterjee, S. K. Gupta, B. Mondal, Mixed convective transport in a lid-driven cavity containing a nanofluid and a rotating circular cylinder at the center, International Communications in Heat and Mass Transfer 56 (2014) 71–78.