INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 4, Issue 1)
Available online at www.ijariit.com

Comparative Analysis of Discretization Schemes

Naveen T
naveent922@gmail.com
Nitte Meenakshi Institute of Technology, Bengaluru, Karnataka

ABSTRACT

In the field of CFD, there are mainly two different types of discretization methods are used one is structured grid-based and the other one is unstructured grid-based, the structured grid is known for the accuracy and the unstructured grid is known for the ease with which it can be generated around complex configuration. An emerging third approach is the use of Meshless discretization which does not involve a mesh in the classical sense but works on a random distribution of points. Creating points around a configuration should be easier compared to generating grids hence, in this work we compare finite volume method with the Meshless method using a simple convection-diffusion equation whose exact solution is known.

Keywords: *Meshless Method, Finite Volume Method.*

1. INTRODUCTION

CFD is a branch of science which deals with obtaining the numerical solution of partial differential equations governing fluid flow. With the development of computers and increase in computational power, industrial relevance problems can be solved by this method. Unstructured mesh based finite volume methods have gained a lot of popularity in solving industrial problems mainly for their flexibility in handling geometrical complexities. This has led to researchers proposing unstructured mesh-based methodologies to solve inviscid and viscous flows. In the past decade, a new method of discretization called Mesh less method has been gaining popularity since it only requires a distribution of points on the domain for obtaining the solution. In real life problems generating a grid for complex configuration is very difficult and is also expensive. One of the earliest work pertains to paper on "Generalized finite difference method" by Chung [1]. In this work, we are comparing finite volume method with the meshless method.

2. FINITE VOLUME METHOD

In this method, the domain under consideration is discretized into a number of volumes which are non-overlapping, called finite volumes. The points in the domain where the unknowns are present are called control points. In this work, we have used a cell-centered approach where the cell centroids are the control points.

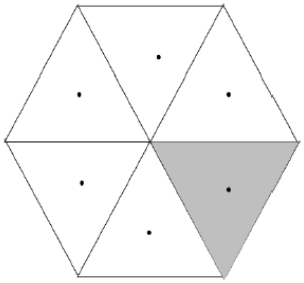

Fig -1: Volume Of Cell-Centered Method

2.1 Problem Definition

The convection-diffusion equation in 2D is shown in equation 1. This equation is solved on a square domain defined by vertices (0, 0), (0, L), (L,0), (L, L). Also, we can see from the figure 2 the domain is divided into a number of finite volumes with different mesh configurations ranging from 106cells to 6784cells.

$$\frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} + v\frac{\partial \phi}{\partial y} = \alpha\left[\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2}\right] \quad \text{…. Eqn (1)}$$

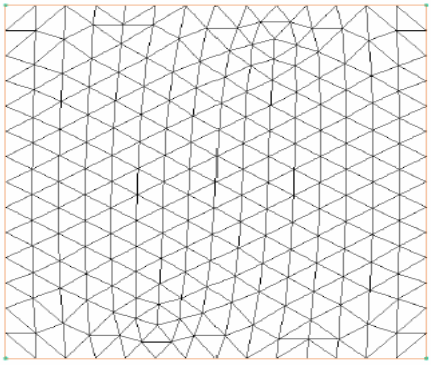

Fig-2: Domain with 424 Cells.

In the equation 1
u = x-component of velocity.
v = y-component of velocity.
α = diffusive co-efficient.
L = length of the domain.
Here, α can be written as
$\alpha = uL/\text{Pe}$
Where Pe = Peclet number which is defined as

$$Pe = \frac{convective\ strength}{diffusive\ strength}$$

The convection-diffusion equation is solved in the above domain. We have used 4 different mesh configurations having 106cells, 424cells, 1696cells, and 6784cells.

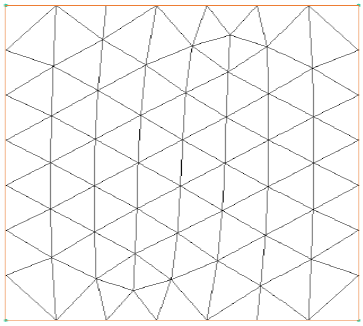

Fig -3: Mesh with 106 cells.

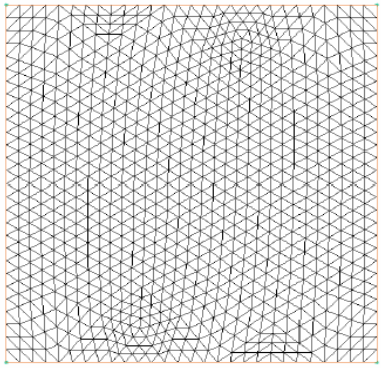

Fig-4: Mesh with 424 Cells.

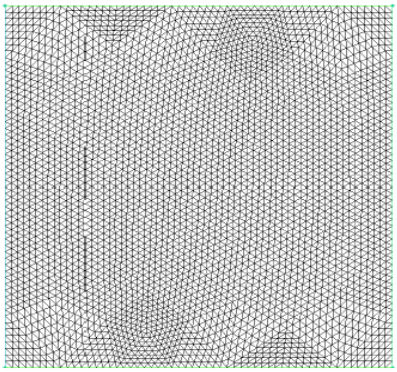

Fig-5: Mesh with 6784 Cells.

In the equation 1
(u,v) = $(u,0)$
The boundary conditions are given as:

- $\emptyset(0,y) = sin(\pi y)$ At the inlet.
- $\emptyset(x,0) = \emptyset(x,L) = 0$ At the walls.
- $\left.\frac{\partial \emptyset}{\partial y}\right|_{x=L} = 0$ At the outlet.

The equation is solved for 2 Peclet numbers: 1 and 100.
The exact solution is given as:

$$\emptyset_{exact} = \sin(\pi y)\left[\frac{r_2 * \exp(r_1 x + r_2 l) - r_1 * \exp(r_2 x + r_1 l)}{r_2 * \exp(r_2 l) - r1 * \exp(r_1 * \exp(r_1 l)}\right]$$

Where, $r_{1,2} = \frac{u}{2\alpha} \pm \sqrt{\frac{U^2}{4\alpha^2} + \pi^2}$

2.2 General Approach

Initialization: the solution is initialized as

- $\emptyset(0,y) = \sin(\pi y)$ At inlet.
- $\emptyset_i = 0$ For all cell centroid and vertex points.

Input: The solver is written for above-mentioned meshes, the grids were generated using GAMBIT software. n_p, n_c, n_e Where n_p is number of points, n_c number of cells, n_e and number of edges is the input data which is extracted from the mesh data.

Vertex Value Finding: In this step, the $\emptyset$ values of all internal vertices were updated using an area weighted average of neighboring vertices whereas the values of the outlet vertices were updated using the outlet boundary condition.

Green-Gauss Procedure: Green-Gauss procedure is implemented and edge sweeping was done to calculate $(\nabla\emptyset)_J$ over the covolumes. The interface viscous flux values are calculated by these values and were also passed to the neighboring cells to calculate $\nabla\emptyset_j$ at the cell centroids using diamond path reconstruction.

$$(\nabla\emptyset)_J = \frac{1}{\Omega_J}\sum_k\left(\sum_p w_p \emptyset_k^p\right)\hat{n}_k \Delta S_k. \qquad \text{…eqn2}$$

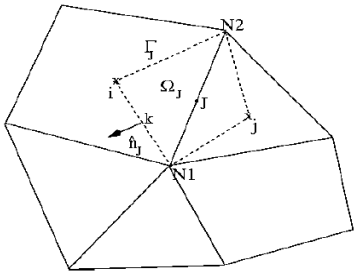

Fig-6: Diamond Path Covolume

Convective Flux Calculation: after all the edges are swept the ∇Φ values at the cell centroids were available. Using this the left and right values at each edge were reconstructed which were used to calculate the convective fluxes at each interface using up winding (CIR scheme).

Explicit Time Stepping: At this stage, the net flux at each interface was available and then this was used to calculate$\Delta_t \emptyset^{explicit}$. By giving a suitable Courant number based on convection process.

Convergence Criterion: The convergence criterion is residue based, and is given by,

Res (n) =$\sum_{i,j}\left(\emptyset_{i,j}^{n} - \emptyset_{i,j}^{n-1}\right)^2$

If convergence criterion is cc, the simulation stops when Res (n)/Res (1) $\leq 10^{-cc}$. In these simulations a, value of 8 is chosen for cc.

Output: output is a global error which is calculated as below.

$$error = \sqrt{\frac{\sum_{i=1}^{nc}(\emptyset_{exact} - \emptyset_{num})^2}{nc}}$$

Language: the program is written in Fortran and Tecplot 360 is used for plotting results.

3. MESHLESS METHOD

One of the alternate methods to solve flow over complex geometries is a meshless method. This involves discretizing the domain into a cluster of points in this work. The meshless method works on the principles of generalized finite difference. The solution variation in the neighborhood of a point given by truncated Taylor's series. The sum of squares of errors is minimized to obtain gradients of inviscid flux, gradients, and hessians of field variable $\emptyset$. Hessians are used for approximating viscous flux derivatives. The 2D Convection-diffusion equation is solved on a square domain as shown below.

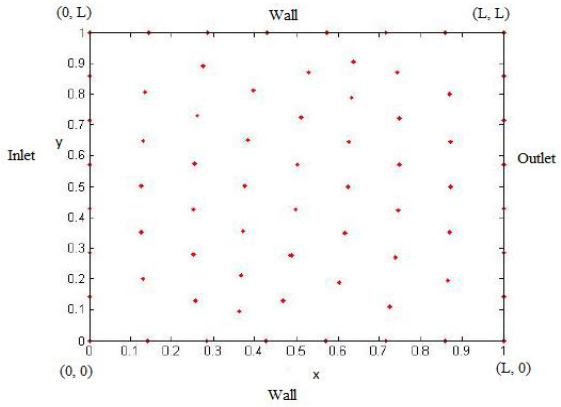

Fig -7: Point Distribution

The convection-diffusion equation is solved on a square domain defined by the vertices (0, 0), (0, L), (L, 0), (L, L) as shown in figure 2.2. Also, the domain contains a distribution of a number of random points where the solution is obtained.

3.1 Problem Definition

In the equation 1

(u,v) = $(u,0)$

The boundary conditions are given as:

- $\emptyset(0,y) = sin(\pi y)$ At the inlet.
- $\emptyset(x,0), \emptyset(x,L) = 0$ At the walls.
- $\left.\frac{\partial \emptyset}{\partial y}\right|_{x=L} = 0$ At the outlet.

The equation is solved for 2 Pe numbers: 1 and 100.

The same 4 meshes mentioned before are used as the domain to solve the convection-diffusion equation. In this method, the solution is calculated on the vertex points unlike on the cell centers on the Finite Volume Method.

A number of random points: 68, 241, 905, 3505.

3.2 General Approach

Initialization: The solution is initialized as

- $\emptyset(0,y) = \sin(\pi y)$ At inlet.
- $\emptyset_i = 0$ For all points.

Convective Term Discretization: The Convective term is discretized by using Upwind-Least Squares Finite Difference method (LSFD-U). The equation is as given below.

$$f_{xi} = \frac{\sum_J w_J \Delta y_J^2 \ \sum_J w_J \Delta f_J \Delta x_J - \sum_J w_J \Delta x_J \Delta y_J \ \sum_J w_J \Delta f_J \Delta y_J}{\sum_J w_J \Delta x_J^2 \ \sum_J w_J \Delta y_J^2 - \left(\sum_J w_J \Delta x_J \Delta y_J\right)^2}$$

Using linear least square procedure, solution gradients are obtained at points which are used for second order discretization scheme.

Diffusive Term Discretization: The diffusive term is discretized using Quadratic Least Squares procedure [4]. This process involves the use of Quadratic Least squares procedure to obtain second derivatives of the solution at the point.

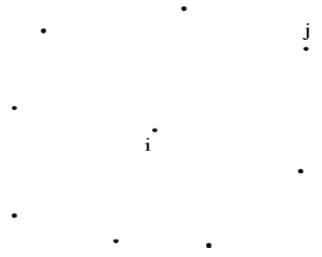

Fig-8: A Cluster of Points.

In the above figure point 'i' is the point of interest and the points 'j' are the neighboring points, the solution estimate at neighboring point j is approximated using Taylor series with terms retained upto second order.

$$\emptyset_j^e = \emptyset_i \Delta x_j + \emptyset_{yi} \Delta y_j + \emptyset_{xxi} \frac{\Delta x_j^2}{2} + \emptyset_{xyi} \Delta x_j \Delta y_j + \emptyset_{yyi} \frac{\Delta y_j^2}{2}$$

Error associated with point 'j' is

$$E_j = \Delta \emptyset_j - \left\{ \emptyset_{xi} \Delta x_j + \emptyset_{yi} \Delta y_j + \emptyset_{xxi} \frac{\Delta x_j^2}{2} + \emptyset_{xyi} \Delta x_j \Delta y_j + \emptyset_{yyi} \frac{\Delta y_j^2}{2} \right\}$$

The solution derivatives are obtained by minimizing $\sum_j E_j^2$ which results in the matrix vector equation.

By solving the above system of equations we get the values of the solution vector.

x = $\left[\emptyset_{xi}, \emptyset_{yi}, \emptyset_{xxi}, \emptyset_{xyi}, \emptyset_{yyi}\right]$

Second derivatives of the solution obtained are first-order accurate.

Explicit Time Stepping: Inviscid flux gradients and viscous flux gradients are available at all nodes. The update formula is as given below

$$\emptyset_i^{n+1} = \emptyset_i^n - \Delta t * \left((f_{xi}) - \frac{1}{pe} (\emptyset_{xxi} + \emptyset_{yyi}) \right)$$

Value of time increment is decided by CFL condition.

$$dt = \frac{Cr * \min(\Delta r)}{u}$$

Where Δr is minimum distance between points in the entire domain?

Convergence Criterion: The convergence criterion is residue based, and is given by,

Res (n) = $\sqrt{\sum_i (\emptyset_i^n - \emptyset_i^{n-1})^2}$

If convergence criterion is cc, the simulation stops when Res (n) $\leq 10^{-cc}$. In these simulations a, value of 8 is chosen for cc.

Output: Output is a global error which is calculated as below.

$$error = \sqrt{\frac{\sum_{i=1}^{np} (\emptyset_{i,exact} - \emptyset_{i,num})^2}{np}}$$

4. RESULTS

The contour plots for the Peclet number 1 and 100 are as shown below. The exact solution contour and numerical solution contours are shown subsequently.

4.1 Finite Volume Method.

Exact solution plot

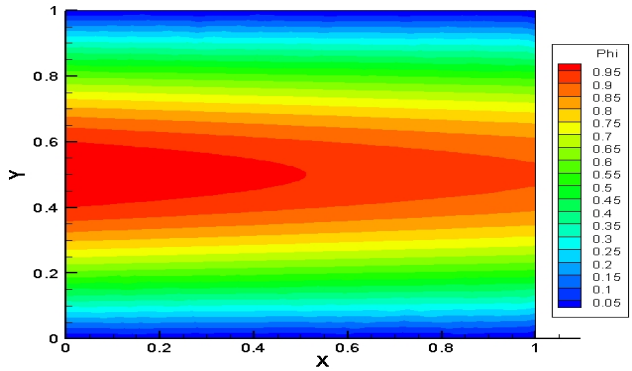

Fig-9: Pe=100 for 6784 Cells.

Numerical solution plot

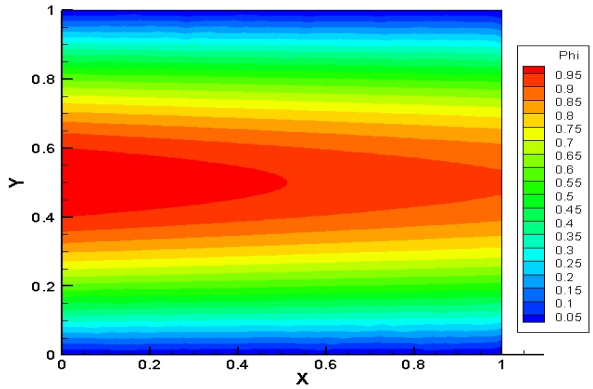

Fig-10: Pe=100,6784cells.

Exact solution plot

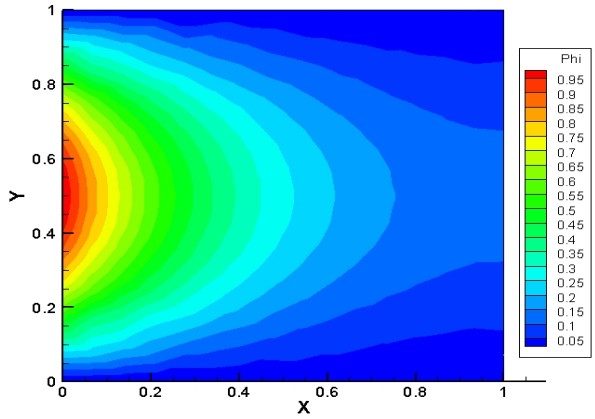

Fig-11: Pe=1, 424cells

Numerical solution plot

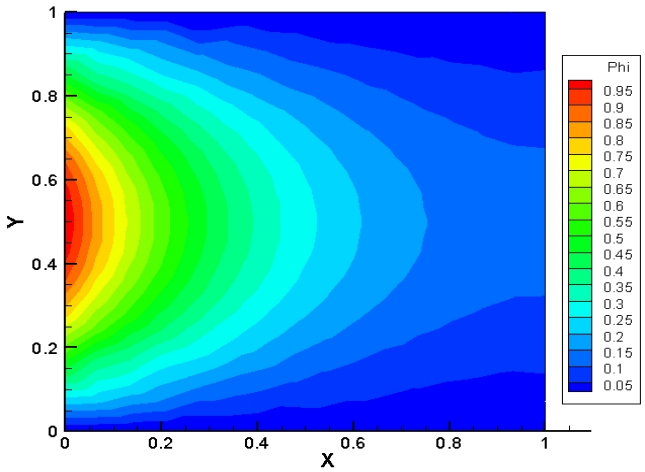

Fig-12: Pe=1,424 cells

Error Plots

A graph is plotted with the natural logarithm of error on the y-axis and the natural logarithm of the square root of (1/nc) on the x-axis. Where nc = number of cells.

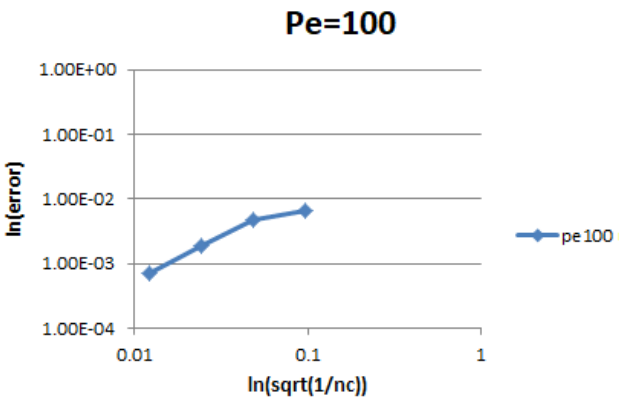

Fig-13: Plot for Pe=100

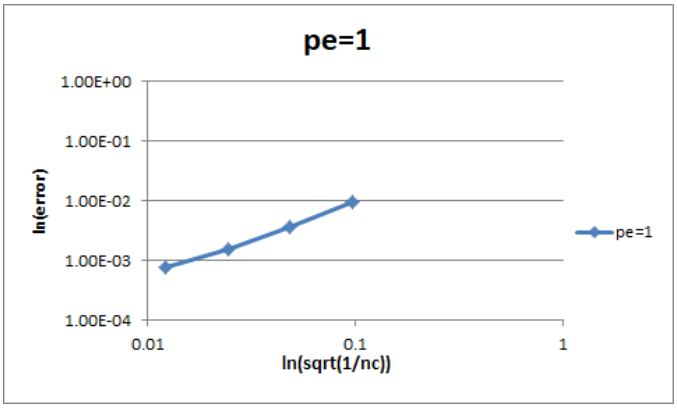

Fig-14: Plot for Pe=1

When we take the slope of the above curves we get the rate of error drop.

Table-1: Error Fall Rate

Pe=1	1.13
Pe=100	1.09

Here we can see that as the number of cells increases the error decreases.

4.2 Meshless Method

The contour plots for the Peclet number 1 and 100 are as shown below. The exact solution contour and numerical solution contours are shown below.

Numerical solution contours

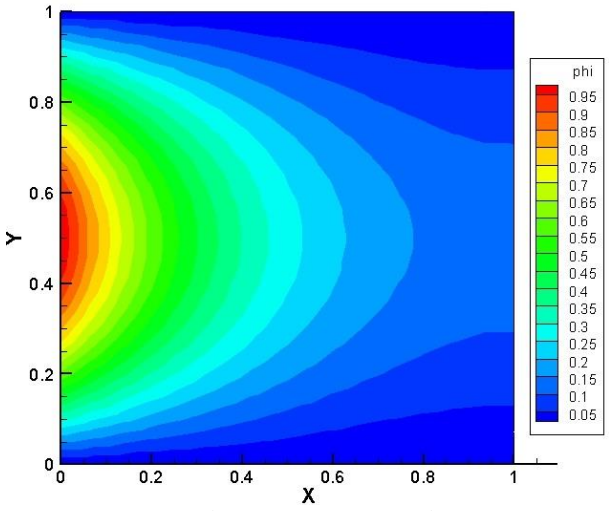

Fig-15: Pe=1, 241 Points

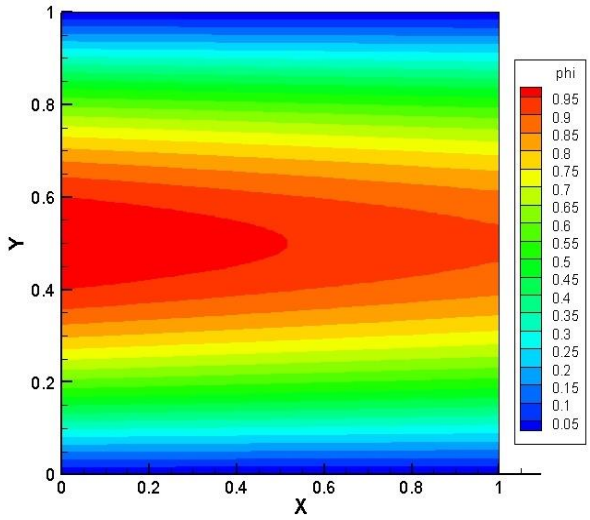

Fig-16: Pe=100, 3505 Points

Exact solution contours

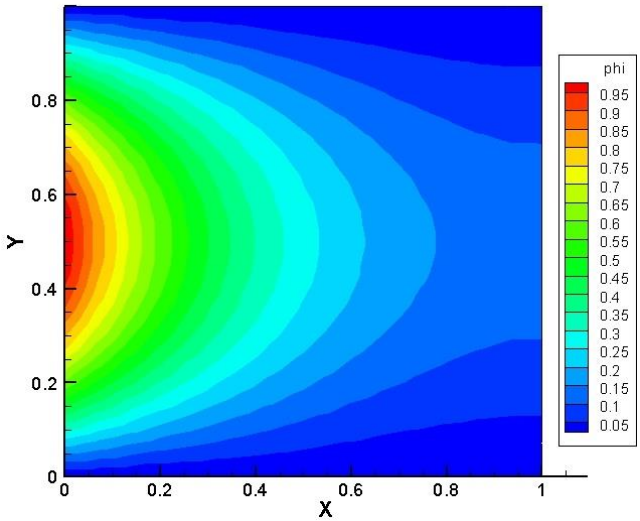

Fig-17: Pe=1, 241 Points

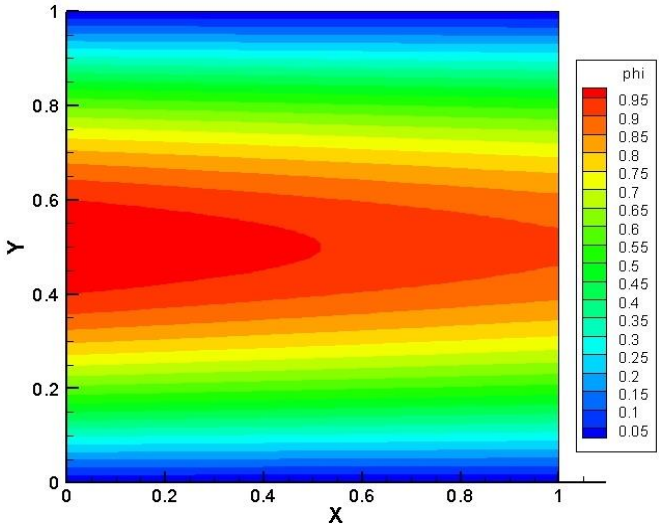

Fig-18: Pe=100, 3505 Points

Error Plots

A graph is plotted with the natural logarithm of error on the y-axis and the natural logarithm of (1/np) on the x-axis. Where np = number of points.

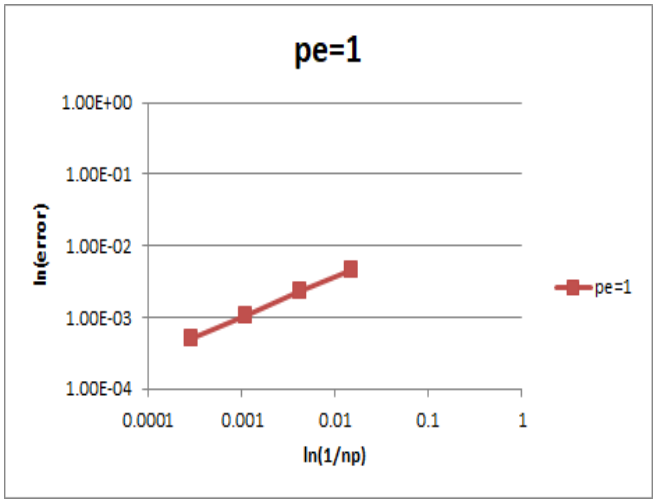

Fig-19: Plot for Pe=1

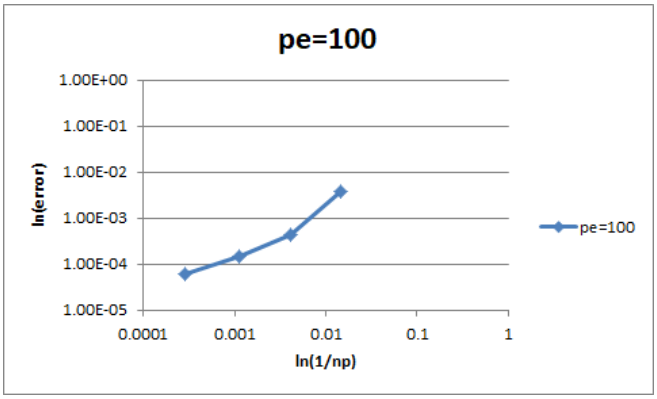

Fig 20: Plot for Pe=100

When we take the slope of the above curves we get the rate of error drop.

Table-1: Error Fall Rate

Pe=1	0.61
Pe=100	0.75

5. COMPARISION

In order to compare the two methods, we are plotting the error plots of the solution with the scale of Finite Volume Method. We are fixing the error to a particular level and plotting an error line on the graph the error line cuts the two curves at specific points from which we drop perpendicular lines to the x-axis this gives the grid resolution required by Finite Volume Method to match the accuracy of Mesh less Method using the same unstructured grid.

The plots of Error vs. Square root of (1/nc) are as below.

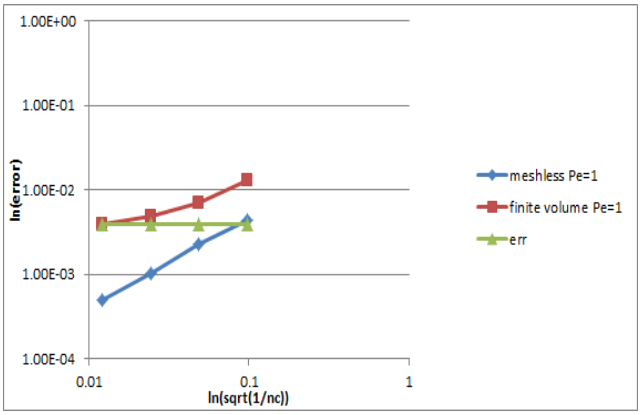

Fig-20: Error Plots for Pe=1

In figure 20, the error line shown as 'err' in the figure is fixed at 4.0E-3 this line cuts the two curves at distinct when we drop perpendiculars from these points to x-axis we get h1, h2. Where, h1= grid resolution of the Meshless method, h2= grid resolution of Finite Volume method.

When Pe=1, from the graphs we can tell that for a fixed error level of 4.0E-3 the Mesh less method, requires a grid with 135 cells whereas Finite Volume method requires 5850 cells.

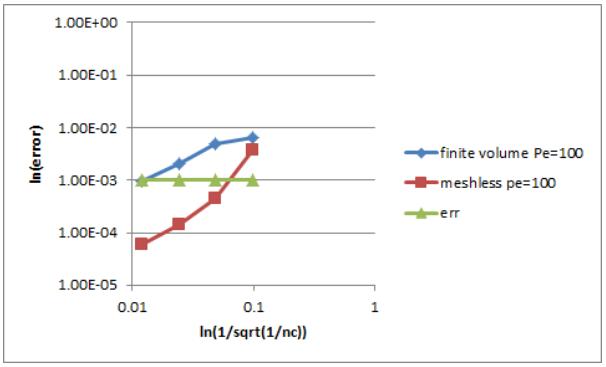

Fig-21: Error plots for Pe=100

From figure 21. In the similar way as described above we can infer that for Pe=100, from the graphs we can tell that for a fixed error level of 1.0E-3 the Meshless method requires a grid with 890 cells whereas Finite Volume method requires 6250 cells.

Table-3: Error Fall Rates

Pe=1, Finite Volume	0.57
Pe=1, Meshless method	1.06

Table-4: Error Fall Rates

Pe=100, Finite Volume	0.93
Pe=100, Meshless method	1.96

6. CONCLUSION

We can clearly see from the results that for a given error level a coarser grid can be used for Mesh less method compared to Finite volume method. Also generating points on the complex configurations is easier compared to generating a traditional mesh. Since a coarser grid can be used in the Meshless method it also saves a lot of Computational time required to get the solution.

7. REFERENCES

[1]. Chung K.C., "A generalized finite difference method for heat transfer problems of irregular geometries" Numerical Heat Transfer, Vol. 4, pp. 345-357, 1981.
[2]. Anderson J.D.Jr., "Computational Fluid Dynamics, the basics with applications", McGraw-Hill Inc., Singapore, 1995.
[3]. Anup S. Ninawe, Munikrishna N., and Balakrishnan N., "Viscous flow computations using the meshless solver, LSFD-U", David Zingg ed., Computational Fluid Dynamics 2004, Springer-Verlag, Berlin, Germany, pp. 509–514, 2004.
[4]. Balakrishnan N., "New least squares based finite difference method", Report No. 99 FM 9, Department of Aerospace Engineering, Indian Institute of Science, Bangalore, India, 1999.

BIOGRAPHY/BIOGRAPHIES

Naveen T
Student

A student of final year M tech, Thermal Power Engineering at the Visvesvaraya Technological University, Bangalore, Karnataka.

