

Application The Pressure Correction Method for Incompressible Viscous Flow

Alireza Baheri¹, Ahmad Navaserarab²

¹ Department of Mechanical Engineering, Islamic Azad University, Dezful Branch, Dezful , Iran

² Department of Mechanical Engineering, Islamic Azad University, Dezful Branch, Dezful , Iran

Abstract

In the paper, we will focus on the use of the pressure correction method to solve incompressible, viscous flow. Incompressible flow is governed by elliptic partial differential equations, and the relaxation technique, which is essentially an iterative process, is a classical numerical method for solving elliptic problems. In contrast, viscous, incompressible flow is governed by the incompressible Navier-Stokes equations, which exhibit a mixed elliptic parabolic. The purpose of the present section is to describe an iterative process called the pressure correction technique, which has found widespread application in the numerical solution of the incompressible Navier-Stokes equations. The pressure correction technique has been developed for practical engineering solutions by Patankar and Spalding. The technique is embodied in an algorithm called SIMPLE (semiimplicit method for pressure-linked equations), pioneered by Patankar and Spalding, which has found widespread application over the past 20 years for both compressible and incompressible flows.

Keywords: Relaxation technique, Incompressible and viscous flow, Simple method, Navier-Stokes equations.

1. Introduction

In the present section, we will apply this method to the solution of the incompressible, viscous flow between two parallel plates as sketched in Fig. 1. The upper plate is located a distance H above the lower plate and is in motion with velocity u_e relative to the lower plate. Although the plates are theoretically infinite in extent, the computational domain is finite, with length L and height H , as shown by the shaded region in Fig. 1. We will treat the boundary conditions around this finite computational domain in the same fashion, with p and v fixed and u allowed to float at the inflow

boundary, and with only p fixed at the outflow boundary.

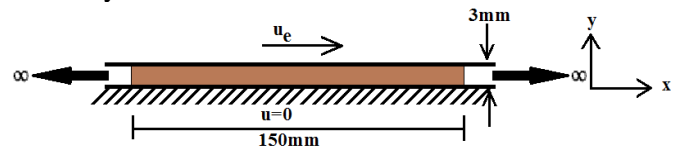


Fig 1: The finite computational domain for the application of the pressure correction method for the solution of the incompressible flow between two plates in relative motion

The pressure correction method is an iterative method, starting from arbitrarily assumed initial conditions. We will induce a two-dimensional flow within the computational domain by setting the initial conditions to be an arbitrary two dimensional flow field. Then, during the iterative procedure, we will watch this originally two-dimensional flow field converge to the exact solution for Couette flow.

2. The Numerical Approach

The physical problem is sketched in Fig. 2. We will carry out the present solution in dimensional terms, rather than nondimensionalizing the governing equations and dealing with a nondimensional space. We offer this calculation in part as an example that CFD solutions are frequently carried out using dimensional terms throughout the calculation. Hence, as shown in Fig. 2, we will treat a computational domain which is 152 mm long in the x direction and 3 mm high in the y direction. The upper plate is moving with velocity u_e , and the lower plate is stationary. The fluid is air at standard sea level conditions, with a density $\rho = 1.05 \text{ Kg} / \text{m}^3$. Since we will employ a very coarse grid for the example, we treat the case of a low velocity; e.g., we set $u_e = 0.3 \text{ m} / \text{s}$ for the present calculation. At this low velocity, there is absolutely no doubt about the assumption that the flow is incompressible. Also, nothing is to be gained

in terms of the objective of this example by considering higher values of u_e . Based on the height D of 3mm, the Reynolds number for this case is 63.6. There are three systems of grid points shown in Fig. 2; the solid points are where p is calculated, the open points are where u is calculated, and the points denoted by x are where v is calculated. The use of a staggered grid requires careful attention to the indexing system that identifies each set of points, and somewhat complicates the coding of the computer program.

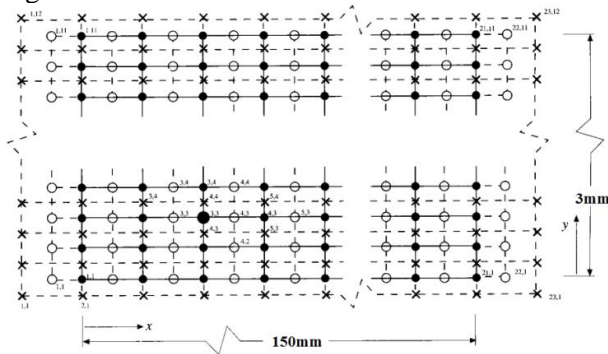


Fig 2: Staggered computational grid. p points, solid circles; u points, open circles; v points, x

The pressure correction method is an iterative solution for the flow field. Hence, we need to set the initial conditions for the flow variables in order to start the iterative process. The choice is arbitrary. For the present calculation, we set the following initial conditions on all interior points, except for point $(i,j) = (15,5)$, which will be addressed later:

$$u = v = 0$$

$$p^* = p' = 0$$

The specification of the initial conditions for the pressure correction p' equal to zero seems reasonable. But why is the pressure itself, p^* , set equal to zero? The answer is—simply for convenience. An examination of the x - and y -momentum equations, respectively, shows that only the pressure difference between adjacent grid points appears. Therefore, the individual values of p^* are not so important—it is the pressure difference that counts. Therefore, it is totally appropriate to set $p^*=0$ for the initial conditions, because the pressure difference will be dictated by the values of the pressure correction calculated for subsequent iterations.

The boundary values are as follows:

$$u = u_e \quad \text{and} \quad v = 0 \quad \text{upper wall}$$

$$u = v = 0 \quad \text{lower wall}$$

$$p' = 0 \quad \text{and} \quad v = 0 \quad \text{inf low boundary}$$

$$p' = 0 \quad \text{outflow boundary}$$

Step 1. Guess at values of p^* at all interior grid points. Also, arbitrarily set values of $(pu)^n$ and $(pv)^n$ at all the appropriate grid points. As stated above, p^* , pu^* , and pv^* are all set to zero for the beginning of the iterative process, except for $u_e = 0.3m/s$ at the upper wall and for $v_{15,5}^* = 0.15m/s$ at the velocity spike.

Step 2. Solve for $(pu^*)^{n+1}$ from Eq.1 and $(pv^*)^{n+1}$ from Eq.2 at all interior grid points.

$$\frac{\partial(pu)}{\partial t} + \frac{\partial(pu^2)}{\partial x} + \frac{\partial(puv)}{\partial y} = -\frac{\partial p}{\partial x} + \mu \left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right]$$

$$\frac{\partial(pv)}{\partial t} + \frac{\partial(puv)}{\partial x} + \frac{\partial(pv^2)}{\partial y} = -\frac{\partial p}{\partial y} + \mu \left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right]$$

Using a forward difference in time and forward differences and central difference in space, Eq.1 and Eq.2 becomes

$$\begin{aligned} & \frac{(pu)_{i,j}^{n+1} - (pu)_{i,j}^n}{\Delta t} + \frac{(pu^2)_{i+1,j}^n - (pu^2)_{i,j}^n}{\Delta x} + \\ & \frac{(puv)_{i,j+1}^n - (puv)_{i,j}^n}{\Delta y} = -\frac{p_{i+1,j}^n - p_{i,j}^n}{\Delta x} + \\ & \mu \left[\frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{(\Delta x)^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{(\Delta y)^2} \right] \\ & \frac{(pv)_{i,j}^{n+1} - (pv)_{i,j}^n}{\Delta t} + \frac{(puv)_{i+1,j}^n - (puv)_{i,j}^n}{\Delta x} + \\ & \frac{(pv^2)_{i,j+1}^n - (pv^2)_{i,j}^n}{\Delta y} = -\frac{p_{i+1,j}^n - p_{i,j}^n}{\Delta y} + \\ & \mu \left[\frac{v_{i+1,j}^n - 2v_{i,j}^n + v_{i-1,j}^n}{(\Delta x)^2} + \frac{v_{i,j+1}^n - 2v_{i,j}^n + v_{i,j-1}^n}{(\Delta y)^2} \right] \end{aligned}$$

$$(pu)_{i,j}^{n+1} = (pu)_{i,j}^n - \Delta t \frac{(pu^2)_{i+1,j}^n - (pu^2)_{i,j}^n}{\Delta x} - \Delta t \frac{(puv)_{i,j+1}^n - (puv)_{i,j}^n}{\Delta y} - \Delta t \frac{p_{i+1,j}^n - p_{i,j}^n}{\Delta x} + \mu \Delta t \left[\frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{(\Delta x)^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{(\Delta y)^2} \right]$$

$$(pv)_{i,j}^{n+1} = (pv)_{i,j}^n - \Delta t \frac{(puv)_{i+1,j}^n - (puv)_{i,j}^n}{\Delta x} - \Delta t \frac{(pv^2)_{i,j+1}^n - (pv^2)_{i,j}^n}{\Delta y} - \Delta t \frac{p_{i+1,j}^n - p_{i,j}^n}{\Delta y} + \mu \Delta t \left[\frac{v_{i+1,j}^n - 2v_{i,j}^n + v_{i-1,j}^n}{(\Delta x)^2} + \frac{v_{i,j+1}^n - 2v_{i,j}^n + v_{i,j-1}^n}{(\Delta y)^2} \right]$$

$$(pu^*)_{i+1,j}^{n+1} = (pu^*)_{i,j}^n + A\Delta t - \frac{\Delta t}{\Delta x} (p_{i+1,j}^* - p_{i,j}^*) \quad (1)$$

$$(pv^*)_{i,j+1}^{n+1} = (pv^*)_{i,j}^n + B\Delta t - \frac{\Delta t}{\Delta y} (p_{i,j+1}^* - p_{i,j}^*) \quad (2)$$

$$A = -\frac{(pu^2)_{i+1,j}^n - (pu^2)_{i,j}^n}{\Delta x} - \frac{(puv)_{i,j+1}^n - (puv)_{i,j}^n}{\Delta y} + \mu \left[\frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{(\Delta x)^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{(\Delta y)^2} \right]$$

In the above equations, the values of Δx , Δy , and

$$B = -\frac{(puv)_{i+1,j}^n - (puv)_{i,j}^n}{\Delta x} - \frac{(pv^2)_{i,j+1}^n - (pv^2)_{i,j}^n}{\Delta y} + \mu \left[\frac{v_{i+1,j}^n - 2v_{i,j}^n + v_{i-1,j}^n}{(\Delta x)^2} + \frac{v_{i,j+1}^n - 2v_{i,j}^n + v_{i,j-1}^n}{(\Delta y)^2} \right]$$

Δt for the present calculations are

$$\Delta x = 3/30 = 0.1mm$$

$$\Delta y = 150/15 = 10mm$$

$$\Delta t = 0.001s$$

The value of Δt was chosen somewhat arbitrarily. However, if Δt is chosen to be too large, experience with the present calculation shows that the calculation becomes unstable. Examining Eqs. 1 and 2, we see that Δt plays the role of a "relaxation factor"; the larger Δt , the larger is the change in pu^* and pv^* from one iteration to the next. It seems reasonable that if this change becomes too large, instabilities could arise. The value $\Delta t = 0.001s$ was found to be acceptable for the present calculation; no effort was made to optimize this value.

Step 3. Using the values for pu^* and pv^* obtained from step 2, solve for p' fiord the pressure correction formula, Eq. 3, repeated below,

$$ap'_{i,j} + bp'_{i+1,j} + bp'_{i-1,j} + cp'_{i,j+1} + cp'_{i,j-1} + d = 0 \quad (3)$$

$$a = 2 \left[\frac{\Delta t}{\Delta x^2} + \frac{\Delta t}{\Delta y^2} \right]$$

$$b = -\frac{\Delta t}{\Delta x^2}$$

$$c = -\frac{\Delta t}{\Delta y^2}$$

$$d = \frac{1}{\Delta x} [(pu^*)_{i+1,j} - (pu^*)_{i-1,j}] + \frac{1}{\Delta y} [(pv^*)_{i,j+1} - (pv^*)_{i,j-1}]$$

Step 4. Calculate p^{n+1} at all internal grid points from Eq. 4

$$p_{i,j}^{n+1} = (p^*)_{i,j}^n + \alpha p' \quad (4)$$

where α is an underrelaxation factor. In the present calculations, the value of α was set as 0.1:

Step 5. Designate the values of $p_{i,j}^{n+1}$ obtained from step 4 as the new values of $(p^*)^n$ to be inserted into the equivalents of Eq. 1 and 2 written at all interior grid points. Return to step 2, and repeat steps 2 to 5 until convergence is achieved. For the present calculation, convergence of this primary iteration loop

was achieved after approximately 500 iterations. Again, no effort was made to optimize the calculation so as to obtain the smallest number of iterations necessary for convergence.

Because of the insertion of the v velocity spike at point $(i, j) = (10, 2)$ in the initial conditions, the flow field is two-dimensional during the ensuing iterations. This is illustrated in Fig. 3, which shows v profiles as a function of distance y across the duct at the axial station where $i = 10$. Hence, these profiles include the grid point $(10, 2)$ where the initial value of the velocity spike, $v = 0.15 \text{ m/s}$, was inserted. Indeed, this velocity spike is shown in Fig. 3 by the dashed line at $y = 1.2 \text{ mm}$ extending to a value of $v = 0.15 \text{ m/s}$. In Fig. 3, R denotes the iteration number, hence the velocity spike at the zero the iteration (the initial conditions) is denoted by $R = 0$. Three other velocity profiles are shown in Fig. 3, each one the

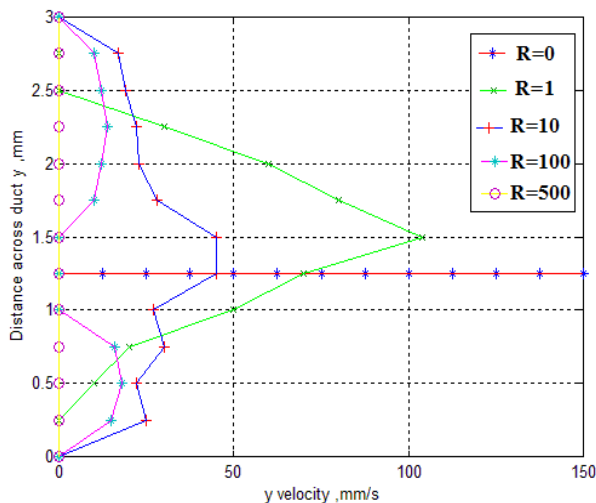


Fig 3: Profiles of the y component t of velocity v across the duct at the axial station denoted by $i = 10$. Profiles are shown at venous

stages during the iterative process. The iteration number is denoted by R corresponding to the results obtained after R iterations. Note that the peak value of v has already been reduced to 0.104 m/s after only one iteration, as seen in the profile labeled $R = 1$. The profile labeled $R = 10$ shows that the peak value of v continues to be reduced and that finite values of v are spreading both

upward and downward away from grid point $(10, 2)$; indeed, the region of two-dimensional flow introduced by the velocity spike is spreading throughout the flow field in both the x and y directions, although the magnitude of v progressively gets smaller as the iterations progress. Note in Fig. 3 that the profile labeled $R=100$ shows a major reduction in v by the end of the fiftieth iteration. Finally, after 500 iterations, when convergence is obtained throughout the entire

flow field, v has essentially gone to zero at all grid points. Reflecting on the results shown in Fig. 3, the pressure correction formula, Eq. 3 and its equivalent at each grid point, is certainly doing its intended job it is setting up a pressure field that pushes the velocity field in the correct direction, in this case in the direction of v going toward zero.

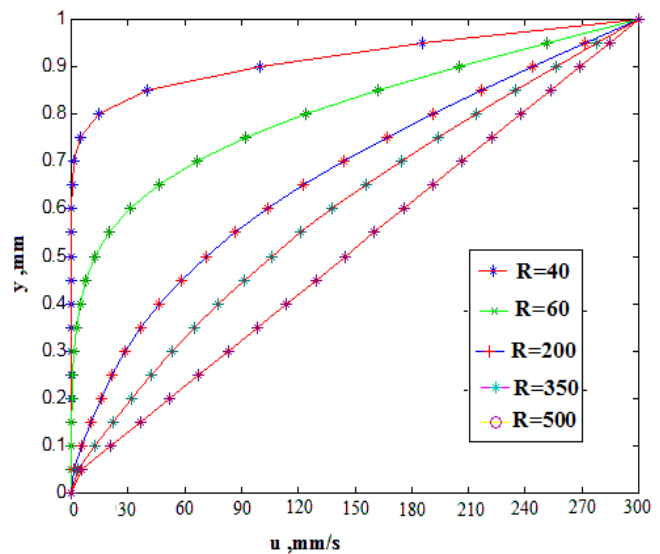


Fig 4: Velocity profiles for the x component of velocity as a function of vertical distance across the duct. Profiles are shown for various iteration numbers, ranging from 40 to 500. At $R = 300$, the velocity profile has converged to the Couette flow solution.

3. Conclusion

In summary, in the present section we have illustrated the use and behavior of the pressure correction method for the solution of an incompressible viscous

flow. From our results, it is interesting to observe the relative roles played by pressure and viscosity in the formulation of the velocity field. In Fig. 3 we see the vertical velocity spike decaying fairly rapidly; the values of v throughout the whole flow field became small after about 80 iterations. The rapid decay of v is due to pressure gradients being set up in the flow and propagating via pressure waves that move rapidly throughout the flow field. Again, the calculated pressure corrections are acting to rapidly reduce v . In contrast, in Fig. 4 we see the horizontal velocity profiles more slowly converging to the proper solution. Here, the values of u are dominated by viscosity (shear stress), the effects of which are propagated more slowly than those due to pressure waves. Indeed, the values of u do not converge to the proper solution until about 500 iterations, well after the values of v have become very small. This numerical behavior is directly analogous to actual physical behavior in real flows. Flow fields are driven under the impetus of pressure gradients and shear stress, and generally the influence of pressure propagates more rapidly throughout the flow field than that of viscosity.

REFERENCES

- [1] Anderson, John D., Jr.: *Computational Fluid Dynamics, The Basics With Applications*, McGraw-Hill, New York, 1989.
- [2] Anderson, John D., Jr.: *Hypersonic and High Temperature Gas Dynamics*, McGraw-Hill, New York, 1989.
- [3] Rouse, Hunter, and Simon Ince: *History of Hydraulics*, Iowa Institute of Hydraulic Research, Ames, Iowa 1957.
- [4] Tokaty, G. A.: *A History and Philosophy of Fluid Mechanics*, G. T. Foulis, Henly-on-Thames, England, 1971.
- [5] Anderson, John D., Jr.: *The History of Aerodynamics, and Its Impact on Flying Machines*, Cambridge University Press, New York (in preparation).
- [6] Kothari, A. P., and I. D. Anderson, Jr.: "Flows Over Low Reynolds Number Airfoils-s-Compressible Navier-Srokes Numerical Solutions," AIAA paper 85-0107, presented at AIAA 23rd Aerospace Sciences Meeting, Reno, Nev., Jan. 14-17, 1985.
- [7] Pohlen, L. 1., and T. 1. Mueller: "Boundary Layer Characteristics of the Miley Airfoil at Low Reynolds Numbers," *J Aircr.*, vol. 21, no. 9, pp. 658-664, September 1984.
- [8] Anderson, John D., Jr.: *Fundamentals of Aerodynamics*, 2d ed., McGraw-Hill, New York, 1991.
- [9] Bush, Richard J., Jr., Merle Jager, and Brad Bergman: "The Application of Computational Fluid Dynamics to Aircraft Design," AIAA paper 86-2651, 1986.
- [10] Moretti, G., and M. Abbett: "A Time-Dependent Computational Method for Blunt Body Flows," *AIAA J*, vol. 4, no. 12, pp. 2136-2141, December 1966.
- [11] Anderson, Dale A., John C. Tannehill, and Richard H. Pletcher: *Computational Fluid Mechanics and Heat Transfer*, McGraw-Hill, New York, 1984.
- [12] Fletcher, C. A.: *Computational Techniques for Fluid Dynamics*, vol. I: *Fundamental and General Techniques*, Springer-Verlag, Berlin, 1988.
- [13] Fletcher, C. A.: *Computational Techniques for Fluid Dynamics*, vol. II: *Specific Techniques for Different Flow Categories*, Springer-Verlag, Berlin, 1988.
- [14] Hirsch, Charles: *Numerical Computation of internal and External Flows*, vol. 1: *Fundamentals of Numerical Discretization*, Wiley, New York, 1988.
- [15] Hirsch, Charles: *Numerical Computation of Internal and External Flows*, vol. 11: *Computational Methods for Inviscid and Viscous Flows*, Wiley, New York, 1990.
- [16] Hoffmann, K. A.: *Computational Fluid Dynamics for Engineers*, Engineering Education System, Austin, Tex., 1989.