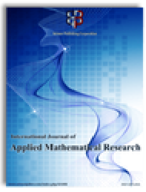
*International Journal of Applied Mathematical Research*, 5 (1) (2016) 63-68

SPC

**International Journal of Applied Mathematical Research**

*Website: www.sciencepubco.com/index.php/IJAMR*
*doi: 10.14419/ijamr.v5i1.5616*
***Research paper***



# Numerical solution of Navier stokes equation using control volume and finite element method

**Musa adam Aigo**[1] *

[1] *Umm Al-Qura University College Al-Qunfudah Mathematics department, Macca, SAUDI Arabia*
[1] *Dalanj University, Mathematics department, Sudan*
* *Corresponding author E-mail: maaigo@uqu.edu.sa*

## Abstract

The aim of this paper is twofold first we will provide a numerical solution of the Navier Stokes equation using the Projection technique and finite element method. The problem will be introduced in weak formulation and a Finite Element method will be developed, then solve in a fast way the sparse system derived. Second, the projection method with Control volume approach will be applied to get a fast solution, in iterations count.

***Keywords:*** *Navier stokes equation; Finite element; control volume; Projection.*

## 1. Introduction

We consider a fairly general version of the Navier Stokes equations, but we will explain the steps involved in simplifying those equations so that they are appropriate for the problems we actually wish to consider, namely steady state, incompressible flow in a two dimensional region, for example in [13], the author discuss the solution of the steady convection-diffusion equation as well as the solution of the steady, incompressible Navier Stokes equation. The additive Schwartz method was used with domain decomposition method to to solve the Navier Stokes equation, see [14]. We will consider the Navier equation or system of diffusion equation might be solved. Let us consider the numerical integration of the Navier-Stokes equations describing transient incompressible fluid flows in primitive variables subject to body forces $f$.

$$\begin{aligned} \partial_t u + u.\nabla u + \nabla P - \nu\Delta u &= f, \quad \nabla.u = 0 \text{ in } \Omega\times]0,T[ \\ u_{t=0} &= u_0 \\ u_{t=T} &= u_1 \end{aligned}$$

The terms:

- $\partial_t u$ : variation
- $u.\nabla u$ : Convection term
- $\partial_t u + u.\nabla u$: Inertia per volume
- $\nu\Delta u$ Diffusion term
- $P(x,t)$ is the pressure
- $u_0$: is a smooth initial velocity
- $f$ is a smooth function $(\mathscr{C}^\infty)$
- $\nabla.u = 0$: the incompressibility condition or the continuity equation.

The variable $P(x,t)$ is necessary to impose the incompressibility condition $\nabla.u = 0$. This set of equations is solved with appropriate initial and boundary conditions. We will consider only homogeneous Dirichlet boundary conditions. Initial conditions will provide an incompressible velocity field at the beginning of the time integration. With the superscript 0 for this initial field. In Navier-Stokes equations, the nonlinear advection terms account some difficulties in the mathematical analysis of their solutions and in the numerical integration process because they generate qualitative change of the solutions in the phase space (called bifurcations), transition, and, eventually, turbulence. The existence and uniqueness proofs of the solutions of the Navier-Stokes equations have been investigated by physicists, fluid dynamistic, and mathematicians since the early 19th century, when those equations were derived. The books by Galdi [1] and Témam [2] summarize the state of the art in this domain and also introduce the reader to challenging open questions. In [7], Chorin present a numerical solution of incompressible viscous flow. Chorin use the velocities and the pressure as variables and is equally applicable to problems in two and three dimensions, the method used is based on artificial compressibility parameter into the motion and the final solution is independent of this parameter. The method using the velocities and pressure in two dimensional incompressible flow problems have previously presented, for example in [8] the authors give a procedure which appears quite natural and may indeed in their problem be quite appropriate. In [9], the ICE technique for numerical fluid dynamics has been revised considerably, and generalized in such a way as to extend the applicability to fluid flows with arbitrary equation of state and the full viscous stress tensor. The method is useful for the numerical solution of time-dependent fluid flow problems in several space dimensions, for all Mach numbers from zero (incompressible limit) to infinity (hypersonic limit). In this paper, our goal is to extend and improve previous technique and solve in a fast way the Navier stokes equation. we will try avoid this problem by introducing two method, which allows us to compute solution. first of all, we will give a simple summary about Navier stokes equations. The simplification to the Navier Stokes equations occurs if we can assume that the flow problem has a steady behavior. For instance, if we turn on the tap in a sink, the flow will at first be very irregular; however, once the sink has filled up, and the water is pouring out over the sides as fast as it is pouring in, at least the rough form of the flow will be the same over time.

We might imagine that, to a reasonable approximation, even the small scale structure of this flow becomes fixed as time progresses. A flow which does not depend on time is known as a *steady state flow*.

In cases where we know, or hope, that the fluid flow problem has a steady state solution, we can certainly try to determine that solution by solving the Navier Stokes equations and looking at the behavior of the solution as time increases indefinitely.

An alternative way to reach the steady state solution is to assume that the variables $\mathbf{v}$, $\rho$, and $p$ do not depend on time, plug them into the Navier Stokes equations, and after simplification, arrive at the *steady state compressible Navier Stokes equations*:

$$\begin{aligned} -\mu\Delta\mathbf{v}+\rho(\mathbf{v}\cdot\nabla)\mathbf{v}+\nabla p &= 0 \qquad (1)\\ \rho\nabla\cdot\mathbf{v} &= 0 \qquad (2)\end{aligned}$$

The Steady State Incompressible Navier Stokes Equations.

Any physical fluid or gas will have variations in density. In many cases, especially when (temperate) water is involved, these variations are so small that the density may be taken to be a constant, as though the fluid were incompressible. In that case, we may simplify the equations by dividing through by the density to get the *steady state incompressible Navier Stokes equations*:

$$\begin{aligned} -\frac{1}{\rho}\Delta\mathbf{v}+(\mathbf{v}\cdot\nabla)\mathbf{v}+\frac{1}{\rho}\nabla p &= 0 \qquad (3)\\ \nabla\cdot\mathbf{v} &= 0 \qquad (4)\end{aligned}$$

To make our equations slightly less cluttered, we replace the dynamic viscosity by the *kinematic viscosity* $\nu$:

$$\nu=\frac{\mu}{\rho} \qquad (5)$$

and we replace the pressure by a scaled pressure, $\bar{p}$:

$$\bar{p}(\mathbf{x})=\frac{p(\mathbf{x})}{\rho} \qquad (6)$$

However, pressure is not an “interesting” quantity, so we often don’t care about its particular values or scaling. For convenience, then, we will ignore this rescaling of the pressure, and persist in using the old symbol $p$ when in fact we will mean the new, rescaled pressure. At this point, our cleaned up version of the incompressible steady Navier Stokes equations looks like this:

$$\begin{aligned} -\nu\Delta\mathbf{v}+(\mathbf{v}\cdot\nabla)\mathbf{v}+\nabla p &= 0 \qquad (7)\\ \nabla\cdot\mathbf{v} &= 0 \qquad (8)\end{aligned}$$

By using the appropriate scalings, our set of equations for the kinematic velocity $v$ look as simple as the previous set of equations for the mass velocity $u$, but are in fact simpler, since we have eliminated consideration of the density $\rho$.

This is the vector form of the Navier Stokes equations that we will be considering from now on. Our only further modification is to specify the spatial dimension, and, for convenience, to write out a scalar version of the equations. It might be helpful now to assume that we are working in a two-dimensional spatial region, and to write the steady incompressible Navier Stokes equations in a sort of scalar form. Thus, instead of $\mathbf{x}$ we will write $(x,y)$, and instead of the vector quantity $\mathbf{v}$ we will write $(u,v)$, where we are keeping track of the horizontal and vertical components of the kinematic velocity. From this perspective, our system now becomes three scalar equations, since our vector momentum equation is replaced by a pair of horizontal and vertical momentum equations:

$$\begin{aligned} -\nu(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2})+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+\frac{\partial p}{\partial x} &= 0 \qquad (9)\\ -\nu(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2})+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+\frac{\partial p}{\partial y} &= 0 \qquad (10)\\ \frac{\partial u}{\partial x}+\frac{\partial v}{\partial y} &= 0 \qquad (11)\end{aligned}$$

The outline of the paper is the following: In Section 2, we introduce the finite element method and projection method for solving Navier stokes equations. In section 3, we will solve the Navier-Stokes equations using variables (pressure and velocities), using a control volume approach. In last section we conclude.

## 2. Projection method

In this section, we will introduce the method and technique used to solve the Navier stokes equation. This technique is based on finite element method and projection technique [10] which allows us to solve the problem.

### 2.1. Basic idea

We recall that the finite element technique [11], [12] for finding an approximate solution of boundary value and initial value problems characterized by partial differential equation. It produces a stable solution of the problem to minimize the error using the variational method. The most distinctive feature of finite element method that separate it from others is the division of a given domain into a set of simple sub-domains, called finite elements. Any geometric shape that allows computation of the solution or its approximation, or provides necessary relations among the values of the solution at selected points, called nodes, of the sub-domain, qualifies as finite element. Other features of the method include seeking continuous, often polynomial, approximations of the solution over each element in terms of nodal values, and assembly of elements equations by imposing the interpellant continuity of the solution and balance of interpellant forces. There are three stages in the whole process where errors are generally introduced in most cases. The first is the partition of the domain into smaller sub-domains and then assembling it back to generate the original domain which introduces some errors in the domain during the process. Second stage is when element equations are derived. The dependent unknowns($u$) of the problem are approximated with the idea that any continuous function can be represented by a linear combination of unknown functions $\phi_i$ and undetermined coefficients $c_i$ ( $u\approx u_h=\Sigma c_i\phi_i$ ). Algebraic relations among the undetermined coefficients $c_i$ are obtained by satisfying the governing equations over each element in a weighted integral sense. The approximation functions $\phi_i$ are often taken to be polynomials and are derived using the concepts from interpolation theory. Therefore they are termed as interpolation functions. So in the second stage, errors are introduced both in representing the solution $u$ as well as in evaluating the integrals. And lastly errors are introduced in solving the assembled system of equations. Then we will develop a control volume approach and Projection technique to solve two dimensional flow. In this second case, we will approximate numerically all integral this allows us to transform the problem into discrete vector equations, then with the help of the projection technique the system will be solved. We recall that the projection method operates as a two-stage fractional step scheme, a method which uses multiple calculation steps for each numerical time-step. In many projection algorithms, the steps are split as follows: First the system is progressed in time to a mid-time-step position. This is called the predictor step. At this step an initial projection may be implemented such that the mid-time-step velocity field is enforced as divergence free. The corrector part of the algorithm is then progressed. These use the time-centered estimates of the velocity, density, etc. to form final time-step state. last step, the projection method is applied to enforce the divergence restraint on the velocity field. The system now is fully updated to the next step time. These steps of the projection method will be presented in details in Section 3. This technique is developed and presented with more details in [5], [6].

### 2.2. Discretization and Projection approach

The velocity $u$ and the pressure $P$ are coupled together by the incompressibility condition $\nabla.u = 0$ which makes the equation difficult to solve numerically. In the late of 60's Chorin [3] and Temam [4] constructed the the Projection method which decoupled the velocity and the pressure. The projection method is an effective means of numerically solving time-dependent incompressible fluid-flow problems. It was originally introduced by Alexandre Chorin in 1967 as an efficient means of solving the incompressible Navier-Stokes equations. The key advantage of the projection method is that the computations of the velocity and the pressure fields are decoupled. A possible algorithm, proposed by Chorin, is

$$\begin{aligned}
\frac{1}{\delta t}\left(u^{m+1}-u^m\circ X^m\right)+\nabla p^m-\mu\Delta u^m &= 0\\
u_{|\Gamma} &= u_\Gamma\\
\mu\partial_n u_{|\Gamma\text{ out}} &= 0\\
-\Delta p^{m+1} &= -\nabla.u^m\circ X^m\\
\partial_n p^m{}_{|\Gamma} &= 0\\
p^{m+1}_{|\Gamma\text{ out}} &= 0
\end{aligned}$$

where $u\circ X(x) = u(x-u(x)\delta t)$ since $\partial_t u + u.\nabla u$ is approximated by the method of characteristics (in practice we use the Characteristics-Galerkin Method) in our numerical implementation. We use the Chorin's algorithm with free boundary condition at outlet: i.e. $u = 0, \nu\partial_n u = 0$. to compute a correction, $q$, to the pressure.

$$-\Delta q = \nabla.u, q = 0 \text{ on } \Gamma_{out}$$

and define $u^{m+1} = \tilde{u} + P\nabla q\delta t$, $p^{m+1} = p^m - q$ where $\tilde{u}$ is the $(u^{m+1}, v^{m+1})$ of Chorin's algorithm, and where $P$ is the $L^2$ projection with mass lumping and exactly a sparse matrix. The geometry is that of a channel with a backward facing step so that the in ow section is smaller than the outflow section. This geometry produces a fluid recirculation zone that must be captured correctly. This can only be done if the triangulation is sufficiently fine, or well adapted to the flow. Of course, a numerical treatment requires a bounded domain, the domain will be considered is bounded hexagon domain $\Omega = \underbrace{A_0A_1A_2A_3A_4A_5}$ where

$$\begin{aligned}
A_0: &= \{x=0; y=t; t\in[0,1]\}\\
A_1: &= \{x=2t; y=0; t\in[0,1]\}\\
A_2: &= \{x=2; y=-t/2; t\in[0,1]\}\\
A_3: &= \{x=2+18t; y=-1/2; t\in[0,1]\}\\
A_4: &= \left\{x=20; y=-0.5+\frac{3}{2}t; t\in[0,1]\right\}\\
A_5: &= \{x=20t; y=1; t\in[0,1]\}
\end{aligned}$$

The convergence of the method will be tested with $L^2$ norm of the error. Figure **??**, show the error according to the number of iteration. We solved the incompressible Navier stokes equation in the hexagon domain using $u_{Bcint} = 4y(1-y)(y>0)(x<2)$ and $u_{Bcout} = 4./1.5(y+0.5)(1-y)(x>19)$; and the Boundary condition is defined as

$$u_{bc} = u_{BCin} + u_{BCout}$$

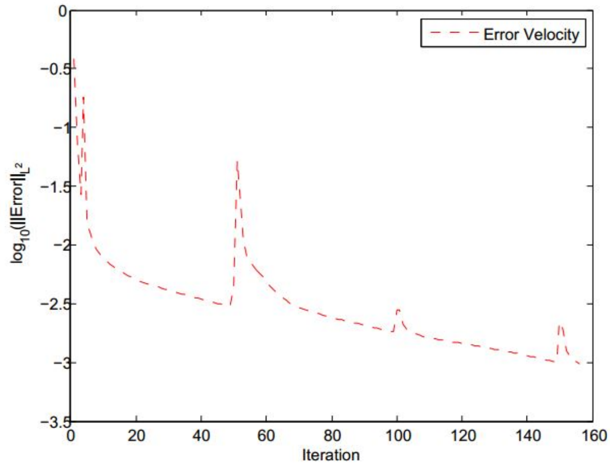


**Figure 1:** Error vs Iteration

## 3. Two dimensional flow

We will solve the Navier-Stokes equations using variables (pressure and velocities), using a control volume approach on a staggered grid. We will introduce the problem that must be solved in discrete form and the strategy used to get the solution adjusted to some boundary conditions.

### 3.1. Equations

We consider the case of two dimensional flow where $\mathbf{u} = (u,v)$ The Navier-Stokes momentum equation can be derived as a particular form of the Cauchy momentum equation. The momentum equation is:

$$\begin{aligned}
\frac{\partial}{\partial t}\int_V udV &= -\oint_S u\mathbf{u}.\mathbf{nds} - \frac{\mathbf{1}}{\rho}\oint_{\mathbf{S}}\mathbf{pn_x ds} + \nu\oint_{\mathbf{S}}\nabla\mathbf{u}.\mathbf{nds}\\
\frac{\partial}{\partial t}\int_V vdV &= -\oint_S v\mathbf{u}.\mathbf{nds} - \frac{\mathbf{1}}{\rho}\oint_{\mathbf{S}}\mathbf{pn_y ds} + \nu\oint_{\mathbf{S}}\nabla\mathbf{v}.\mathbf{nds}\\
\oint_S \mathbf{u}.\mathbf{nds} &= 0
\end{aligned}$$

The domain of computation is the square volume $[-1,1]^2$. The velocity at the end of each time step must satisfy conservation of mass,i.e $\int_S \mathbf{u}.\mathbf{nds} = \mathbf{0}$ . Integrate over the boundary and after simplification we get:

$$u^{n+1}_{i+1,j} - u^{n+1}_{i-1/2,j} + v^{n+1}_{i,j+1/2} - v^{n+1}_{i,j-1/2} = 0\,.$$

When the mass conservation equation will be applied to a control volume centered, we obtain the velocities at the edges. They could be interpolated from values at the cell center.

- Unsteady term
  - Rate of change of $x$-momentum, can be computed over the control volume:

$$\partial_t\int_V udV = \frac{u^{n+1}_{i+1/2,j} - u^n_{i+1/2,j}}{\delta t}h^2$$

  - Rate of change of $y$-momentum, can be computed over the control volume:

$$\partial_t\int_V vdV = \frac{v^{n+1}_{i,j+1/2} - v^n_{i,j+1/2}}{\delta t}h^2$$

- Advection term

- flow at $x$-momentum, can be computed and some approximation can be done to simplify the approximation.

$$\begin{aligned}\int_V u\mathbf{u}.\mathbf{nds} &= h\left((u^2)^n_{i+1,j}-(u^2)^n_{i,j}\right)\\ &+ h\left((uv)^n_{i+1/2,j+1/2}-(uv)^n_{i+1/2,j-1/2}\right)\end{aligned}$$

$$(u^2)^n_{i+1,j}=\left(\frac{u^n_{i+3/2,j}+u^n_{i+1/2,j}}{2}\right)^2$$

$$(u^2)^n_{i,j}=\left(\frac{u^n_{i-1/2,j}+u^n_{i+1/2,j}}{2}\right)^2$$

$$(uv)^n_{i+1/2,j+1/2}=\frac{u^n_{i+1/2,j}+u^n_{i+1/2,j+1}}{2}\;\frac{v^n_{i,j+1/2}+v^n_{i+1,j+1/2}}{2}$$

$$(uv)^n_{i+1/2,j-1/2}=\frac{u^n_{i+1/2,j}+u^n_{i+1/2,j-1}}{2}\;\frac{v^n_{i,j-1/2}+v^n_{i+1,j-1/2}}{2}$$

- In the same way, flow at $y$-momentum, can be computed, and we have the following approximation

$$\begin{aligned}\int_V v\mathbf{u}.\mathbf{nds} &= \left((uv)^n_{i+1/2,j+1/2}-(uv)^n_{i-1/2,j+1/2}\right)h\\ &+ \left((v^2)^n_{i,j+1}-(v^2)^n_{i,j}\right)h\end{aligned}$$

$$\begin{aligned}(v^2)^n_{i,j+1} &= \left(\frac{v^n_{i,j+3/2}+v^n_{i,j+1/2}}{2}\right)^2\\ (v^2)^n_{i,j} &= \left(\frac{u^n_{i,j-1/2}+u^n_{i,j+1/2}}{2}\right)^2\\ (uv)^n_{i+1/2,j+1/2} &= \frac{u^n_{i+1/2,j}+u^n_{i+1/2,j+1}}{2}\;\frac{v^n_{i,j+1/2}+v^n_{i+1,j+1/2}}{2}\\ (uv)^n_{i-1/2,j+1/2} &= \frac{u^n_{i-1/2,j}+u^n_{i-1/2,j+1}}{2}\;\frac{v^n_{i,j+1/2}+v^n_{i-1,j+1/2}}{2}\end{aligned}$$

- Pressure term, the integration over the boundary gives:

$$\left(\oint_S pn_x ds,\oint_S pn_y ds\right)\simeq h\left(p_{i+1,j}-p_{i,j},p_{i,j+1}-p_{i,j}\right)$$

- Viscous diffusion:

$$\begin{aligned}\oint_S \nabla u\mathbf{n}ds &\simeq \left(\frac{\partial u}{\partial x}\right)^n_{i+1,j}-\left(\frac{\partial u}{\partial x}\right)^n_{i,j}\\ &+ \left(\frac{\partial u}{\partial y}\right)^n_{i+1/2,j+1/2}-\left(\frac{\partial u}{\partial y}\right)^n_{i+1/2,j-1/2}\end{aligned}$$

where the derivatives at the boundary can be computed using these approximation:

$$\begin{aligned}\left(\frac{\partial u}{\partial x}\right)^n_{i,j} &\simeq \frac{u^n_{i+1/2,j}-u^n_{i-1/2,j}}{h},\\ \left(\frac{\partial u}{\partial y}\right)^n_{i+1/2,j+1/2} &\simeq \frac{u^n_{i+1/2,j+1}-u^n_{i-1/2,j}}{h}\end{aligned}$$

this lead to

$$\begin{aligned}\left(\frac{\partial u}{\partial x}\right)^n_{i+1,j}-\left(\frac{\partial u}{\partial x}\right)^n_{i,j} &= u^n_{i+3/2,j}+u^n_{i-1/2,j}\\ &- 2u^n_{i+1/2,j}\end{aligned}$$

$$\begin{aligned}\left(\frac{\partial u}{\partial y}\right)^n_{i+1/2,j+1/2}-\left(\frac{\partial u}{\partial y}\right)^n_{i+1/2,j-1/2} &= u^n_{i+1/2,j+1}\\ &- 2u^n_{i+1/2,j}\\ &+ u^n_{i+1/2,j-1}\end{aligned}$$

Therefore we have

$$\begin{aligned}\oint_S \nabla u.\mathbf{n}ds &\simeq u^n_{i+3/2,j}+u^n_{i-1/2,j}+u^n_{i+1/2,j+1}\\ &+ u^n_{i+1/2,j-1}-4u^n_{i+1/2,j}\\ \oint_S \nabla v.\mathbf{n}ds &\simeq v^n_{i,j+3/2}+v^n_{i,j-1/2}+v^n_{i+1,j+1/2}\\ &+ v^n_{i-1,j+1/2}-4v^n_{i,j+1/2}\end{aligned}$$

We collect all the terms together we obtain the following system

$$\begin{aligned}\frac{u^{n+1}_{i+1/2,j}-u^n_{i+1/2,j}}{\delta t}h^2 &= h\left((u^2)^n_{i+1,j}-(u^2)^n_{i,j}+(uv)^n_{i+1/2,j+1/2}\right)\\ &- h(uv)^n_{i+1/2,j-1/2}\\ &+ \nu\left((u^2)^n_{i+1,j}-(u^2)^n_{i,j}+(uv)^n_{i+1/2,j+1/2}\right)\\ &- \nu(uv)^n_{i+1/2,j-1/2}\\ &- h\left(p_{i+1,j}-p_{i,j}\right)\\ \frac{v^{n+1}_{i,j+1/2}-v^n_{i,j+1/2}}{\delta t}h^2 &= -h\left((uv)^n_{i+1/2,j+1/2}-(uv)^n_{i-1/2,j+1/2}\right)\\ &- h\left((v^2)^n_{i,j+1}-(v^2)^n_{i,j}\right)\\ &+ \nu\left(v^n_{i,j+3/2}+v^n_{i,j-1/2}+v^n_{i+1,j+1/2}\right)\\ &+ \left(v^n_{i-1,j+1/2}-4v^n_{i,j+1/2}\right)\\ &- \frac{1}{\rho}h\left(p_{i,j+1}-p_{i,j}\right)\\ u^{n+1}_{i+1,j}-u^{n+1}_{i-1/2,j} &+ v^{n+1}_{i,j+1/2}-v^{n+1}_{i,j-1/2}=0\end{aligned}$$

All previous equation can be in term of $\mathbf{u}$ i.e in vector form, and we have exactly the following system: The divergence condition is approximated numerically using: $\nabla.u=\lim_{\varepsilon\to 0}\frac{1}{\varepsilon}\oint\mathbf{u}.\mathbf{n}ds$
This allows us to define a discrete vector equations

$$\begin{aligned}\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\delta t} &= -M^n_{i,j}-\nabla P_{i,j}+N^n_{i,j} &(12)\\ \nabla.u^{n+1}_{i,j} &= 0 &(13)\end{aligned}$$

Equation (12) represent the evolution of the velocity and Equation (13) represent the constraint on velocity. We notice that there is no explicit equation for the pressure. To avoid this problem we will use projection technique. Equations (12), (13) will be solved using projection technique. We will introduce an auxiliary vector $u^{'}$ so that (12),(13) can be rewritten as:

$$\begin{aligned}\frac{u^{'}_{i,j}-u^n_{i,j}}{\delta t} &= -M^n_{i,j}+N^n_{i,j} &(14)\\ \frac{u^{n+1}_{i,j}-u^{'}_{i,j}}{\delta t} &= -\nabla P_{i,j} &(15)\end{aligned}$$

Taking the divergence of $\frac{u^{n+1}_{i,j}-u^{'}_{i,j}}{\delta t}=-\nabla P_{i,j}$ divergence of one can obtain an equation for the pressure

$$\Delta P_{i,j}=\frac{1}{\delta t}\nabla.u^{'}_{i,j}$$

So, we are going to solve the new system obtained using Projection technique:

$$\begin{aligned}u^{'}_{i,j} &= u^n_{i,j}+\delta t\left(-M^n_{i,j}+N^n_{i,j}\right) &(16)\\ \Delta P_{i,j} &= \frac{1}{\delta t}\nabla.u^{'}_{i,j} &(17)\\ u^{n+1}_{i,j} &= u^{'}_{i,j}-\delta t\nabla P_{i,j} &(18)\end{aligned}$$

In our discretization we will consider these parameters: the

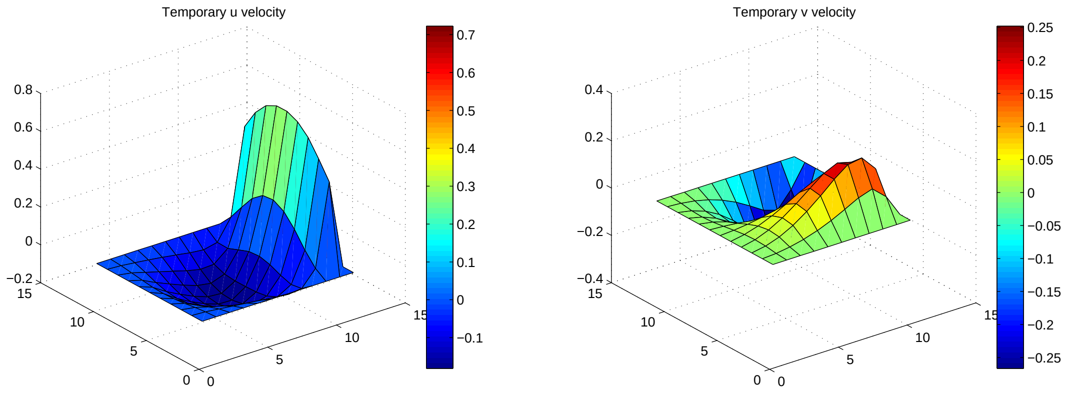
- Coefficients: $\mu=0.1$, $\beta=1.2$;
- $n_x=10$: number of nodes in $x$-direction
- $n_y=10$; number of nodes in $y$-direction
- $dt=0.02$; time step-size
- $nstep=200$; maximum number of step in time
- $maxit=100$; maximum of iteration

**Table 1:** Pressure values

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | -0.0642 | -0.0538 | -0.0555 | -0.0810 | -0.1446 | -0.2687 | -0.4956 | -0.9101 | -1.6758 | 0 |
| 0 | -0.0747 | -0.0665 | -0.0736 | -0.1033 | -0.1652 | -0.2709 | -0.4288 | -0.6255 | -0.7493 | 0 |
| 0 | -0.0727 | -0.0650 | -0.0722 | -0.0981 | -0.1480 | -0.2253 | -0.3220 | -0.4022 | -0.3800 | 0 |
| 0 | -0.0550 | -0.0504 | -0.0560 | -0.0743 | -0.1087 | -0.1595 | -0.2152 | -0.2452 | -0.2096 | 0 |
| 0 | -0.0269 | -0.0275 | -0.0306 | -0.0389 | -0.0551 | -0.0795 | -0.1040 | -0.1105 | -0.0912 | 0 |
| 0 | 0.0020 | -0.0035 | -0.0033 | 0.0006 | 0.0066 | 0.0142 | 0.0265 | 0.0463 | 0.0524 | 0 |
| 0 | 0.0211 | 0.0128 | 0.0167 | 0.0333 | 0.0649 | 0.1149 | 0.1861 | 0.2683 | 0.3003 | 0 |
| 0 | 0.0235 | 0.0147 | 0.0200 | 0.0449 | 0.0991 | 0.1976 | 0.3598 | 0.5908 | 0.7954 | 0 |
| 0 | 0.0124 | 0.0011 | 0.0004 | 0.0217 | 0.0817 | 0.2113 | 0.4690 | 0.9604 | 1.8416 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 2:** Velocity *u*

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2.0000 |
| 0.0034 | -0.0034 | -0.0135 | -0.0228 | -0.0323 | -0.0432 | -0.0553 | -0.0634 | -0.0292 | 0.2630 | 1.7370 |
| 0.0151 | -0.0151 | -0.0414 | -0.0648 | -0.0870 | -0.1065 | -0.1152 | -0.0890 | 0.0440 | 0.4750 | 1.5250 |
| 0.0283 | -0.0283 | -0.0699 | -0.1051 | -0.1363 | -0.1576 | -0.1515 | -0.0801 | 0.1304 | 0.5984 | 1.4016 |
| 0.0370 | -0.0370 | -0.0878 | -0.1299 | -0.1657 | -0.1866 | -0.1685 | -0.0667 | 0.1851 | 0.6571 | 1.3429 |
| 0.0374 | -0.0374 | -0.0888 | -0.1317 | -0.1691 | -0.1917 | -0.1735 | -0.0672 | 0.1931 | 0.6662 | 1.3338 |
| 0.0292 | -0.0292 | -0.0718 | -0.1092 | -0.1444 | -0.1706 | -0.1655 | -0.0835 | 0.1498 | 0.6245 | 1.3755 |
| 0.0156 | -0.0156 | -0.0427 | -0.0683 | -0.0951 | -0.1211 | -0.1337 | -0.0982 | 0.0638 | 0.5108 | 1.4892 |
| 0.0034 | -0.0034 | -0.0136 | -0.0238 | -0.0357 | -0.0510 | -0.0679 | -0.0741 | -0.0196 | 0.2890 | 1.7110 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2.0000 |



(a) temporary *u* velocity

(b) temporary *v* velocity

**Figure 2:** (a): Temporary *u* velocity used in projection method ((b): Temporary *v* velocity used in projection method

Equations (16), (17), (18) will be solved using a refined grid and we get the following figures: Figures 3(a) and 3(b) shows vectors specified by $u$ and $v$ at equally spaced points in the $(x,y)$ plane. Figure 3(c) shows the pressure $p$ and Figure 3(d) shows velocity vectors as arrows with components $(u,v)$. Tables **??**, **??** and **??** gives the value of our unknowns function in the grid, one can see that we have used a $p=0$ on the boundary of the domain $[-1,1]^2$. Figures 2(a) and 2(b) gives the temporary $u$ and $v$ velocity used in the projection technique.

## 4. Conclusion

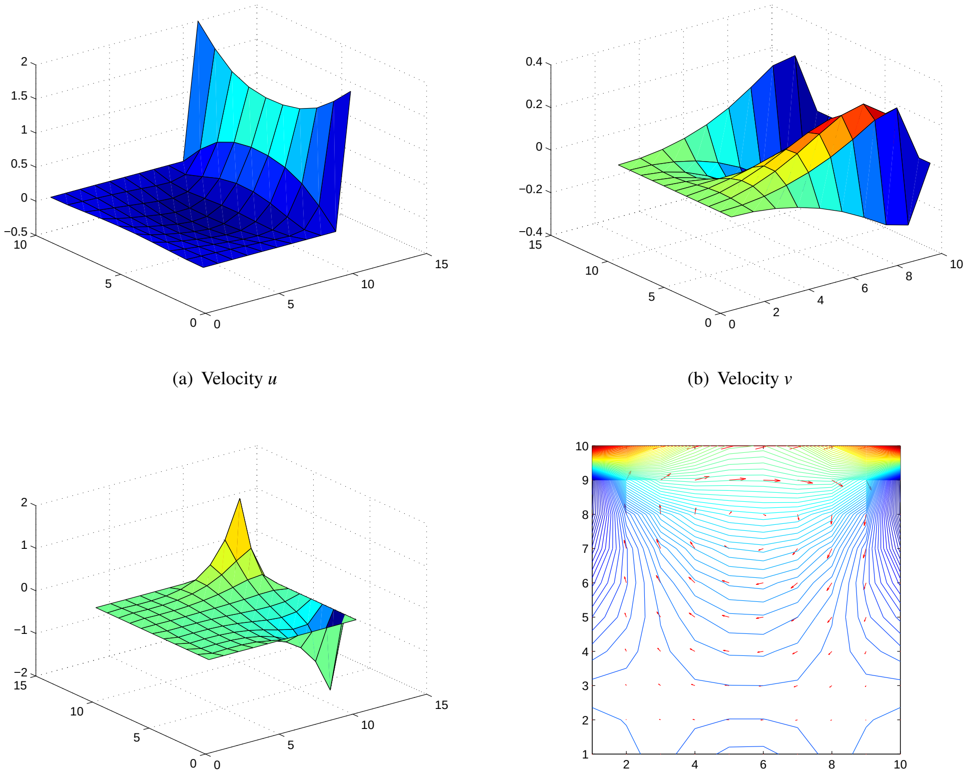
The Navier Stokes equation is solved using two different method based on variational form and Finite element method. Numerical solution of Navier stokes equation is always not obvious because it implies re-scaling those values to get a figure for the computer that is considered as the standard tool at the time of comparison. Furthermore, this kind of procedure will conclude to the detriment of spectral methods. Since they are global methods, each computational node is coupled to more neighbors than standard finite difference, finite volume or finite element methods and, therefore, imply more computational work per grid point. In real computations, the user should give the numerical error bar of the simulation in order to promote better results. We consider that this will bring an enormous improvement in assessing fluid flow computational results. The control volume approach bring us the solution of the Navier stokes equation as well the system of Navier stokes equation. In the next work we are going to consider the case of Navier stokes equation in three dimensional case. Also, we will try to solve Navier Stokes equation in two dimensional singular domain using Domain decomposition method and artificial boundary condition.

## Acknowledgement

I would like to thank Al-Qunfudah center Scientific Research for the most support and encouragement.

**Table 3:** Velocity *v*

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0 | -0.0034 | -0.0168 | -0.0396 | -0.0720 | -0.1151 | -0.1704 | -0.2339 | -0.2630 | 0 |
| 0 | 0.0034 | 0.0168 | 0.0396 | 0.0720 | 0.1151 | 0.1704 | 0.2339 | 0.2630 | 0 |
| 0 | 0.0117 | 0.0396 | 0.0816 | 0.1363 | 0.1997 | 0.2596 | 0.2851 | 0.2119 | 0 |
| 0 | 0.0132 | 0.0418 | 0.0821 | 0.1313 | 0.1824 | 0.2187 | 0.2098 | 0.1235 | 0 |
| 0 | 0.0086 | 0.0266 | 0.0513 | 0.0808 | 0.1098 | 0.1268 | 0.1134 | 0.0587 | 0 |
| 0 | 0.0004 | 0.0014 | 0.0031 | 0.0065 | 0.0116 | 0.0166 | 0.0171 | 0.0091 | 0 |
| 0 | -0.0082 | -0.0251 | -0.0476 | -0.0723 | -0.0933 | -0.1012 | -0.0849 | -0.0417 | 0 |
| 0 | -0.0136 | -0.0427 | -0.0837 | -0.1330 | -0.1825 | -0.2144 | -0.1998 | -0.1137 | 0 |
| 0 | -0.0122 | -0.0413 | -0.0858 | -0.1452 | -0.2153 | -0.2811 | -0.3051 | -0.2218 | 0 |
| 0 | -0.0034 | -0.0170 | -0.0408 | -0.0765 | -0.1274 | -0.1953 | -0.2694 | -0.2890 | 0 |
| 0 | 0.0034 | 0.0170 | 0.0408 | 0.0765 | 0.1274 | 0.1953 | 0.2694 | 0.2890 | 0 |



(a) Velocity *u*

(b) Velocity *v*

(c) Pressure *p*

(d) Velocity vectors $(u,v)$

**Figure 3:** Solution of two dimensional flow Navier stokes equation (a): Velocity *u* ((b): Velocity *v*, (c): pressure, and (d) Velocity vectors as arrows with components $(u,v)$.

## References

[1] Galdi, Giovanni P. An Introduction to the Mathematical Theory of the Navier-Stokes Equations. *Springer Monographs in Mathematics* 2011.

[2] Temam, Roger. Navier-Stokes equations and nonlinear functional analysis. *CBMS-NSF regional conference series in applied mathematics, publisher Society for Industrial and Applied Mathematics, Philadelphia*, ISBN 0-89871-183-5, year = 1983.

[3] A.J. Chorin. On the convergence of discrete approximations to the Navier-Stokes equations. *Math. Comp*, 23 (1969), pp. 341-353.

[4] R. Temam. Sur l'approximation de la solution des equations de Navier-Stokes par la méthode des pas fractionnaires II. *Arch. Rat. Mech. Anal.*, 33 (1969), pp. 377-385.

[5] Chorin, A. J. The numerical solution of the Navier-Stokes equations for an incompressible fluid. *Bull. Am. Math. Soc.* 73 (1967) 928-931.

[6] Chorin, A. J.; J. E. Marsden. A Mathematical Introduction to Fluid Mechanics (3rd ed.). *Springer-Verlag.* (1993) ISBN 0-387-97918-2.

[7] Chorin, A. J. A numarical method for solving imcompressible viscous flow problems. *Journal of computational physics* 135, 118-125 (1997).

[8] F.H. Harlow and J.E. Welch. *Phys. Fluids* 8, 2182 (1965).

[9] Francis H Harlow, Anthony A Amsden. A numerical fluid dynamics calculation method for all flow speeds. *Journal of Computational physics.* Volume 8, Issue 2,(1971) 197-213.

[10] Claude Brezinski. Projection methods for systems of equations. *Studies in computational mathematics* 7. ISBN 0444 82 7773.

[11] J.N. Reddy. An introduction to the finite element method. T*hird edition. McGraw Hill Newyork* 2005.

[12] Klaus Jurgen Bathe. Finite element procedures. *Prentice hall Pearson Education,Inc.* ISBN 978 0979004902.

[13] E. M.Ronquist. A Domain Decomposition Solver for the Steady Navier-Stokes Equation. *ICOSAHOM'95: Proceedings of the Third International Conference on Spectral and High Order Methods. 1996 Houston Journal of Mathematics, University of Houston.*

[14] X.C. Cai and O.B. Widlund, Multiplicative Schwarz algorithms for some nonsymmetric and indefinite problems. *SIAM J. Numer. Anal.*, 30(4), pp. 936-952 (1993).