Finite-difference and -volume methods for the incompressible Navier-Stokes equations

Rauan

December 14, 2023

Abstract

The goal of these notes is to review the finite-difference schemes for solving incompressible Navier-Stokes equations, with the emphasis on those which are well-established and tested by time. We will discuss in detail the scheme formulation, its advantages and shortcomings including stability and accuracy aspects. Special attention will be paid to the boundary conditions and their implementation. Of particular interest is to highlight the motivation behind the scheme and to trace the historic development of finite-difference methods as applied to incompressible flows through the contributions of the key papers in this field. After studying these notes one must get a coherent picture of Finite-difference and -volume approaches to incompressible flows as well as to see a clear logical connection among methods.

Contents

1 Introduction

In these notes we will be concerned with the discretization of the Navier-Stokes equations describing the flow of an incompressible fluid:

$$\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v} \cdot \nabla \boldsymbol{v} = -\nabla p + \nu \nabla \cdot \nabla \boldsymbol{v} \tag{1a}$$

$$\nabla \cdot \boldsymbol{v} = 0, \tag{1b}$$

which are written here in the non-dimensional form, i.e. $\nu \equiv Re^{-1}$ for brevity; also $\boldsymbol{v}$ is the velocity and p pressure fields. Since both are necessarily functions of time t and space $\boldsymbol{x}$, their discretization will be denoted by f_i^n, where n is the time level t^n and i is the *mesh* point $\boldsymbol{x}_i$: for example, in the x-direction the interval $x_i < x < x_{i+1}$ is referred to as the *cell* of the mesh. The cell centre corresponds to $x_{i+\frac{1}{2}}$. The mesh points x_i and cell centres $x_{i\pm\frac{1}{2}}$ can be regarded as two overlapping interpenetrating meshes, which together are said to constitute a *staggered* mesh as illustrated in figure 2(a). When designing a finite difference or finite volume scheme, we have to choose whether to use the same or different sets of grid points for velocity and pressure. The obvious choice seems to have a single set of points, at which all the variables and all the equations are discretized. Such a grid has the name of a *collocated* or *regular* grid. Albeit simple and easy in operation, the collocated grids were out of favor for a long time because of their tendency to generate spurious oscillations in the solution (cf. §10.2 in [28]) and hence staggered grids are used since the work of Harlow and Welch [10] introducing the marker-and-cell (MAC) method, cf. figure 2(a).

The staggered arrangement increases complexity of a scheme. Programming becomes more difficult, since it requires accounting for three (or four in the three-dimensional case) indexing systems. Interpolations must be used to compute nonlinear terms of momentum equations. Further complications arise when the grid is nonuniform. All these difficulties, however, can be relatively easily handled in computations with structured grids such as those shown in figure 2(b). For this reason and because of the benefit of removing the splitting problem, the staggered arrangement was by far the most popular choice during early years of CFD. The difficulties of handling a staggered arrangement increase significantly when unstructured grids are used. When such grids started to be broadly applied in general-purpose codes in recent years, collocated arrangements returned to favor. This area of CFD is still evolving. We only mention that methods have been developed to cure the splitting problem leading to pressure oscillations, but the cure is not ideal and leads to extra complexities at the implementation level.

In numerical terms, the problem (1) can be formulated as follows: given the solution p^n, $\boldsymbol{v}^n$ at the previous time layer t^n, find the next time-layer pressure p^{n+1} and velocity $\boldsymbol{v}^{n+1}$ such that they together satisfy the momentum equation, and the velocity is divergence-free $\nabla \cdot \boldsymbol{v}^{n+1} = 0$ and satisfies the boundary conditions.

A conspicuous feature of (1) is that pressure p is not determined by a time-evolution equation, but rather is implicitly defined by the incompressibility condition (1b), which plays the role of a constraint that the velocity field $\boldsymbol{v}$ must satisfy. The pressure instantaneously adapts to the evolving velocity field in such a way as to satisfy that constraint. This is reflected in the fact that p satisfies a Poisson equation, which can be derived by taking the divergence of (42a) and combining the result with (1b). From

the mathematical viewpoint, this means that the incompressible flow equations have some features of an elliptic system. We can say that the equations are of the mixed hyperbolic (convective terms), parabolic (viscous terms), and elliptic (pressure and incompressibility) type. The elliptic nature of the pressure solution has a physical meaning. It shows that, in an incompressible flow, the pressure field in the entire flow domain adjusts instantaneously to any, however localized, perturbation. This is in perfect agreement with the fact that weak perturbations, for example sound waves, propagate at infinite speed in incompressible fluids.

Given the Poisson equation for p, it may replace the continuity equation (1b) and its solution can in principle be substituted back into equation (42a) to obtain an evolution equation for the velocity field alone. Alternatively, the pressure can be immediately eliminated by taking the curl of momentum equation (42a), which leads to the vorticity-stream function formulation of incompressible flow. Formal manipulations of this type are useful for various theoretical purposes, but experience has shown that they are rarely advantageous for computational purposes. In most situations, it is preferable to simply approximate and solve the system (1) directly for the so-called *primitive* variables $\boldsymbol{v}$ and p.

If, however, one ends up solving the Poisson equation for pressure, an interesting and important question arises as to what boundary conditions should be used for the pressure field. Such conditions are required at every point of the boundary for the Poisson problem to be well-posed. The conditions, however, do not naturally follow from the flow physics for the boundaries between fluid and solid walls, unless, of course, a full fluid-structure interaction problem is solved. Since the latter option is, in most cases, an unnecessary complication, we have to find a way to derive the pressure boundary conditions from the equations themselves.

- ~~Discuss in detail the origin of spurious oscillations on collocated grids~~
- ~~Discuss the boundary conditions for pressure, cf. §10.3.1 in~~ [28]~~, and how to avoid dealing with the latter.~~
- Consultation appointment to discuss this paper's mistakes and grammar. 15Nov 2PM.

ToDo

1.1 Oscillations on collocated grids

To give a simple and explanatory example to see the reasons for oscillatory behaviour consider a 1-Dimensional ($\boldsymbol{v} = u$) continuity equation (1b) solved on both Staggered and Collocated Grids (figures 1(a) and 1(b) respectively) with velocities $u_{Left} = 1$ and $u_{Right} = 2$ at the boundaries in Finite Volume formulation.

First, consider the staggered arrangement of the cells. Continuity equation (1b) for cell "O" in figure 1(a) after integrating over the volume and application of Gauss-Ostrogradsky theorem becomes

$$\int_O (\nabla \cdot u) dV \equiv (u_e^O - u_w^O)\Delta y = 0, \tag{2}$$

where u_e^O, u_w^O represent values of the velocity u at the faces of the "O" cell, solving the equation (2) leads to $u_e^O = u_w^O$. By repeating the process for all control volumes the solution for all cells results in $u = 1$ for all cell faces.

Moving to the collocated grid arrangement in figure 1(b) and solving continuity equation (2) requires interpolation for values of u at cell faces. By applying linear interpolation to discretized continuity equation (2) and assuming uniform grid:

$$u_e^O = \frac{u_E + u_O}{2}, \quad u_w^O = \frac{u_W + u_O}{2}. \tag{3}$$

This leads to the identity $u_E = u_W$, which can be satisfied by infinitely many solutions for velocity u at cell centres, couple of examples could be:

u =[1,0,1,0,1,0,...],

u =[0,1,0,1,0,1,...],

u =[1,1,1,1,1,1,...] (the correct solution),

u =[1,0.3,1,0.3,1,0.3,...],

and infinitely many more.

To display the infinite solution behaviour from above rigorously, we shall construct a system of linear equations based on a collocated mesh. Consider cell centres at grid coordinates x_{i-1}, x_i, x_{i+1}. Assume that the values of u at the cell centre depend only on neighbouring cells W, E (can be extended to further neighbours such as EE, WW etc. if needed) and cell O. Denoting these x coordinates as W, O, E respectively, the continuity equation (1b) can be represented for i^{th} cell that is not at the boundary

$$A_W^i u_W + A_O^i u_O + A_E^i u_E = 0, \tag{4}$$

where A_W, A_O, A_E represent the coefficients in front of the velocity terms for each cell.

By taking equations ((2),(3)) the coefficients in equation (4) can be obtained, resulting in:

$$A_W^i = -\frac{1}{2}, A_O^i = 0, A_E^i = \frac{1}{2}, \quad \text{where} \quad i = 2, 3, \ldots, n-1.$$

The equation (4) changes for the cells next to boundaries:

$$A_w^1 u_w + A_O^1 u_O + A_E^1 u_E = 0,$$
$$A_W^n u_W + A_O^n u_O + A_e^n u_e = 0,$$

with coefficients:

$$A_w^1 = -1, A_O^1 = \frac{1}{2}, A_E^1 = \frac{1}{2},$$
$$A_W^n = -\frac{1}{2}, A_O^n = -\frac{1}{2}, A_e^n = 1.$$

The values at the boundaries u_w^1 and u_e^n are known, and thus can be moved to the right-hand side of the corresponding equations, which leads to the system of linear equations

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & \ldots & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ldots & -\frac{1}{2} & 0 & \frac{1}{2} \\ 0 & \ldots & 0 & -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \\ u_4 \\ \vdots \\ u_{n-2} \\ u_{n-1} \end{bmatrix} = \begin{bmatrix} (1)(u_w^1) \\ 0 \\ 0 \\ \vdots \\ 0 \\ (-1)(u_e^n) \end{bmatrix}. \tag{5}$$

The system (5) can be satisfied by many different combinations since it is singular. The correct solution is one of the infinitely many possible ones, therefore the numerical scheme keeps flip-flopping, such phenomena are also known as spurious oscillations or checkerboard oscillations.

The checkerboard oscillations were overcome by the introduction of Staggered Grids [10] in 1965 which were used until Rhie & Chow's (1983) [22] proposal of a special interpolation technique on collocated grids.

The oscillatory behaviour is usually observed for pressure variable in system (1) [28], to display this we take x-component of momentum equation (42a) and by denoting $\boldsymbol{D}$, all but pressure terms we integrate over the finite volume of cell O,

$$\boldsymbol{D}u = -\frac{\partial p}{\partial x} \implies \sum_{faces} (F^d + F^c) \cdot \boldsymbol{n} dA = -\left.\frac{\partial p}{\partial x}\right|_O V_O, \tag{6}$$

where F^d, F^c are diffusive and convective fluxes of u, $\boldsymbol{n}$ is an outward normal vector to the surface dA enclosing the finite volume of cell O.

After application of any interpolation scheme (central, upwinding etc.) to fluxes on the left side of discretized momentum (6), one is able to express the value of u at the cell centre as a linear function of its neighbours $\lambda_N^O u_O = f(u_N^O)$, where λ_N^O is an interpolating coefficient. Expressing u_O in the discretized momentum equation (6) leads to

$$u_O = \frac{1}{\lambda_N^O}\left(f(u_N^O) - \left.\frac{\partial p}{\partial x}\right|_O V_O\right). \tag{7}$$

(a) Staggered (b) Collocated

Figure 1: Staggered vs Collocated grids.

By substituting the above result (7) into the discretized continuity (2), we obtain

$$\frac{1}{\lambda_E^O}\left(f(u_N^E) - \frac{\partial p}{\partial x}\bigg|_E V_E\right) - \frac{1}{\lambda_N^W}\left(f(u_N^W) - \frac{\partial p}{\partial x}\bigg|_W V_W\right) = 0. \tag{8}$$

The discretized pressure term $\frac{\partial p}{\partial x}\big|_O = \frac{p_e - p_w}{\Delta x} = \frac{p_E - p_W}{2\Delta x}$ substituted into the semi-discretized finite volume continuity equation (8), leads to

$$\frac{1}{\lambda_E^O}\left(f(u_N^E) - \frac{p_{EE} - p_O}{2\Delta x}V_E\right) - \frac{1}{\lambda_W^O}\left(f(u_N^W) - \frac{p_O - p_{WW}}{2\Delta x}V_W\right) = 0, \tag{9}$$

where pressure with lower subscripts p_{EE}, p_O, p_{WW} represent the values of pressure at the cell centres with coordinates $i+2, i, i-2$.

From the discretized equation (9) above it is evident that the value of p_O does not depend on the values of its neighbours p_E, p_W, but only on it's further neighbours p_{EE}, p_{WW}. Therefore two solutions for pressure (one for even and another for odd cells) may satisfy initial system (1).

1.2 Boundary conditions for pressure

An interesting and important question arises as to what boundary conditions should be used for the pressure field. Such conditions are required at every point of the boundary for the Poisson problem to be well-posed

Applying $(\nabla\cdot)$ to the momentum equation (42a), first taking into account the commutativity of time derivative and Laplace operator ∇^2 with $(\nabla\cdot)$, then eliminating these terms by applying continuity equation (1b), finally leads to Poisson equation for pressure

$$\nabla\cdot(\nabla p) \equiv \nabla^2 p = \rho\nabla\cdot[-\boldsymbol{v}\cdot\nabla\boldsymbol{v}]. \tag{10}$$

The conditions, however, do not naturally follow from the flow physics for the boundaries between fluid and solid walls. We have to find a way to derive the pressure boundary conditions from the equations themselves.

Recalling calculus: the gradient of any function f is defined as the unique vector field whose dot product with any vector v at each point x is the directional derivative of f along v, i.e. $\nabla f\cdot v = \frac{\partial f}{\partial v}$.

Consider $\boldsymbol{n}$ to be the normal vector to the solid wall boundary. We apply $(\cdot\boldsymbol{n})$ to the momentum equation (42a) and get

$$\frac{\partial p}{\partial \boldsymbol{n}}\bigg|_{\partial\Omega} = \rho\left[\nu\nabla\cdot\nabla\boldsymbol{v} - \frac{\partial \boldsymbol{v}}{\partial t} - \boldsymbol{v}\cdot\nabla\boldsymbol{v}\right]\cdot\boldsymbol{n}\bigg|_{\Omega}. \tag{11}$$

To compute the pressure values at the boundaries we need to take into account the impermeability condition. The velocity component perpendicular to the wall is zero, meaning that the flow can not penetrate the wall, mathematically written as

$$\boldsymbol{v}\cdot\boldsymbol{n}\Big|_{\partial\Omega} = 0. \tag{12}$$

By taking into account impermeability condition (12) equation (11) leads to boundary condition for the pressure at the wall computed as

$$\frac{\partial p}{\partial \boldsymbol{n}}\bigg|_{\partial\Omega} = \rho\left[\nu\nabla\cdot\nabla\boldsymbol{v} - \frac{\partial \boldsymbol{v}}{\partial t} - \boldsymbol{v}\cdot\nabla\boldsymbol{v}\right]\cdot\boldsymbol{n}\bigg|_{\Omega} \implies \frac{\partial p}{\partial \boldsymbol{n}}\bigg|_{\partial\Omega} = \rho\left[\nu\nabla\cdot\nabla\boldsymbol{v}\right]\cdot\boldsymbol{n}\bigg|_{\partial\Omega}. \tag{13}$$

The diffusive term on the right-hand side is nonzero even after the application of no-slip conditions $\boldsymbol{v} \cdot \boldsymbol{t}\Big|_{\partial\Omega} = 0$ at the wall, where $\boldsymbol{t}$ is a tangential vector to the wall, i.e. diffusion at the wall is directed in the normal direction from the boundary.

Boundary conditions for pressure (11) and (13) define a solution of the Poisson equation for pressure (10) up to a constant, i.e. we need to fix value for p at any point of our domain. Another method to obtain boundary conditions for pressure is to apply $(\boldsymbol{t}\cdot)$ to the momentum equation (42a) which will lead to Dirichlet boundary condition, where the value of pressure at a single point x_0 is also needed to uniquely determine the boundary values. Gresho [8] describes the equivalence of the derived Dirichlet and Neumann conditions for the Pressure Poisson equation.

The results obtained for pressure boundary conditions require special treatment and recomputation for every change of velocity near the boundaries, which increases the complexity of solvers and generates more errors. Section 9 displays an approach that completely eliminates pressure from the problem statement, thus removing the requirement of its boundary conditions.

2 Marker-and-cell scheme

The system (1) does not involve $\partial p/\partial t$, hence, the pressure cannot be advanced in time by means of its time derivative. In fact, the only place p appears in the equations at all is in the pressure gradient term in the momentum equation (42a). If this term were differenced in a purely explicit manner, the scheme would not involve p^{n+1} in any way, so there would be no way to compute it. In order to advance the pressure in time, the temporal difference approximation to ∇p in (42a) must be at least partially implicit so that p^{n+1} appears in the difference scheme. This in turn implies that the temporal difference approximation to $\nabla \cdot \boldsymbol{v}$ in (1b) must likewise be at least partially implicit since the difference approximation to (42a) alone does not provide enough equations to determine both $\boldsymbol{v}^{n+1}$ and p^{n+1}. But once $\nabla \cdot \boldsymbol{v}$ is partially implicit, it might as well be fully implicit, which entails no additional labour and has the significant advantage of satisfying (1b) exactly at each time level of the calculation. We, therefore, difference (42a) in a fully implicit manner

$$(\nabla \cdot \boldsymbol{v})^{n+1} = 0. \tag{14}$$

The momentum equation can be differenced in a fully explicit manner except for the pressure gradient

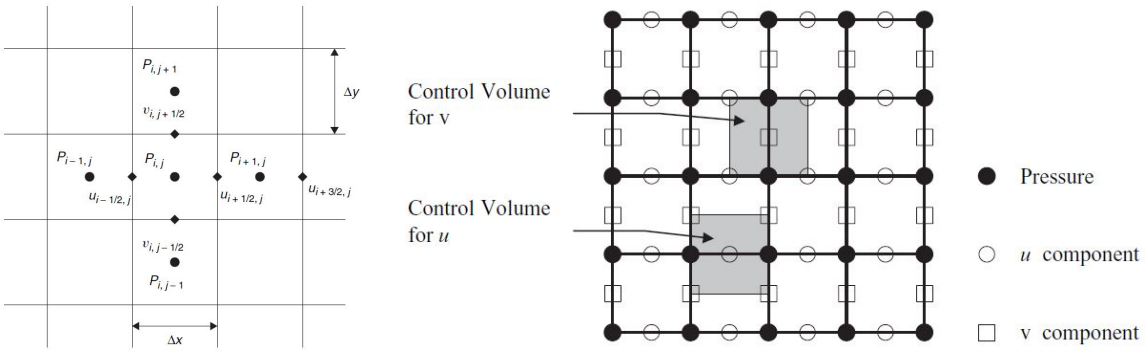
$$\frac{\boldsymbol{v}^{n+1} - \boldsymbol{v}^{n}}{\Delta t} + (\boldsymbol{v} \cdot \nabla \boldsymbol{v})^{n} = -\nabla p^{*} + \nu \left(\nabla \cdot \nabla \boldsymbol{v}\right)^{n}, \tag{15}$$

where $p^{*} = \gamma p^{n+1} + (1-\gamma) p^{n}$ with the weighting factor $0 < \gamma \leq 1$ controlling the degree of implicitness of the scheme. The system ((14),(15)) is now closed that allows uniquely determine $\boldsymbol{v}^{n+1}$ and p^{*} regardless of the value of γ. Varying γ produces small changes of order of Δt in the time history of the pressure field, but has no effect on the velocity field. The precise value of γ is immaterial, therefore, we may well adopt the simplest choice $\gamma = 1$ so that $p^{*} = p^{n+1}$, i.e. approximating the pressure gradient in a fully implicit manner. This choice constitutes the MAC scheme [10]. Equations ((14),(15)) are ordinarily solved by iterative methods, of which the traditional choices were successive over-relaxation (SOR).

`Include a section on Chorin's scheme [4] before discussing a general view in §4.` ToDo

3 Chorin's Projection Scheme

The proposed method by Chorin [4] was solved on a staggered grid as illustrated in figure 2(b) using Finite Differences. The algorithn can be summarized as follows: the time t is discretized, at every time step two separate equations are solved and combined based on the decomposition theorem of Ladyzhenskaya sometimes referred to as Helmholtz-Hodge Decomposition or simply as Hodge decomposition. The theorem states that the vector field $\mathbf{u}$ defined on a simply connected domain can be uniquely decomposed into a divergence-free (solenoidal) part $\mathbf{u}_{\text{sol}}$ and an irrotational part $\mathbf{u}_{\text{irrot}}$, leading to $\mathbf{u} = \mathbf{u}_{\text{sol}} + \mathbf{u}_{\text{irrot}}$. In the case of Navier-Stokes equations, advection-diffusion terms from initial



(a) MAC (b) general

Figure 2: Staggered grid.

to intermediate time are considered as a divergence-free part and the pressure gradient at the time step from intermediate to last as an irrotational part. The method developed by Chorin is focused on solving two equations (Burgers' and Poisson), which are obtained after introducing intermediate velocity $\tilde{\boldsymbol{v}}$.

Chorin's scheme was computed in single time step as follows:

1. Solve for intermediate velocity field $\tilde{\boldsymbol{v}}$ equation (16), generally not satisfying incompressible condition, since pressure gradient does not appear in

$$\frac{\tilde{\boldsymbol{v}} - \boldsymbol{v}^n}{\Delta t} = -\boldsymbol{v}^n \cdot \nabla_d \boldsymbol{v}^n + \nu \Delta_d \boldsymbol{v}^n, \tag{16}$$

   where superscript denotes known values of $\boldsymbol{v}$ at n'th time step and subscript d denotes discretized gradient and Laplacian operators. Chorin solved the above Burgers' equation (16) using the alternating direction implicit method.

2. This step is based on the application of Helmholz-Decomposition to final velocity field $\boldsymbol{v}^{n+1}$,

$$\boldsymbol{v}^{n+1} = \tilde{\boldsymbol{v}} - \nabla_d(\Delta t p^{n+1}). \tag{17}$$

   By imposing the divergence operator to the above equation (17) and making use of the continuity (1b) at time step $n+1$ a discretized version of the pressure-Poisson equation (10) is written as

$$\Delta_d p^{n+1} = \frac{1}{\Delta t} \nabla_d \tilde{\boldsymbol{v}}. \tag{18}$$

   Pressure condition at the boundary as in section 1.2 is

$$\boldsymbol{n} \cdot \nabla_d p^{n+1} = \boldsymbol{n} \cdot \left(-\frac{\boldsymbol{v}^{n+1} - \boldsymbol{v}^n}{\Delta t} - (\boldsymbol{v} \cdot \nabla_d \boldsymbol{v})^n + \nu \Delta_d \boldsymbol{v}^n \right). \tag{19}$$

   Equation (18) is solved with the corresponding boundary condition (19). Chorin used the Successive Point Over-Relaxation method, however, any appropriate iterative/direct/multigrid approaches might be used.

3. Use the solution for pressure from Step 2 and compute velocity at the next time step

$$\boldsymbol{v}^{n+1} = \tilde{\boldsymbol{v}} - \Delta t \nabla_d p^{n+1}. \tag{20}$$

4 Pressure-correction methods: general view

As discussed by Dukowicz & Dvinsky [6], both MAC [10] and Chorin's [4, 23, 24] methods can be put in a single form for time-discretization:

$$\text{Predictor}: \quad \frac{\widetilde{\boldsymbol{v}} - \boldsymbol{v}^n}{\Delta t} + \boldsymbol{v}^n \cdot \nabla \boldsymbol{v}^* = \nu \nabla \cdot \nabla \boldsymbol{v}^*, \tag{21a}$$

$$\text{Corrector}: \quad \begin{aligned} \frac{\boldsymbol{v}^{n+1} - \widetilde{\boldsymbol{v}}}{\Delta t} &= -\nabla p^*, \\ \nabla \cdot \boldsymbol{v}^{n+1} &= 0, \end{aligned} \tag{21b}$$

where Δt is the time step and $\boldsymbol{v}^*$ stands for $\boldsymbol{v}^n$ in MAC and the intermediate velocity $\widetilde{\boldsymbol{v}}$ in Chorin's method; the pressure time level is unspecified in MAC and is taken to be the new time level by Chorin. The general strategy in (21) is to decompose each time step into two substeps. On the first substep, the momentum equation is solved for the velocity components. The pressure gradient is either removed from the equation or approximated by an estimate. The obtained velocity field cannot be considered a solution at the new time level since it does not satisfy the incompressibility condition. The second substep is, therefore, needed, at which the correct pressure distribution is found and the correction of velocity is made.

If one adopts the *projection* point of view, as inspired by the existence theory of the NSEs, then the scheme (21) can be interpreted as a projection of the intermediate velocity field $\widetilde{\boldsymbol{v}}$ onto a subspace of vector fields with zero divergence. Namely, the term projection reflects the fact that we find a preliminary solution, which is not divergence-free, and then project it onto the space of divergence-free vector functions. This projection point of view, pioneered by Chorin [4] is based on the observation that equation (42a) can be put in the form $\frac{\partial \boldsymbol{v}}{\partial t} + \nabla p = -\boldsymbol{v} \cdot \nabla \boldsymbol{v} + \nu \nabla \cdot \nabla \boldsymbol{v}$, the left-hand side of which is a Hodge decomposition, so that by acting with the operator $\boldsymbol{P}$, which projects a vector field onto the space of divergence-free vector fields with appropriate boundary conditions, we get $\frac{\partial \boldsymbol{v}}{\partial t} = \boldsymbol{P}\left[-\boldsymbol{v} \cdot \nabla \boldsymbol{v} + \nu \nabla \cdot \nabla \boldsymbol{v}\right]$, which is the continuous version of (21a). On the other hand, the scheme (21) can be interpreted from the *operator-splitting* or *fractional time-step* (aka *time-splitting*) point of view on, which originates as a purely numerical method which is general and not specific to a particular set of equations [27], since by adding the two sets of equations to eliminate the intermediate velocity $\widetilde{\boldsymbol{v}}$. Regardless of which point of view to take in order to interpret (21), the conceptual and practical advantage of decoupling the "pressure correction" from the momentum equation calculations.

- ~~Add a section on spatial discretization using~~ §4.5 in [1] as well as research papers [2, 6, 14, 20], etc. Use the notations for discrete advection, Laplacian, divergence, and gradient operators as per equation (3) in Perot [20].
  The articles listed describe time discretization techniques, and spatial operators are kept arbitrary.
- Also discuss other numerical schemes in detail such as the ~~semi-implicit method for pressure-linked equations (SIMPLE)~~ [18, 19], ~~the pressure-implicit with splitting of operators (PISO)~~ [11, 12], Colonius' (discrete streamfunction), etc. ToDo
- Add a section on velocity-vorticity methods (§10.5.1 [28], §4.6 [1]).
- Since the scheme (21) is first-order accurate in time, discuss its improvements in terms of accuracy [2, 6, 14, 20].
- When a reference is made to specific schemes such as the Crank-Nicholson, Adams-Bashforth, and so on, for self-contained discussion their description should be included in an appendix.

5 Spatial Discretization

The initial system of equations (1) can be rewritten using discrete spatial operators as

$$\begin{aligned} \frac{\partial \boldsymbol{v}}{\partial t} + \mathbf{H}\left(\boldsymbol{v}\right) &= -Gp^{n+1} + \nu L\left(\boldsymbol{v}\right), \\ D\boldsymbol{v}^{n+1} &= 0, \end{aligned} \tag{22}$$

where $\mathbf{H}, G, L, D$ are discrete convective, gradient, Laplacian and divergence operators respectively.

Bringen and Chow [1] performed discretization on uniform staggered MAC grid illustrated on figure 2(a) and used the discrete operators at time step n:

$$\begin{aligned}
\mathbf{H}(\phi^n_{i+1/2,j}, \psi^n_{i,j+1/2}) &= \left(\frac{\left(\phi^n_{i,j}\right)^2 - \left(\phi^n_{i+1,j}\right)^2}{\Delta x} + \frac{(\phi\psi)^n_{i+1/2,j-1/2} - (\phi\psi)^n_{i+1/2,j+1/2}}{\Delta y} \right., \\
&\left. \frac{(\phi\psi)^n_{i-1/2,j+1/2} - (\phi\psi)^n_{i+1/2,j+1/2}}{\Delta x} + \frac{\left(\psi^n_{i,j}\right)^2 - \left(\psi^n_{i,j+1}\right)^2}{\Delta y} \right), \\
\left(G_x(\phi^n_{i+1/2,j}), G_y(\phi^n_{i,j+1/2})\right) &= \left(\frac{\phi^n_{i+1,j} - \phi^n_{i,j}}{\Delta x}, \frac{\phi^n_{i,j+1} - \phi^n_{i,j}}{\Delta y} \right), \\
L(\phi^n_{i+1/2,j}, \psi^n_{i,j+1/2}) &= \left(\frac{\phi^n_{i+3/2,j} - 2\phi^n_{i+1/2,j} + \phi^n_{i-1/2,j}}{\Delta x^2} + \frac{\phi^n_{i+1/2,j+1} - 2\phi^n_{i+1/2,j} + \phi^n_{i+1/2,j-1}}{\Delta y^2} \right., \\
&\left. \frac{\psi^n_{i+1,j+1/2} - 2\psi^n_{i,j+1/2} + \psi^n_{i-1,j+1/2}}{\Delta x^2} + \frac{\psi^n_{i,j+3/2} - 2\psi^n_{i,j+1/2} + \psi^n_{i,j-1/2}}{\Delta y^2} \right), \\
D(\phi^n_{i,j}, \psi^n_{i,j}) &= \left(\frac{\phi^n_{i+1/2,j} - \phi^n_{i-1/2,j}}{\Delta x} + \frac{\psi^n_{i,j+1/2} - \psi^n_{i,j-1/2}}{\Delta y} \right),
\end{aligned} \tag{23}$$

where ϕ, ψ are physical quantities. In the case of velocities they are evaluated at the cell surface leading to $1/2$ indices appearing, whereas for pressure the quantities are computed at the cell centre (marker). Convective operator $\mathbf{H}$ was evaluated in a conservative form

$$\mathbf{H}(\boldsymbol{v}) = \mathbf{H}(u, v) = \left(\frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y}, \frac{\partial uv}{\partial x} + \frac{\partial v^2}{\partial y} \right). \tag{24}$$

5.1 Magnetization formulation, not needed?

An alternative formulation of the incompressible NavierStokes equations (1) based on a new variable called "magnetization", impulse, or gauge can be made [2]. Two new variables, $\boldsymbol{m}$ and χ, are introduced that are related to the fluid velocity by

$$\boldsymbol{m} = \boldsymbol{v} + \nabla\chi.$$

The vector field $\boldsymbol{m}$ and the potential χ can be chosen to satisfy evolution equations in such a way that the fluid velocity and pressure derived from them satisfy the Navier-Stokes equations. Given $\boldsymbol{m}$, one possibility is to let $\boldsymbol{m}$ satisfy in Ω the evolution equation

$$\begin{aligned}
\boldsymbol{m}_t + (\boldsymbol{v} \cdot \nabla)\boldsymbol{v} &= \nu\nabla^2\boldsymbol{m} \\
\boldsymbol{v}|_{\partial\Omega} &= \boldsymbol{v}_b,
\end{aligned}$$

where $\boldsymbol{v} = \mathbf{P}(\boldsymbol{m})$ is the operator which projects a vector field onto the space of divergence-free vector fields with appropriate boundary conditions.

6 Modern schemes for solving Navier-Stokes equations

This section is focused on several popular methods described in two-dimensional cases.

6.1 SIMPLE - Semi-Implicit Method for Pressure Linked Equations

As discussed in section 1.1 there is a possibility that a solution on collocated grids will generate spurious oscillations due to interpolation, the use of staggered grids can address this issue. One of the methods on staggered grids was developed by Patankar and Spalding [19]. They named it SIMPLE, algorithm was based on a predictor-corrector procedure with successive pressure correction

$$p = \bar{p} + p', \tag{25}$$

where p is the actual pressure, $\bar{p}$ is the estimated pressure, and p' is the pressure correction. Similarly to the pressure correction (25) , the velocity vector is decomposed into two terms

$$\boldsymbol{v} = \bar{\boldsymbol{v}} + \boldsymbol{v}' = (\bar{u} + u', \bar{v} + v'). \tag{26}$$

Corrections of pressure are related to the velocity corrections by approximate momentum equation,

$$\frac{\partial \boldsymbol{v}'}{\partial t} = -Gp \implies \boldsymbol{v}' = -G(p\Delta t). \tag{27}$$

Combining velocity correctiion (26) with right hand side of expression (27) and substituting the result into the continuity equation (1b), we obtain the so-called pressure-correction Poisson equation

$$DGp' \equiv \nabla \cdot \nabla p' = -\frac{1}{\Delta t}\left(D\boldsymbol{v} - D\bar{\boldsymbol{v}}\right) = \frac{1}{\Delta t} D\bar{\boldsymbol{v}}, \tag{28}$$

where we set $\nabla \cdot \boldsymbol{v} = 0$ to enforce the mass conservation at the current time step. An iterative procedure used to obtain the solution is described below.

1. Guess the pressure $\bar{p}$ distribution at each cell centre.
2. Solve the momentum equation in control volumes formulation at the staggered grid points $(i+1/2, i-1/2, j+1/2, j-1/2)$ to find $\bar{\boldsymbol{v}}$.
3. Solve the pressure correction equation (28) to find p' at integer points $(i,j), (i,j-1), (i,j+1), (i-1,j), (i+1,j)$. The corner grid points are avoided, therefore, the scheme is called "semi-implicit".
4. Correct the pressure and velocity using the equations (25),(26) and (27):

$$\begin{aligned} p &= \bar{p} + p', \\ \boldsymbol{v} &= \bar{\boldsymbol{v}} - \Delta t \left[Gp' - \mathbf{H}(\boldsymbol{v}') + \nu L(\boldsymbol{v}')\right]. \end{aligned} \tag{29}$$

5. Replace the previous intermediate values of pressure and velocity $(\bar{p}, \bar{\boldsymbol{v}})$ with the corrected values $(p, \boldsymbol{v})$ and return to Step 2 .
6. Repeat Steps 2 to 5 until desired convergence is achieved.

There are several modifications of the current method that lead to better convergence. These algorithms were named SIMPLER (SIMPLE revised) and SIMPLEC (SIMPLE consistent), with SIMPLEC being the most effective scheme.

6.2 PISO - Pressure Implicit with Splitting of Operators

Issa [11,12] proposed an additional correction step to the SIMPLE algorithm. The sequential procedure is described below.

1. Set p^n, $\boldsymbol{v}^n$ from previous timestep.
2. Solve the momentum equation in control volumes formulation at the staggered grid points $(i+1/2, i-1/2, j+1/2, j-1/2)$ to find $\boldsymbol{v'}$.

$$\frac{1}{\Delta t}\left(\boldsymbol{v'} - \boldsymbol{v}^n\right) = -\mathbf{H}(\boldsymbol{v}) + \nu L(\boldsymbol{v}) - G\bar{p}. \tag{30}$$

3. Solve first pressure correction equation (28) to find p' at integer points $(i,j), (i,j-1), (i,j+1), (i-1,j), (i+1,j)$, the equation below has conservation of mass enforced $\nabla \cdot \boldsymbol{v}' = 0$.

$$DGp'' \equiv \nabla \cdot \nabla p' = -\frac{1}{\Delta t}\left(\nabla \cdot \boldsymbol{v}' - \nabla \cdot \boldsymbol{v}^n\right) = \frac{1}{\Delta t}\left(\nabla \cdot \boldsymbol{v}^n\right) = \frac{1}{\Delta t} D\boldsymbol{v}^n. \tag{31}$$

4. Apply first correction and obtain $\boldsymbol{v}''$ from

$$\frac{1}{\Delta t}\left(\boldsymbol{v}'' - \boldsymbol{v}^n\right) = D(-\mathbf{H}(\boldsymbol{v}') + \nu L(\boldsymbol{v}')) - G(p'). \tag{32}$$

5. Solve the second pressure correction

$$DGp'' \equiv \nabla \cdot \nabla p'' = \frac{1}{\Delta t}\, D\boldsymbol{v} + D\left(-\mathbf{H}(\boldsymbol{v}'') + \nu L(\boldsymbol{v}'')\right). \tag{33}$$

6. Apply second correction and obtain $\boldsymbol{v}''' = \boldsymbol{v}^{n+1}$ from

$$\frac{1}{\Delta t}\left(\boldsymbol{v}''' - \boldsymbol{v}^{n}\right) = -\mathbf{H}(\boldsymbol{v}'') + \nu L(\boldsymbol{v}'') \tag{34}$$

7 Discrete streamfunction method

- ~~Non-dimensionalization. Shall we keep u component under value of 1? can multiply the freestream and inlet by 0.5 if needed.~~ Normalized to max value of 1.
- ~~Scaling of G and D operators.~~ $\left(C^T\right)\left(\hat{M}\right)\left(\hat{G}\right)p = \left(C^T\right)\left(\hat{M}\right)\left(\hat{M}^{-1}G\right)p = C^TGp = (-DC)^Tp = 0$.
- ~~Symmetric BC on top to be included in all operators.~~
- ~~Indexing change for equations and figures.~~
- ~~Motivation.~~ Added some, where possible!
- BC discussion (analytic). [7, 15, 17, 21, 25, 26]. Finished [7, 15, 17, 21]. ToDo
- Outlet diffusion discretization is problematic. Requested PhD thesis of Jin and Persillon through UofA library. Wrote an email to their supervisor Prof. Braza and sent a message to Persillon.
- Substitute all refs/eqrefs with smart \cref and use one bracket for \cite.
- Time step n is in superscript, however, domain dim's are $n \times m$, dimension index is in subscript. Is it OK?
- Workaround for the outlet (headache division by zero) might be using uniform grid in the direction of the flow and refine grid only vertically.

7.1 Problem statement

Let the velocity vector $\boldsymbol{v}(t,x,y) = (u(t,x,y), v(t,x,y))$ and pressure $p(t,x,y)$ be the solutions to Eq. (1) on a semi-infinite domain together with boundary conditions:

$$\text{Momentum: } \frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v}\cdot\nabla\boldsymbol{v} = -\frac{1}{\rho}\nabla p + \frac{\mu}{\rho}\nabla\cdot\nabla\boldsymbol{v}, \tag{35a}$$

$$\rho \text{ - density}, \mu \text{ - dynamic viscosity.}$$

$$\text{Continuity: } \nabla\cdot\boldsymbol{v} = 0, \tag{35b}$$

$$0 \le x < \infty, 0 \le y < \infty, t \ge 0.$$

$$\text{Inlet and Freestream BC: } \boldsymbol{v}(t,0,y) = \boldsymbol{v}(t,x,\infty) = \frac{1}{2}\left(U_0 + A\cos\left(kx - \omega t + \phi_0\right), 0\right), \tag{35c}$$

$$\{|A| < U_0, k, \omega, \phi_0\} \subset \mathbb{R}.$$

$$\text{No-slip BC: } \boldsymbol{v}(t,x,0) = (0,0). \tag{35d}$$

$$\text{Initial condition: } \boldsymbol{v}(0,x,y) = (0,0). \tag{35e}$$

7.2 Artificial boundary conditions

Though there are methods that may solve problems on unbounded domains, the majority of the algorithms can only be applied to finite ones. The review papers on such BCs by Sani [25] and Tsynkov [26] list and compare different ABCs applied to the various flow scenarios. Below we will discuss the options which are most applicable to Eqs. (35).

In solutions of fluid flow, the reflection of waves at the outlet is common. Such behaviour is due to the elliptic nature of the incompressible Navier-Stokes. One of the earliest and most successful works (Engquist [7]) that addressed this problem was applied to the wave and shallow water equations. Although the method is not fully applicable to our equations, the authors systematically addressed and resolved the reflective behaviour of the outlet.

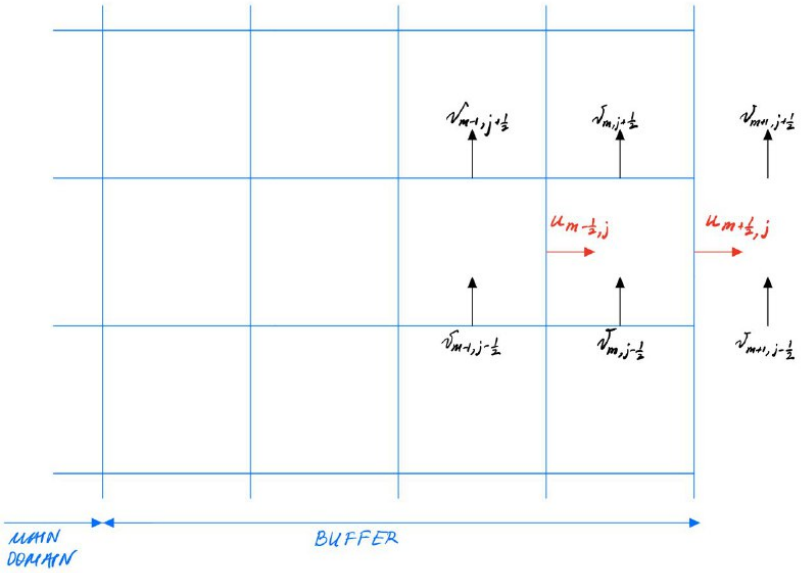


Figure 3: Buffer BC by Liu.

Liu [17] studied the spatial instability of planar Poiseuille flow. The governing equations in this paper were solved for perturbed velocity components with tilde $\tilde{u} = u - u_0, \tilde{v} = v - v_0$, where u_0, v_0 are mean values of velocity components u, v. The general idea is the introduction of an additional buffer domain (Fig. 3) at the exit that involves the use of:

1. Semi-implicit perturbed mass conservation

$$\tilde{u}^{n+1}_{m+\frac{1}{2},j} = \tilde{u}^{n}_{m-\frac{1}{2},j} - \frac{\tilde{v}^{n}_{m-1,j+\frac{1}{2}} - \tilde{v}^{n}_{m-1,j-\frac{1}{2}}}{\Delta y_j}\Delta x_m.$$

2. Zero change in acceleration of the tangential perturbed velocity component computed semi-implicitly as

$$\frac{\partial^2 \tilde{v}}{\partial x^2} = 0 \implies \tilde{v}^{n+1}_{m+1,j-\frac{1}{2}} = 2\tilde{v}^{n}_{m,j-\frac{1}{2}} - \tilde{v}^{n}_{m-1,j-\frac{1}{2}},$$

   where $\tilde{v}^{n+1}_{m+1,j-\frac{1}{2}}$ is the ghost component outside the domain.

This buffer was not consistent with the physics and required to be as short as possible.

Braza and collaborators [13, 15, 21] considered the shear flow (mixing layer) problem. They developed a set of BCs that allows the eddies to be freely developed downstream as they leave the domain. The most appropriate boundary conditions at top and bottom were those derived by considering boundaries as streamlines with:

$$v = 0, \frac{\partial u}{\partial y} = 0 \tag{36}$$

while the outlet had two types of boundary conditions:

$$\frac{\partial^2 u}{\partial x^2} = 0, \frac{\partial v}{\partial x} = 0, \tag{37a}$$

$$\frac{\partial \boldsymbol{v}}{\partial t} + u\frac{\partial \boldsymbol{v}}{\partial x} - \nu\frac{\partial^2 \boldsymbol{v}}{\partial y^2} = 0, \tag{37b}$$

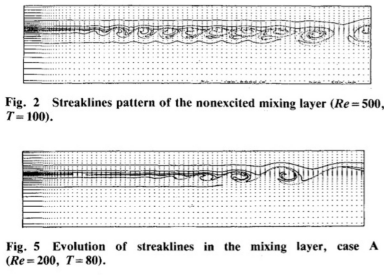


Fig. 2 Streaklines pattern of the nonexcited mixing layer ($Re = 500$, $T = 100$).

Fig. 5 Evolution of streaklines in the mixing layer, case A ($Re = 200$, $T = 80$).

(a) Kourta-Braza results.

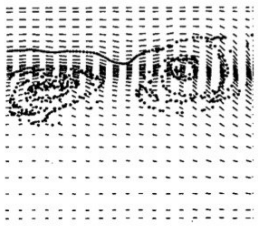


FIG. 10. Zoom of Fig. 10, outlet region, boundary conditions I.

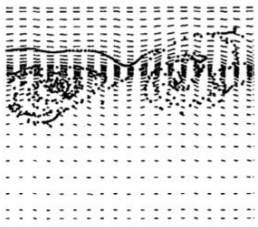


FIG. 11. Zoom of Fig. 11, outlet region, boundary conditions II.

(b) Jin-Braza results.

Figure 4: Resultant streaklines of Braza and collaborators.

where ν is kinematic viscosity. Results shown in Figs. 4(a) and 4(b) correspond to results obtained through Eqs. (37a) and (37b). The first Eq. (37a) made vortices hit against a "wall", while they went out of the domain. However, the second Eq. (37b) was derived similarly to Engquist [7] from wave equation and displayed non-reflective behaviour. In addition, the numerical algorithm used in the corresponding paper [13] required boundary conditions for auxiliary potential function ϕ at the outlet. The corresponding ABC for ϕ was derived from the numerical algorithm by taking into account Eq. (37b). (It might be possible to obtain an ABC for pressure depending on the algorithm used.)

7.2.1 Non-reflecting outlet BC

Velocity vector $\boldsymbol{v}$ is considered as the transported wave quantity incident on the boundary. Due to the elliptic nature of the flow, the boundary is said to satisfy the anisotropic wave equation

$$\frac{\partial^2 \boldsymbol{v}}{\partial t^2} - c_x^2 \frac{\partial^2 \boldsymbol{v}}{\partial x^2} - c_y^2 \frac{\partial^2 \boldsymbol{v}}{\partial y^2} = 0, \tag{38}$$

where c_x, c_y are the characteristic velocities of x, y wave propagation. We can rewrite Eq. (38) in terms of pseudo-differential operators

$$L\boldsymbol{v} \equiv -D_t^2 \boldsymbol{v} + c_x^2 D_x^2 \boldsymbol{v} + c_y^2 D_y^2 \boldsymbol{v} = \boldsymbol{0},$$

where D_x^2, D_y^2 and D_t^2 are second derivatives w.r.t. x, y and t variables respectively. Operator L can be factored into

$$L\boldsymbol{v} = L^+ L^- \boldsymbol{v} = 0,$$

where

$$L^+ \equiv c_x D_x + D_t \sqrt{1 - \left(\frac{c_y D_y}{D_t}\right)^2},$$

$$L^- \equiv c_x D_x - D_t \sqrt{1 - \left(\frac{c_y D_y}{D_t}\right)^2},$$

correspond to waves travelling inside and outside the domain in directions parallel to the x axis. To nullify the reflected waves, a total absorption [7] at the boundary

$$L^+ \boldsymbol{v} = 0 \tag{39}$$

is considered. It is possible to use the Taylor series to approximate the square root term, however, the results were found to be strongly ill posed, whereas, Pade second approximation

$$\sqrt{1 - \left(\frac{c_y D_y}{D_t}\right)^2} \approx 1 - \frac{1}{2}\left(\frac{c_y D_y}{D_t}\right)^2 \tag{40}$$

was found to be well posed [7, 16]. It is possible to rewrite Eq. (39) in terms of Eq. (40) as

$$\left(c_x D_x + D_t - \frac{c_y^2}{2D_t} D_y^2\right) \boldsymbol{v} = 0,$$

which then is matched to the momentum Navier-Stokes equation as

$$\frac{\partial \boldsymbol{v}}{\partial t} + u\frac{\partial \boldsymbol{v}}{\partial x} - \frac{1}{\mathrm{Re}}\frac{\partial^2 \boldsymbol{v}}{\partial y^2} = 0. \tag{41}$$

Taking into account successful results obtained in [13, 15, 21] and similarities between the boundary layer and mixing layer flows we will use Eqs. (36) and (37b) as boundary conditions for top and right boundaries and also truncate the domain to width and height L.

7.3 Nondimensionalization

In numerical computations, it is advantageous to maintain variables of comparable magnitude. This ensures that operations such as the multiplication of a large dimensional pressure variable with a small velocity are not performed. We will truncate the domain and normalize all equations by width L and stream velocity U_0. After introducing the following dimensionless variables (marked with prime $'$):

$$x \to Lx', \quad \boldsymbol{v} \to U_0 \boldsymbol{v}', \quad \nabla \to \frac{1}{L}\nabla', \quad p \to p'\rho U_0^2, \quad t \to \frac{L}{U_0}t',$$

we obtain

$$\frac{U_0}{\frac{L}{U_0}}\frac{\partial \boldsymbol{v}'}{\partial t'} + U_0 \boldsymbol{v} \cdot \left(\frac{1}{L}\nabla'\right) U_0 \boldsymbol{v} = -\frac{1}{\rho}\left(\frac{1}{L}\nabla'\right) p'\rho U_0^2 + \frac{\mu}{\rho}\left(\frac{1}{L}\nabla'\right) \cdot \left(\frac{1}{L}\nabla'\right) U_0 \boldsymbol{v}'.$$

Multiplying both sides of the equation by $\frac{L}{U_0^2}$ and introducing well-known Reynolds number

$$\mathrm{Re} = \frac{\rho L U_0}{\mu},$$

we obtain the non-dimensional momentum equation

$$\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v} \cdot \nabla \boldsymbol{v} = -\nabla p + \frac{1}{\mathrm{Re}} \nabla \cdot \nabla \boldsymbol{v},$$

where prime superscript $'$ is suppressed for the dimensionless variables. The continuity equation is nondimensionalized in a similar manner:

$$\left(\frac{1}{L}\nabla'\right) \cdot U_0 \boldsymbol{v}' = 0 \iff \nabla' \boldsymbol{v}' = 0 \implies \nabla \boldsymbol{v} = 0$$

where we drop prime superscripts in the last identity as well.

7.4 Problem statement on truncated domain

$$\text{Momentum: } \frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v} \cdot \nabla \boldsymbol{v} = -\nabla p + \nu \nabla \cdot \nabla \boldsymbol{v}, \tag{42a}$$
$$\nu = \frac{1}{\mathrm{Re}}, 0 \le x \le 1, 0 \le y \le 1, t \ge 0.$$
$$\text{Continuity: } \nabla \cdot \boldsymbol{v} = 0, \tag{42b}$$
$$0 \le x \le 1, 0 \le y \le 1, t \ge 0.$$
$$\text{Inlet and Freestream BC: } \boldsymbol{v}(t, 0, y) = \boldsymbol{v}(t, x, 1) = \frac{1}{2}\left(1 + A\cos\left(kx - \omega t + \phi_0\right), 0\right), \tag{42c}$$
$$\{A \le 1, k, \omega, \phi_0\} \subset \mathbb{R}.$$
$$\text{No-slip BC: } \boldsymbol{v}(t, x, 0) = (0, 0). \tag{42d}$$
$$\text{Outlet BC: } \frac{\partial \boldsymbol{v}}{\partial t} + u\frac{\partial \boldsymbol{v}}{\partial x} - \nu\frac{\partial^2 \boldsymbol{v}}{\partial y^2}, \tag{42e}$$
$$(x = 1).$$
$$\text{Initial condition: } \boldsymbol{v}(0, x, y) = (0, 0). \tag{42f}$$

7.5 Domain discretization

Following the successful nondimensionalization of the governing equations, we now turn our attention to the process of domain discretization. This crucial step involves dividing the computational domain into discrete elements, allowing us to compute variables and their derivatives on a finite set of points. Consider discretizing the area into m intervals horizontally and n intervals vertically, forming a grid of rectangular cells. To enhance memory access, it is recommended to organize the data using arrays instead of matrices. In this section, however, for the sake of simplicity, we use two indices i, j (column and row as in Harlow [10]) to represent coordinates on the grid.

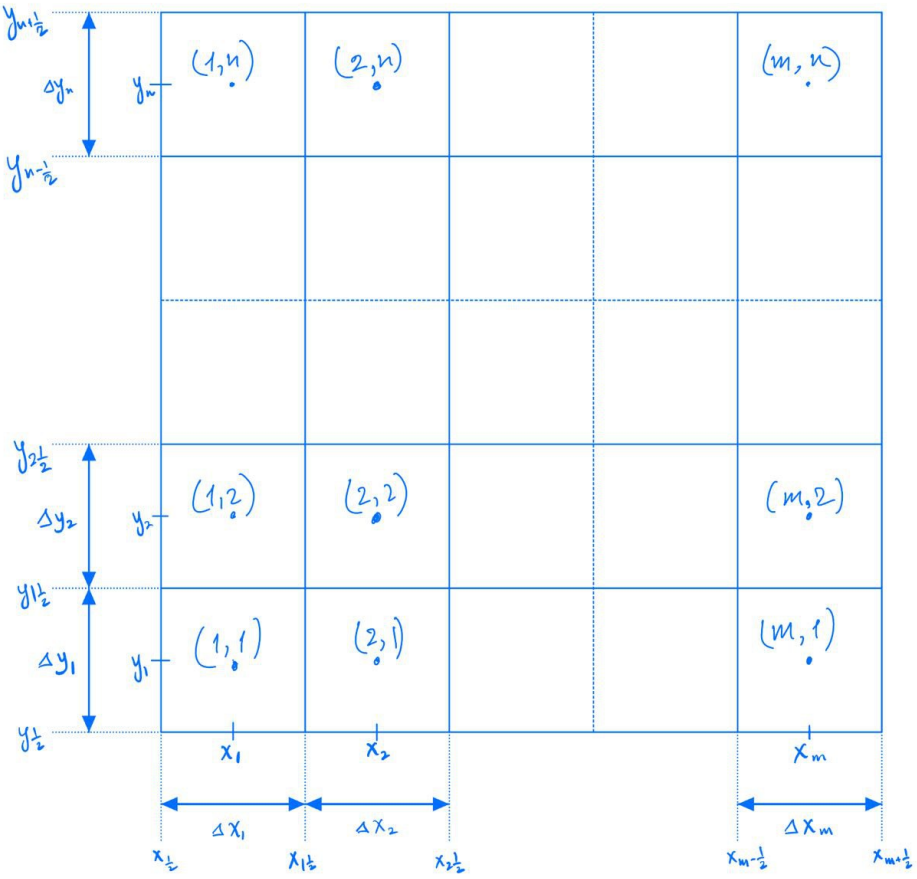


Figure 5: Domain discretization

In order to eliminate the pressure from our system a staggered grid arrangement is used. Details are discussed in Sections 7.6.3, 7.6.4 and 7.8. The u velocity components are stored at the centres of vertically oriented faces, while the values of v are stored at the centres of horizontal ones.

The number of unknown components inside the domain (excluding boundaries) is $m(n-1)$ for v and $(m-1)n$ for u. In Eq. (42e), the values of u for the outlet are not explicitly defined. To address this, we introduce additional n unknowns to the u array making the size of it $(m-1)n+n = mn$. These arrays for u and v are then concatenated into a single vector $\boldsymbol{v}$ of size $mn + (n-1)m$.

Grid indexation starts at $(x_{\frac{1}{2}}, y_{\frac{1}{2}}) = (0,0)$ and ends at $(x_{m+\frac{1}{2}}, y_{n+\frac{1}{2}}) = (1,1)$. The intervals $\Delta x, \Delta y$ have the same indexation as the corresponding cells they belong to. To increase the accuracy in certain domain parts we will refine the grid using the standard ratio rule $\Delta x_{i+1} = k_x \Delta x_i, \Delta y_{j+1} = k_y \Delta y_j$ with constants k_x, k_y close to the value of 1.

7.6 Discrete operators

This subsection focuses on the specific operators used in the discretization of Eqs. (42), providing the mathematical tools necessary to transform these continuous equations into a form that can be solved numerically.

Initial system of Eqs. (42) can be rewritten using discrete spatial operators as

$$\begin{bmatrix} \mathbf{I} & 0 \\ 0 & 0 \end{bmatrix} \frac{\partial}{\partial t} \begin{pmatrix} \boldsymbol{v} \\ p \end{pmatrix} = \begin{bmatrix} \hat{L} & -\hat{G} \\ -\hat{D} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{v} \\ p \end{bmatrix} + \begin{pmatrix} -\hat{\mathbf{H}}(\boldsymbol{v}) \\ 0 \end{pmatrix} + \mathrm{bc}_{\boldsymbol{v},p}, \tag{43}$$

where boundary conditions are in terms of pressure and velocity. In the following subsections, each of the discrete operators is described individually.

7.6.1 Transient terms

Attack Eq. (43) with the following schemes as in Colonius [5] (superscript denotes the time step):

Viscous - Implicit trapezoidal - Crank Nicholson scheme (second-order method in time).

$$\hat{L}\boldsymbol{v} = \frac{1}{2}\left(\hat{L}\boldsymbol{v}^{n+1} + \hat{L}\boldsymbol{v}^{n}\right) \tag{44}$$

Nonlinear - Explicit Adams-Bashforth (second-order method in time).

$$\hat{\mathbf{H}}(\boldsymbol{v}) = \frac{3}{2}\hat{\mathbf{H}}(\boldsymbol{v}^{n}) - \frac{1}{2}\hat{\mathbf{H}}(\boldsymbol{v})^{n-1}. \tag{45}$$

Pressure - Implicit Euler, though the pressure variable will be later eliminated in the algorithm (first-order method in time).

$$\hat{G}p = \hat{G}p^{n+1}. \tag{46}$$

The above schemes result in the following discretized system:

$$\begin{bmatrix} \frac{1}{\Delta t}\mathbf{I} - \frac{1}{2}\hat{L} & \hat{G} \\ \hat{D} & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{v}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \left[\frac{1}{\Delta t}\mathbf{I} - \frac{1}{2}\hat{L}\right]\boldsymbol{v}^{n} - \left[\frac{3}{2}\hat{\mathbf{H}}(\boldsymbol{v}^{n}) - \frac{1}{2}\hat{\mathbf{H}}(\boldsymbol{v}^{n-1})\right] \\ 0 \end{pmatrix} + \begin{pmatrix} \hat{bc}_1 \\ \hat{bc}_2 \end{pmatrix}. \tag{47}$$

7.6.2 Laplacian

As per Section 7.6.1 it is required to discretize implicit viscous term at time $n+1$. In order to obtain the discretized operator acting on a velocity vector we will rewrite Laplacian as a block matrix:

$$\hat{L} = \begin{bmatrix} \hat{L}^{u}_{xx} + \hat{L}^{u}_{yy} & 0 \\ 0 & \hat{L}^{v}_{xx} + \hat{L}^{v}_{yy} \end{bmatrix}. \tag{48}$$

Each pair of the matrices $\hat{L}^{u}_{xx}, \hat{L}^{v}_{xx}$ and $\hat{L}^{u}_{yy}, \hat{L}^{v}_{yy}$ will be equal on a uniform, but different on non-uniform grids. These matrices are computed in a similar way. Below we will discuss how the Laplacian matrix is constructed at different parts of the grid.

Inner part. The uniform grid refinement $\Delta x_{i+1} = k_x \Delta x_i, \Delta y_{j+1} = k_y \Delta y_j$ makes the order of the scheme consistent throughout the whole domain. Without loss of generality, let us show how to compute $\hat{L}^{u}_{xx}$, and the other three matrices are constructed in a similar manner. The power series of of $u_{i-\frac{1}{2}\pm 1,j}$ at nodes $x_{i-\frac{1}{2}\pm 1}$ with respect to $u_{i-\frac{1}{2},j}$ at node $x_{i-\frac{1}{2}}$ are

$$u_{i-\frac{1}{2}+1,j} = u_{i-\frac{1}{2},j} + \left.\frac{\partial u}{\partial x}\right|_{i-\frac{1}{2},j}\left(x_{i-\frac{1}{2}+1} - x_{i-\frac{1}{2}}\right) + \frac{1}{2}\left.\frac{\partial^2 u}{\partial x^2}\right|_{i-\frac{1}{2},j}\left(x_{i-\frac{1}{2}+1} - x_{i-\frac{1}{2}}\right)^2 + O\left(\Delta x^3\right), \tag{49}$$

$$u_{i-\frac{1}{2}-1,j} = u_{i-\frac{1}{2},j} + \left.\frac{\partial u}{\partial x}\right|_{i-\frac{1}{2},j}\left(x_{i-\frac{1}{2}-1} - x_{i-\frac{1}{2}}\right) + \frac{1}{2}\left.\frac{\partial^2 u}{\partial x^2}\right|_{i-\frac{1}{2},j}\left(x_{i-\frac{1}{2}-1} - x_{i-\frac{1}{2}}\right)^2 + O\left(\Delta x^3\right). \tag{50}$$

We can combine Eqs. (49) and (50) by cancelling the first derivative, which will result in

$$\left.\frac{\partial^2 u}{\partial x^2}\right|_{i-\frac{1}{2},j} \approx \frac{u_{i-\frac{1}{2}+1,j}\left(x_{i-\frac{1}{2}}-x_{i-\frac{1}{2}-1}\right)-u_{i-\frac{1}{2},j}\left(x_{i-\frac{1}{2}+1}-x_{i-\frac{1}{2}-1}\right)+u_{i-\frac{1}{2}-1,j}\left(x_{i-\frac{1}{2}+1}-x_{i-\frac{1}{2}}\right)}{\left(\frac{x_{i-\frac{1}{2}+1}-x_{i-\frac{1}{2}-1}}{2}\right)\left(x_{i-\frac{1}{2}}-x_{i-\frac{1}{2}-1}\right)\left(x_{i-\frac{1}{2}+1}-x_{i-\frac{1}{2}}\right)}+O(\Delta x)$$

$$\approx \frac{1}{h_c h_e}u_{i-\frac{1}{2}+1,j}+\frac{2}{h_w h_e}u_{i-\frac{1}{2}}+\frac{1}{h_c h_w}u_{i-\frac{1}{2}-1}+O(\Delta x). \tag{51}$$

where $h_w = x_{i-\frac{1}{2}}-x_{i-\frac{1}{2}-1}, h_c = \frac{x_{i-\frac{1}{2}+1}-x_{i-\frac{1}{2}-1}}{2}, h_e = x_{i-\frac{1}{2}+1}-x_{i-\frac{1}{2}}$ as in figure 6. The coefficients in front of the velocity components are then placed into the $\hat{L}^u_{xx}$ matrix. The order of this approximation becomes 2^{nd} for uniform grids.

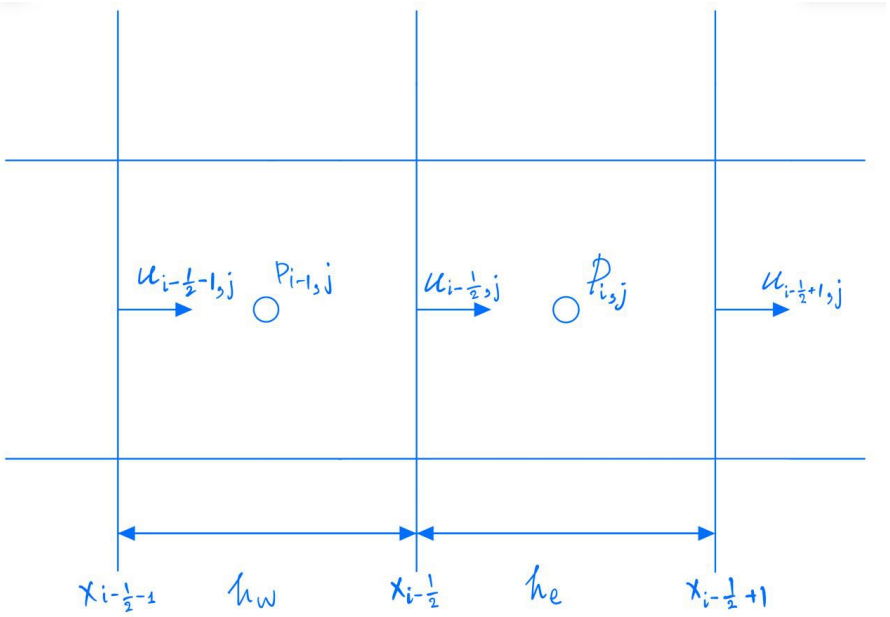


Figure 6: $\hat{L}^u_{xx}$ inner part.

Right boundary. **To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined. To be determined.**

Left boundary. The exact value on the left boundary ($u_{\frac{1}{2},j}$ in a box on Fig. 7) is given as a Dirichlet boundary condition, hence, it is possible to move the corresponding terms together with the coefficients to the right-hand side of the linear system and treat the values explicitly. The values are moved to the vector $\hat{bc}_1$ on the right-hand side of Eq. (47).

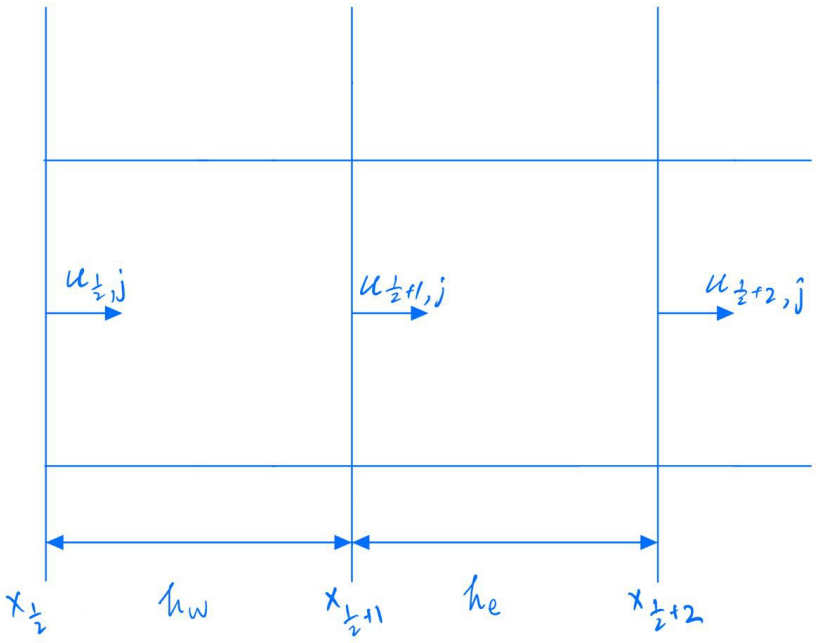


Figure 7: $\hat{L}^u_{xx}$ at the left boundary.

Top boundary. The boundary layer thickness can be made shorter than the truncated vertical domain length, thus, we can incorporate symmetric boundary condition over the top. This BC leads to $v_{\text{top}} = 0, u_{\text{top}} = u_{\text{freestream}}$, which affects only the $\hat{L}^u_{yy}$ and $\hat{L}^v_{yy}$ due to the 5-point stencil discretization. We will focus on $\hat{L}^u_{yy}$ in this part; $\hat{L}^v_{yy}$ is computed in a similar manner.

Consider the neighbouring cells near the wall as in figure 8. Resultant equation (51) can be written as

$$\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},n} = \frac{u_g\,(h_s) + u_{i-\frac{1}{2},n}\,(-2h_c) + u_{i-\frac{1}{2},n-1}\,(h_n)}{h_n h_s h_c}, \tag{52}$$

where $h_s = y_n - y_{n-1}, h_n = y_{n+1} - y_n, h_c = \frac{y_{n+1}-y_{n-1}}{2} = \frac{h_s+h_n}{2}$.

It is possible to make the order of the scheme $O(\Delta y^2)$ near the boundary as per equation (51) if u_g and $u_{i-\frac{1}{2},n-1}$ are set to be equidistant from $u_{i-\frac{1}{2},n}$.The value of u along the top face is known to be freestream velocity u_f, interpolation of $u_f = \frac{u_g + u_{i-\frac{1}{2},n}}{2} \implies u_g = 2u_f - u_{i-\frac{1}{2},n}$, which we can plug into Eq. (52) above

$$\begin{aligned}
\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},n} &= \frac{2u_f h_s}{h_n h_s h_c} + u_{i-\frac{1}{2},n}\left(\frac{-h_s}{h_n h_s h_c} + \frac{-2h_c}{h_n h_s h_c}\right) + u_{i-\frac{1}{2}-1,n}\frac{h_n}{u_c h_s h}, \\
\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},n} &= \frac{2u_f}{h_c h_c} + u_{i-\frac{1}{2},n}\left(\frac{-\,(2h_c + h_s)}{h_c h_s h_c}\right) + u_{i-\frac{1}{2}-1,n}\frac{1}{h_s h_c}, \\
\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},n} &= \frac{2u_f}{{h_c}^2} + u_{i-\frac{1}{2},n}\left(\frac{-\,(2h_c + h_s)}{h_c^2 h_s}\right) + u_{i-\frac{1}{2}-1,n}\frac{1}{h_s h_c}.
\end{aligned} \tag{53}$$

The first summand from the above equation is treated explicitly, i.e. moved to the vector $\hat{bc}_1$ on the right-hand side in Eq. (47), while the other two coefficients are used as elements for the $\hat{L}^u_{yy}$ matrix.

Bottom boundary. We use Eq. (51) and change the distances between velocities as in Fig. 9, which lead to

$$\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},1} = u_{i-\frac{1}{2},2}\left(\frac{1}{h_n h_c}\right) + u_{i-\frac{1}{2},1}\left(\frac{-2}{h_n h_s}\right) + u_g\left(\frac{1}{h_s h_c}\right). \tag{54}$$

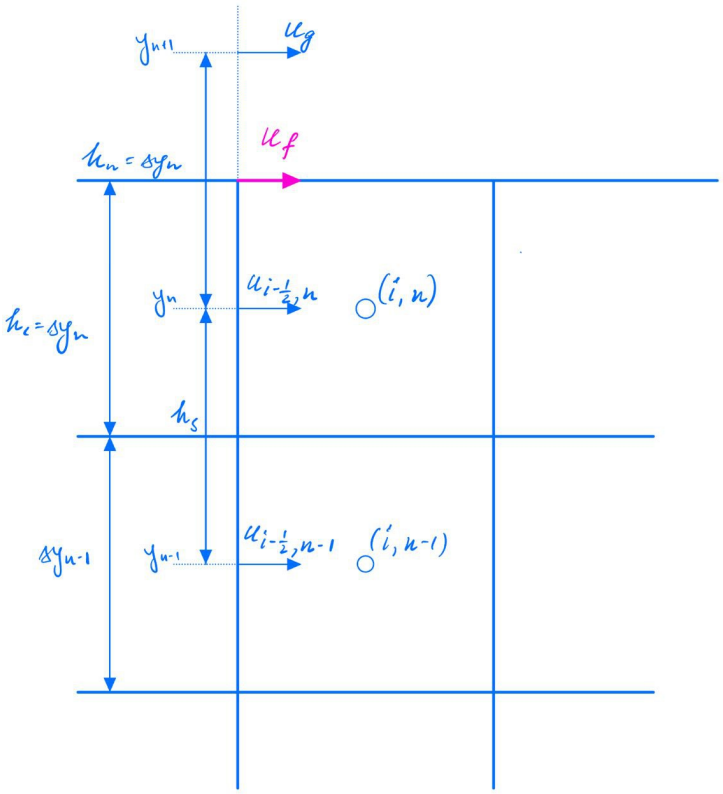


Figure 8: $\hat{L}^u_{yy}$ at top boundary.

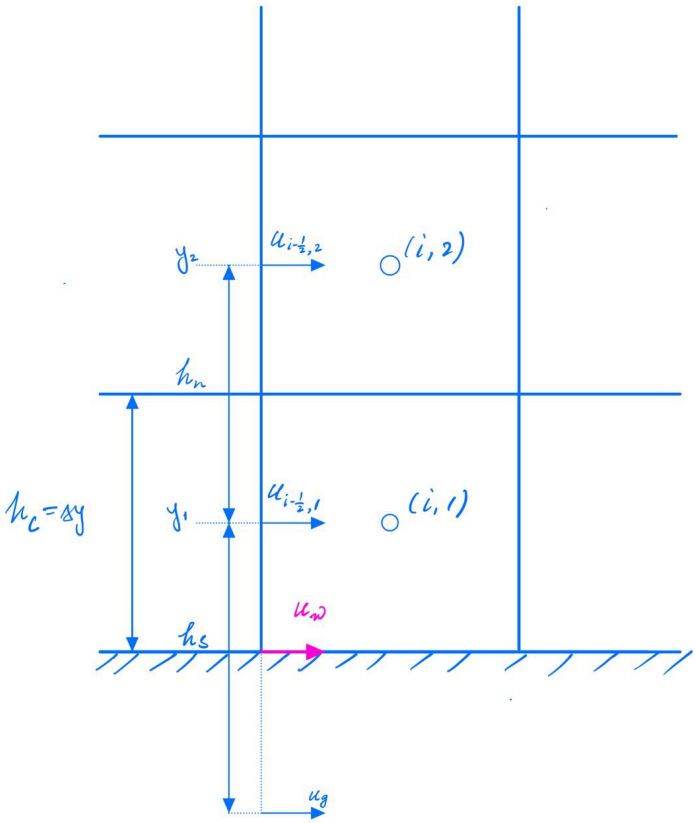


Figure 9: $\hat{L}^u_{yy}$ at the bottom boundary.

The position of u_g is not fixed, we can make the order of the scheme $O(\Delta y^2)$ if $h_s = h_n$. Interpolation of $u_w = \frac{u_{i-\frac{1}{2},1}+u_g}{2} = 0 \Rightarrow u_g = -2u_{i-\frac{1}{2},1}$, then

$$\begin{aligned}\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},1} &= u_{i-\frac{1}{2},2}\left(\frac{1}{h_n h_c}\right) + u_{i-\frac{1}{2},1}\left(\frac{-2}{h_n h_s} + \frac{-2}{h_s h_c}\right) \\ &= u_{i-\frac{1}{2},2}\left(\frac{1}{h_n h_c}\right) + u_{i-\frac{1}{2},1}\left(\frac{-2h_c - 2h_n}{h_s h_c h_n}\right).\end{aligned} \tag{55}$$

The coefficients are distributed into the $\hat{L}^u_{yy}$ matrix similarly to left and top domain cases. It is, in fact, possible to use the third velocity from the boundary instead of the ghost u_g outside. However, the order of such a scheme will be reduced due to the large ratio of the distances between the neighbouring velocities.

7.6.3 Divergence

One of the advantages of the staggered grid is that the discrete divergence D and gradient G operators are equal to the negative transpose of each other. Our goal is to eliminate the pressure from the system, Section 7.8 will explain the use of this property in detail.

The second line of Eq. (43) reads

$$\begin{aligned}\hat{D}\boldsymbol{v} &= \hat{bc}_2, \\ \begin{bmatrix} \hat{D}_x & \hat{D}_y \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix} &= \hat{bc}_2, \\ \frac{1}{\Delta x}D_x u + \frac{1}{\Delta y}D_y v &= \hat{bc}_2, \\ \frac{1}{\Delta_{xy}}\begin{bmatrix} D_x & D_y \end{bmatrix}\begin{bmatrix} u\Delta y \\ v\Delta x \end{bmatrix} = \frac{1}{\Delta_{xy}}Dq &= \hat{bc}_2,\end{aligned}$$

where vector q is referred to as velocity flux, and the matrix $D = [D_x D_y]$ without hat is the divergence matrix of integer coefficients. Matrix

$$\Delta_{xy} = \begin{bmatrix} \frac{1}{\Delta x_1 \Delta y_1} & 0 & \dots & 0 \\ 0 & \frac{1}{\Delta x_2 \Delta y_1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\Delta x_m \Delta y_n} \end{bmatrix} \tag{56}$$

is $nm \times nm$ diagonal matrix with entries corresponding to the inverse volumes of cells.

The second-order central difference scheme was employed in the construction of D, the order is consistent with the other spatial schemes. Consider a finite volume $V_{i,j}$, continuity equation for the cell centre is expressed as

$$\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right)_{i,j} = 0,$$

which can be discretized at cell centre i, j using a central difference scheme into

$$\frac{u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}}{\Delta x_i} + \frac{v_{i,j+\frac{1}{2}} - v_{i,j-\frac{1}{2}}}{\Delta y_j} = 0.$$

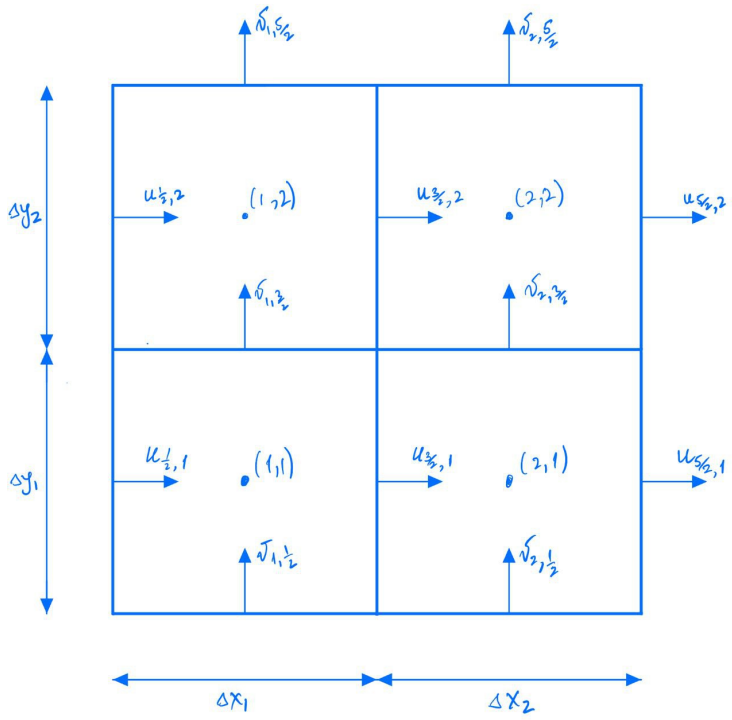


Figure 10: 2×2 grid example for divergence operator.

To provide a good visual example, consider 2×2 grid as in Fig. 10 with the same boundary conditions as in Eqs. (42). The partial derivatives evaluated at cell centres are then:

$$
\begin{aligned}
V_{1,1}: \frac{u_{\frac{3}{2},1} - u_{\frac{1}{2},1}}{\Delta x_1} + \frac{v_{1,\frac{3}{2}} - v_{1,\frac{1}{2}}}{\Delta y_1} &= 0, \\
V_{2,1}: \frac{u_{\frac{5}{2},1} - u_{\frac{3}{2},1}}{\Delta x_2} + \frac{v_{2,\frac{3}{2}} - v_{2,\frac{1}{2}}}{\Delta y_1} &= 0, \\
V_{1,2}: \frac{u_{\frac{3}{2},2} - u_{\frac{1}{2},2}}{\Delta x_1} + \frac{v_{1,\frac{5}{2}} - v_{1,\frac{3}{2}}}{\Delta y_2} &= 0, \\
V_{2,2}: \frac{u_{\frac{5}{2},2} - u_{\frac{3}{2},2}}{\Delta x_2} + \frac{v_{2,\frac{5}{2}} - v_{2,\frac{3}{2}}}{\Delta y_2} &= 0.
\end{aligned}
$$

After applying the boundary conditions (circled on the Fig. 10) to the above equations, we obtain the system

$$
\begin{bmatrix}
\frac{1}{\Delta x_1 \Delta y_1} & 0 & 0 & 0 \\
0 & \frac{1}{\Delta x_2 \Delta y_1} & 0 & 0 \\
0 & 0 & \frac{1}{\Delta x_1 \Delta y_2} & 0 \\
0 & 0 & 0 & \frac{1}{\Delta x_2 \Delta y_2}
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 & 0 & 1 & 0 \\
-1 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & -1 & 0 \\
0 & 0 & -1 & 1 & 0 & -1
\end{bmatrix}
\begin{bmatrix}
u_{\frac{3}{2},1}\Delta y_1 \\
u_{\frac{5}{2},1}\Delta y_1 \\
u_{\frac{3}{2},2}\Delta y_2 \\
u_{\frac{5}{2},2}\Delta y_2 \\
v_{1,\frac{3}{2}}\Delta x_1 \\
v_{2,\frac{3}{2}}\Delta x_2
\end{bmatrix}
=
\begin{bmatrix}
\frac{u_{\frac{1}{2},1}}{\Delta x_1} + \frac{v_{1,\frac{1}{2}}}{\Delta y_1} \\
\frac{v_{2,\frac{1}{2}}}{\Delta y_1} \\
\frac{u_{\frac{1}{2},2}}{\Delta x_1} - \frac{v_{1,\frac{5}{2}}}{\Delta y_2} \\
-\frac{v_{2,\frac{5}{2}}}{\Delta y_2}
\end{bmatrix},
\tag{57}
$$

which in general case is written as $\frac{1}{\Delta_{xy}} Dq = \hat{bc}_2$. The velocity values known at the boundaries are shifted to the right-hand side and explicitly treated. The divergence matrix D contains ± 1, as shown in Eq. (57).

7.6.4 Gradient

The pressure gradients are computed at the same coordinates as the unknown velocities according to Eq. (43). Below we will show that the discrete gradient and divergence operators satisfy

$$
G = -D^T \tag{58}
$$

on staggered/MAC grids. Moreover, the pressure gradients can be evaluated directly using central differences. This calculation does not require any interpolation on such grids, moreover, it is computationally cheaper and simple.

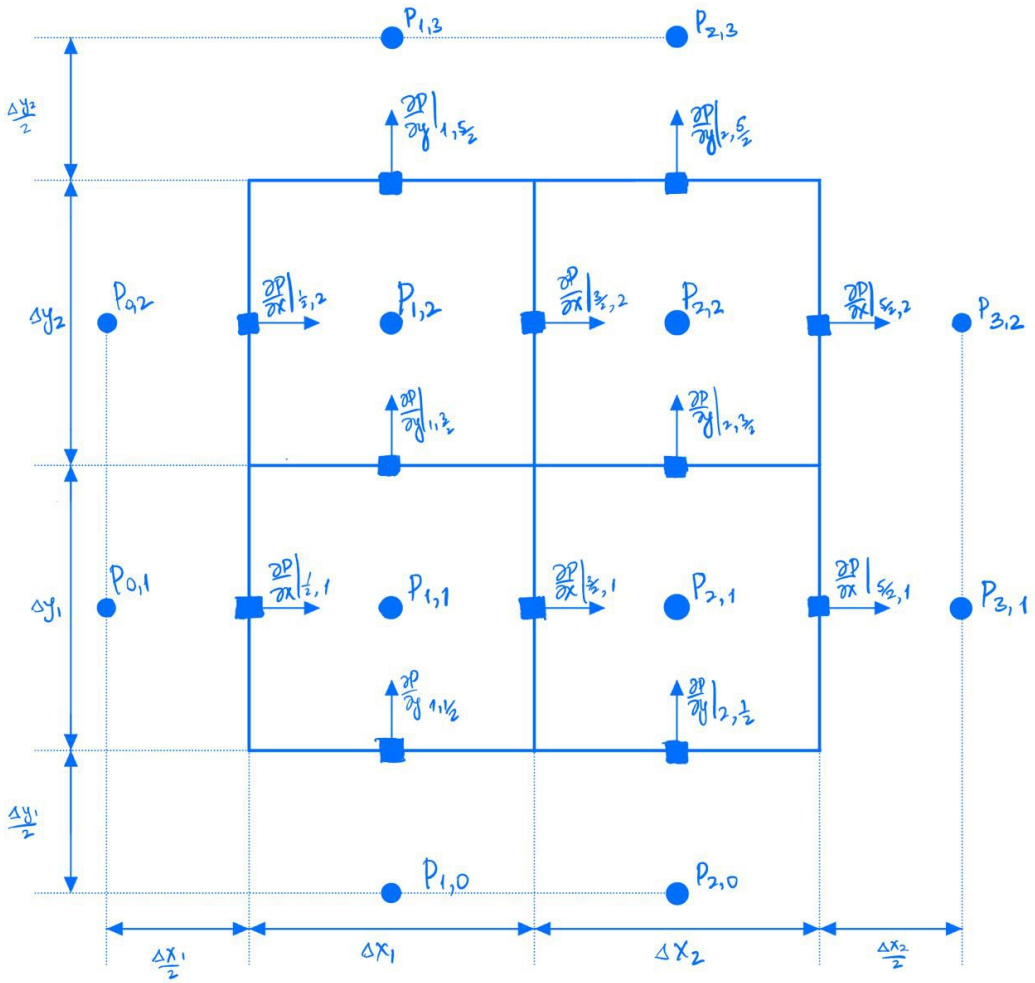


Figure 11: 2×2 grid example for gradient operator.

As an illustrative example, we may consider a 2×2 grid as in Fig. 11 with the same boundary conditions as in Eqs. (42). It is required to determine the pressure gradients across each unknown velocity (square nodes on Fig. 11):

$$
\begin{aligned}
\left.\frac{\partial P}{\partial x}\right|_{\frac{3}{2},1} &= \frac{2}{\Delta x_1 + \Delta x_2}\left(P_{1,2} - P_{1,1}\right), \\
\left.\frac{\partial P}{\partial x}\right|_{\frac{5}{2},1} &= \frac{1}{\Delta x_2}\left(P_{1,3} - P_{1,2}\right), \\
\left.\frac{\partial P}{\partial x}\right|_{\frac{3}{2},2} &= \frac{2}{\Delta x_1 + \Delta x_2}\left(P_{2,2} - P_{2,1}\right), \\
\left.\frac{\partial P}{\partial x}\right|_{\frac{5}{2},2} &= \frac{1}{\Delta x_2}\left(P_{2,3} - P_{2,2}\right), \\
\left.\frac{\partial P}{\partial y}\right|_{1,\frac{3}{2}} &= \frac{2}{\Delta y_1 + \Delta y_2}\left(P_{2,1} - P_{1,1}\right), \\
\left.\frac{\partial P}{\partial y}\right|_{2,\frac{3}{2}} &= \frac{2}{\Delta y_1 + \Delta y_2}\left(P_{2,2} - P_{1,2}\right).
\end{aligned}
$$

The next step is to rewrite the above expressions in matrix form using the pressure boundary conditions outside the grid. The resultant system of linear equations then becomes

$$
\begin{bmatrix}
\frac{2}{\Delta x_1+\Delta x_2} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{\Delta x_2} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{2}{\Delta x_1+\Delta x_2} & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{\Delta x_2} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{2}{\Delta y_1+\Delta y_2} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{2}{\Delta y_1+\Delta y_2}
\end{bmatrix}
\begin{bmatrix}
-1 & 1 & 0 & 0 \\
0 & -1 & 0 & 0 \\
0 & 0 & -1 & 1 \\
0 & 0 & 0 & -1 \\
-1 & 0 & 1 & 0 \\
0 & -1 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
P_{1,1} \\ P_{2,1} \\ P_{1,2} \\ P_{2,2}
\end{bmatrix}
=
\begin{bmatrix}
0 \\ \frac{1}{\Delta x_2}P_{3,1} \\ 0 \\ \frac{1}{\Delta x_2}P_{3,2} \\ 0 \\ 0
\end{bmatrix},
\tag{59}
$$

which can be rewritten as

$$
\nabla p = \hat{bc}_p \iff \hat{M}^{-1} G
\begin{bmatrix}
p_{1,1} \\ p_{1,2} \\ \vdots \\ p_{mn+(n-1)m}
\end{bmatrix}
= \hat{bc}_p,
\tag{60}
$$

where $\hat{M}^{-1}$ is the diagonal matrix containing the distances between neighbouring pressure coordinates, G is the gradient matrix and $[p_1, p_2, ...p_{mn}]^T$ is the vector of pressure values at the cell centres. It can be observed from Eqs. (57) and (59) that Eq. (58) holds and the general case is true by construction.

7.6.5 Advection

Let us rewrite advective derivatives in conservative form.

$$
\begin{aligned}
u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} &= u\frac{\partial u}{\partial x} + 0 + v\frac{\partial u}{\partial y} \\
&= u\frac{\partial u}{\partial x} + u\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) + v\frac{\partial u}{\partial y} \\
&= \left(u\frac{\partial u}{\partial x} + u\frac{\partial u}{\partial x}\right) + \left(u\frac{\partial v}{\partial y} + v\frac{\partial u}{\partial y}\right) \\
&= \frac{\partial uu}{\partial x} + \frac{\partial uv}{\partial y}.
\end{aligned}
\tag{61}
$$

Our goal is to compute advection components at the unknown velocity coordinates. Below we will describe the discretization schemes for Eq. (61).

Inner part. Advective component from x-momentum was expressed in conservative form by Eq. (61). It is discretized using the standard central differencing schemes [5]:

$$
\left(u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right)_{i-\frac{1}{2},j} = \frac{(uu)_{i,j} - (uu)_{i-1,j}}{\frac{\Delta x_i + \Delta x_{i-1}}{2}} + \frac{(uv)_{i-\frac{1}{2},j+\frac{1}{2}} - (uv)_{i-\frac{1}{2},j-\frac{1}{2}}}{\Delta y_i},
$$

which is evaluated at the same point as $u_{i-\frac{1}{2},j}$ (squares in Fig. 12). We need to compute uv at the nodes and uu at cell centres, which are triangles and circles respectively in Fig. 12. Linear interpolation [5] of these values is

$$
\begin{aligned}
(uu)_{i,j} &= \left(\frac{u_{i+\frac{1}{2},j} + u_{i-\frac{1}{2},j}}{2}\right)^2, \\
(uv)_{i-\frac{1}{2},j-\frac{1}{2}} &= \left(u_{i-\frac{1}{2},j-1} + \frac{\Delta y_{j-1}}{2}\frac{u_{i-\frac{1}{2},j} - u_{i-\frac{1}{2},j-1}}{\frac{\Delta y_{j-1}+\Delta y_j}{2}}\right)\left(v_{i-1,j-\frac{1}{2}} + \frac{\Delta x_{i-1}}{2}\frac{v_{i,j-\frac{1}{2}} - v_{i-1,j-\frac{1}{2}}}{\frac{\Delta x_{i-1}+\Delta x_i}{2}}\right) \\
&= \left(\frac{u_{i-\frac{1}{2},j}\Delta y_{j-1} + u_{i-\frac{1}{2},j-1}\Delta y_j}{\Delta y_{j-1} + \Delta y_j}\right)\left(\frac{v_{i,j-1\frac{1}{2}}\Delta x_{i-1} + v_{i-1,j-\frac{1}{2}}\Delta x_i}{\Delta x_{i-1} + \Delta x_i}\right).
\end{aligned}
$$

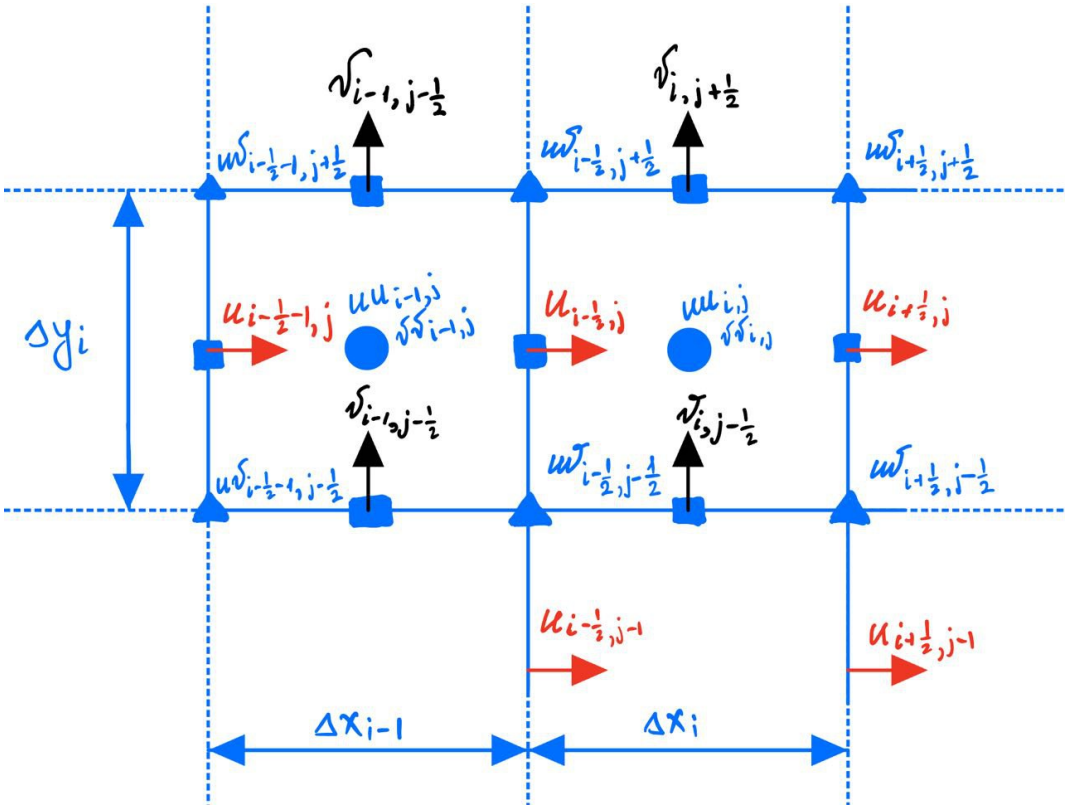


Figure 12: Advection discretization.

Right boundary (outlet).

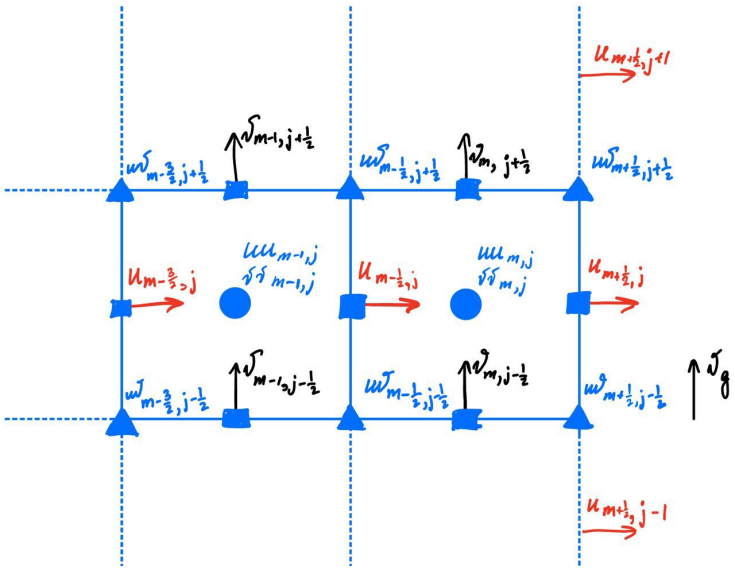


Figure 13: Right boundary advection.

One of the advantages of advective boundary condition $\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0$ (or $\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial y^2}$) is that it can be used to express $\frac{\partial uu}{\partial x}$ in terms of transient term as

$$\left(\frac{\partial uu}{\partial x}\right)_{m+\frac{1}{2},j} = 2\left(u\frac{\partial u}{\partial x}\right)_{m+\frac{1}{2},j} = 2\left(-\frac{\partial u}{\partial t}\right)_{m+\frac{1}{2},j}, \tag{62}$$

or in terms of transient and diffusive components

$$\left(\frac{\partial uu}{\partial x}\right)_{m+\frac{1}{2},j} = 2\left(u\frac{\partial u}{\partial x}\right)_{m+\frac{1}{2},j} = 2\left(-\frac{\partial u}{\partial t} + \nu\frac{\partial^2 u}{\partial y^2}\right)_{m+\frac{1}{2},j}. \tag{63}$$

The other advection summand

$$\left(\frac{\partial uv}{\partial y}\right)_{m+\frac{1}{2},j} = \frac{(uv)_{m+\frac{1}{2},j+\frac{1}{2}} - (uv)_{m+\frac{1}{2},j-\frac{1}{2}}}{\Delta} \tag{64}$$

requires values of v at the nodes $m+\frac{1}{2}, j\pm\frac{1}{2}$. We may use two neighbouring v components and extrapolate them linearly as

$$v_{m+\frac{1}{2},j+\frac{1}{2}} = v_{m-1,j\pm\frac{1}{2}} + 2\frac{v_{m,j\pm\frac{1}{2}} - v_{m-1,j\pm\frac{1}{2}}}{\Delta x_{m-1} + \Delta x_m}\left(\Delta x_m + \frac{1}{2}\Delta x_{m-1}\right). \tag{65}$$

Left boundary (inlet).

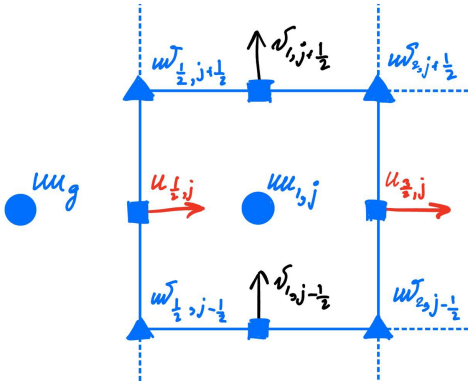


Figure 14: Left boundary advection.

The exact value of the uu_g element located outside the boundary (Fig. 14) is known from the inlet boundary conditions, whereas $uv_{\frac{1}{2},j}$ is zero for all j due to $v_{bc} = 0$ at the inlet.

Top boundary (symmetric).

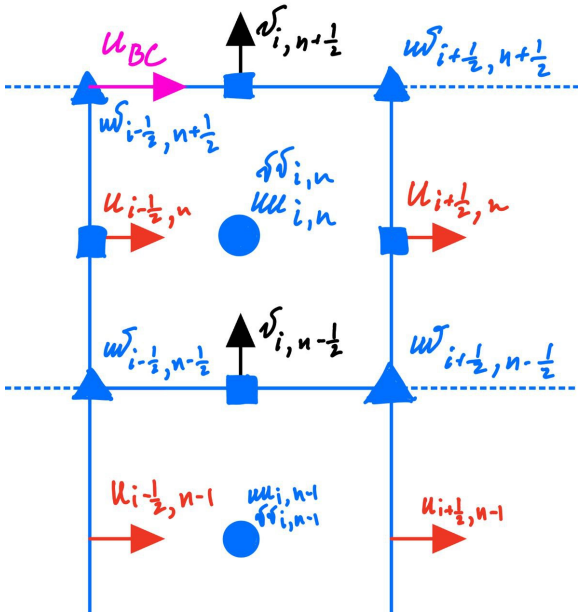


Figure 15: Top boundary advection.

The symmetric boundary condition defines a mirror surface, which has the same effect on the solution as the semi-infinite boundary condition if the boundary layer thickness is less than the truncated domain height. Mathematically such boundary condition can be written as $v = \frac{\partial v}{\partial y} = 0, \frac{\partial p}{\partial y} = 0$.

Since the exact value of the velocity u_{BC} and $v_{BC} = 0$ in a freestream are known, we can directly use them in our advection term computation making the second summand $uv = 0$, in particular

$$\left(\frac{\partial uu}{\partial x} + \frac{\partial uv}{\partial y}\right)_{i-\frac{1}{2},n} = 2\frac{(uu)_{i,n} - (uu)_{i-1,n}}{\Delta x_i + \Delta x_{i-1}} + 0. \tag{66}$$

Bottom boundary (no-slip).

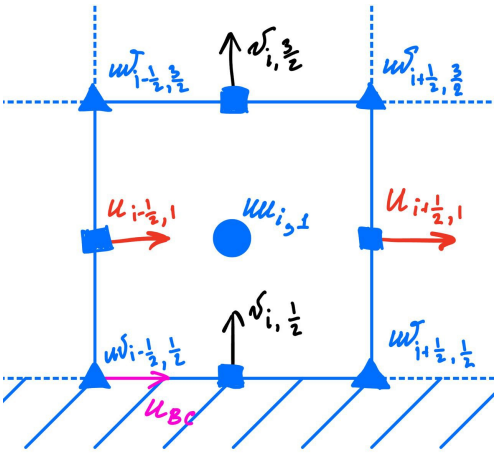


Figure 16: Bottom boundary advection

Computation of the advective terms just above the bottom boundary is the same as for the inner part, with the only exception being $uv_{i-\frac{1}{2},\frac{1}{2}} = 0$ on the boundary, due to the $u_{bc} = 0$ condition on the

plate (purple on Fig. 16).

The advective terms in the y-momentum equation are computed similarly. After the derivative values are evaluated, they are moved to the right-hand side as in Eq. (47) and treated explicitly.

7.6.6 Resulting system

After applying the above discrete operators, we can combine the summands into $\hat{A}$, $\hat{r}^n$; we then rewrite Eq. (47) as

$$\begin{bmatrix} \hat{A} & \hat{G} \\ \hat{D} & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{v}^{n+1} \\ p \end{pmatrix} = \begin{pmatrix} \hat{r}^n \\ 0 \end{pmatrix} + \begin{pmatrix} \hat{bc}_1 \\ \hat{bc}_2 \end{pmatrix}. \tag{67}$$

7.7 Symmetrization

The most effective methods for solving systems of linear equations often work optimally with symmetric matrices, making symmetry a desirable property. In order to achieve a symmetric system, we introduce two matrices: the diagonal scaling matrix $\hat{M}$ and the diagonal flux matrix R. These matrices are defined as

$$R \equiv \begin{bmatrix} \Delta y_j & 0 \\ 0 & \Delta x_i \end{bmatrix},$$
$$\hat{M} \equiv \begin{bmatrix} \frac{1}{2}(\Delta x_i + \Delta x_{i-1}) & 0 \\ 0 & \frac{1}{2}(\Delta y_j + \Delta y_{j-1}) \end{bmatrix}.$$

Using the R matrix we can express $q^{n+1} = R\boldsymbol{v}^{n+1} \implies \boldsymbol{v}^{n+1} = R^{-1}q^{n+1}$.

The only matrix which is asymmetric in (47) is a Laplacian on the non-uniform grid. In order to get a symmetric diffusion matrix we need to move to the new variables. As discussed in 7.5 the domain has m intervals in x direction and n intervals in y direction. The number of unknown velocities is $N_u = (m-1)n + n = mn$ and $N_v = m(n-1)$. The number of unknowns is equal to the number of face velocities in the interior of the domain plus the outlet boundary (extra n in N_u). We can rewrite $\boldsymbol{v}$ as a vector of unknowns

$$\boldsymbol{v} = [\underbrace{u_1; u_2; \dots; u_m; u_1; u_2; \dots; u_m; \dots; u_1; u_2; \dots; u_m}_{n \text{ blocks of } 1,\dots,m}, \underbrace{v_1; v_2; \dots; v_m; v_1; v_2; \dots; v_m; \dots; v_1; v_2; \dots; v_m}_{n-1 \text{ of each } 1,\dots,m \text{ block}}].$$

Define matrices

$$\Delta y_j = \mathrm{diag}([\underbrace{\Delta y_1; \Delta y_1; \dots; \Delta y_1}_{m \text{ times}}; \underbrace{\Delta y_2; \Delta y_2; \dots; \Delta y_2}_{m \text{ times}}; \dots; \underbrace{\Delta y_n; \Delta y_n; \dots; \Delta y_n}_{m \text{ times}}]),$$

$$\Delta x_i = \mathrm{diag}([\underbrace{\Delta x_1; \Delta x_2; \dots \Delta x_m; \Delta x_1; \Delta x_2; \dots \Delta x_m; \dots; \Delta x_1; \Delta x_2; \dots; \Delta x_m}_{n-1 \text{ blocks of } 1,\dots,m}]),$$

which are then combined into

$$R = \begin{bmatrix} \Delta y_j & 0 \\ 0 & \Delta x_i \end{bmatrix}.$$

Let the new vector variable of unknowns (mass flux) be defined as

$$q = R\boldsymbol{v},$$

where R is a diagonal matrix of size $mn \times m(n-1)$. We need to define the difference matrices to cancel out the $\frac{x_{i+1}x_{i-1}}{2}$ central term in diffusion discretization. Let the diagonal matrix

$$\Delta x_i + \Delta x_{i-1} = \mathrm{diag}[\underbrace{\Delta x_2 + \Delta x_1; \Delta x_3 + \Delta x_2; \dots \Delta x_{m+1} + \Delta x_m; \dots; \Delta x_2 + \Delta x_1; \Delta x_3 + \Delta x_2; \dots \Delta x_{m+1} + \Delta x_m}_{n \text{ times for each block of } \Delta x_2 + \Delta x_1 \text{ to } \Delta x_{m+1} + \Delta x_m}],$$

where the $x_{m+1} = x_m$ is a ghost interval lying outside the domain, and matrix

$$\Delta y_j + \Delta y_{j-1} = \mathrm{diag}[\underbrace{\Delta y_2 + \Delta y_1; \dots \Delta y_2 + \Delta y_1}_{m \text{ times}}; \underbrace{\Delta y_3 + \Delta y_2; \dots \Delta y_3 + \Delta y_2}_{m \text{ times}}; \underbrace{\Delta y_n + \Delta y_{n-1} \dots \Delta y_n + \Delta y_{n-1}}_{m \text{ times}}],$$

which are combined into

$$\hat{M} = \begin{bmatrix} \frac{1}{2}\left(\Delta x_i + \Delta x_{i-1}\right) & 0 \\ 0 & \frac{1}{2}\left(\Delta y_j + \Delta y_{j-1}\right) \end{bmatrix}.$$

The identity $q = R\boldsymbol{v}$ implies $\boldsymbol{v} = R^{-1}q$, therefore, the Laplacian matrix multiplied by the velocity vector is

$$L\boldsymbol{v} = LR^{-1}q,$$

premultiply by $\hat{M}$ to obtain

$$\hat{M}LR^{-1}q,$$

where $\hat{M}LR^{-1}$ is symmetric by construction.

Using the above transformation we can modify the initial system of linear equations

$$\begin{bmatrix} \hat{A} & \hat{G} \\ \hat{D} & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{v}^{n+1} \\ p \end{pmatrix} = \begin{pmatrix} \hat{r}^n \\ 0 \end{pmatrix} + \begin{pmatrix} \widehat{bc}_1 \\ \widehat{bc}_2 \end{pmatrix} \tag{68}$$

into

$$\begin{bmatrix} \hat{M}\hat{A}R^{-1} & \hat{M}\hat{G} \\ \hat{D}R^{-1} & 0 \end{bmatrix} \begin{pmatrix} q^{n+1} \\ p \end{pmatrix} = \begin{pmatrix} \hat{M}\hat{r}^n \\ 0 \end{pmatrix} + \begin{pmatrix} \hat{M}\widehat{bc}_1 \\ \widehat{bc}_2 \end{pmatrix}, \tag{69}$$

which can be rewritten as

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{pmatrix} q^{n+1} \\ p \end{pmatrix} = \begin{pmatrix} r^n \\ 0 \end{pmatrix} + \begin{pmatrix} bc_1 \\ bc_2 \end{pmatrix}, \tag{70}$$

where $A = \hat{M}\hat{A}R^{-1} = \frac{1}{\Delta t}\hat{M}R^{-1} - \frac{1}{2}\hat{M}\hat{L}R^{-1}$ is symmetric and $\hat{G} = \hat{M}^{-1}G$ according to equation (59) in momentum part. It is also possible to transform the divergence operator $\hat{D}$ with non-integer coefficients into D with integer coefficients in momentum part of system (70) using equation (57) and multiplying both sides of continuity equation by Δ_{xy} matrix (56):

$$\begin{aligned} \hat{D}\boldsymbol{v}^{n+1} &= \hat{bc}_2 \\ \left(\Delta_{xy}\right)\hat{D}\boldsymbol{v}^{n+1} &= \left(\Delta_{xy}\right)\hat{bc}_2 \\ \left(\Delta_{xy}\right)\frac{1}{\Delta_{xy}}DR\boldsymbol{v}^{n+1} &= \left(\Delta_{xy}\right)\hat{bc}_2 \\ \left(\Delta_{xy}\right)\frac{1}{\Delta_{xy}}Dq^{n+1} &= \left(\Delta_{xy}\right)\hat{bc}_2 \\ Dq^{n+1} &= bc_2. \end{aligned}$$

7.8 Nullspace method and pressure elimination

The goal of this subsection is to show how unknown pressure variables can be eliminated from sytem (70). Publications of Chang [3] and Hall [9] use the idea that in system (70) matrix D is wider than tall, hence it defines a nullspace. The nullspace is the set of all solutions to the homogeneous linear system $Dx = 0$, where x is a vector in the null space of D. Let C be the nullspace matrix containing such vectors x.

The number of rows in the nullspace C is equal to the number of faces with unknown velocities (N_f). In two dimensions C has N_n columns, which is equal to the number of nodes in the grid, whereas in three dimensions the nullspace has N_e columns being the number of edges.

In the two-dimensional case, the matrix C has two non-zero elements in each row, which are $+1$ and -1. The $+1$ value corresponds to the node $90°$ from the normal velocity vector, whereas -1 corresponds to the node $-90°$ from the normal velocity vector. For three dimensional case see Chang [3].

A more intuitive way of constructing the matrix C relies on the utilization of counterclockwise vorticity around the nodes within the domain and along the boundary if such nodes have adjacent velocities from $\boldsymbol{v}$. In the case of the open boundary vortices around the nodes belonging to the open segment are taken into consideration, see figure 17. If the direction of the velocity vector on the

adjacent face aligns with the vorticity's direction, +1 is assigned to the corresponding row; conversely, -1 is assigned in the case of opposite directions of velocity and vorticity. After applying the above procedure we obtain

$$C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \\ -1 & 0 \\ 1 & -1 \end{bmatrix}. \tag{71}$$

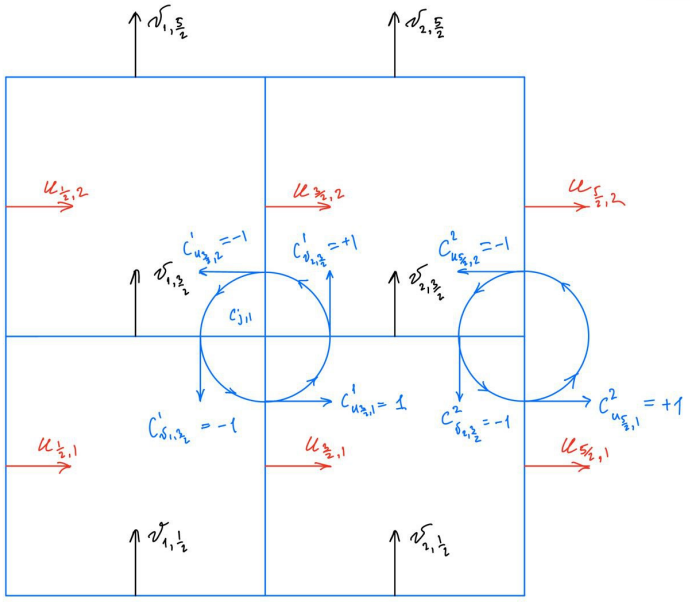


Figure 17: 2×2 example for C matrix.

The matrix C has dimensions corresponding to unknown velocities times the number of nodes around which these velocities revolve. Figure 18 illustrates matrices D and C for a 4×4 grid with an open boundary on the right side of the domain. Yellow squares represent +1 entries, while blue squares correspond to -1.

The desired product becomes $DC = 0$. $D = -G^T$ (subsections 7.6.3 and 7.6.4) leads to important property $(DC)^T = C^T D^T = -C^T G = 0$. Premultiplying momentum equation in (70) by C^T creates $C^T Gp = 0$ term, which eliminates the pressure from our system.

In order to make use of efficient solvers, the system $C^T A$ has to become symmetric, thus, we need to multiply it by C from the right. Hence, the final chord of this method is the introduction of another new variable called streamfunction $\psi : q_h = C\psi$, where q_h is a homogeneous solution to the problem (70) from subsection 7.7 and is discussed in the following subsection 7.9.

7.9 Resulting algorithm

Let us consider the solution to discretized system (70)

$$q^{n+1} = q_p^{n+1} + q_h^{n+1}, \tag{72}$$

where q_h^{n+1} is a solution to homogeneous continuity equation

$$Dq_h^{n+1} = 0, \tag{73}$$

and q_p^{n+1} is a particular solution to a non-homogeneous continuity equation

$$Dq_p^{n+1} = bc_2. \tag{74}$$

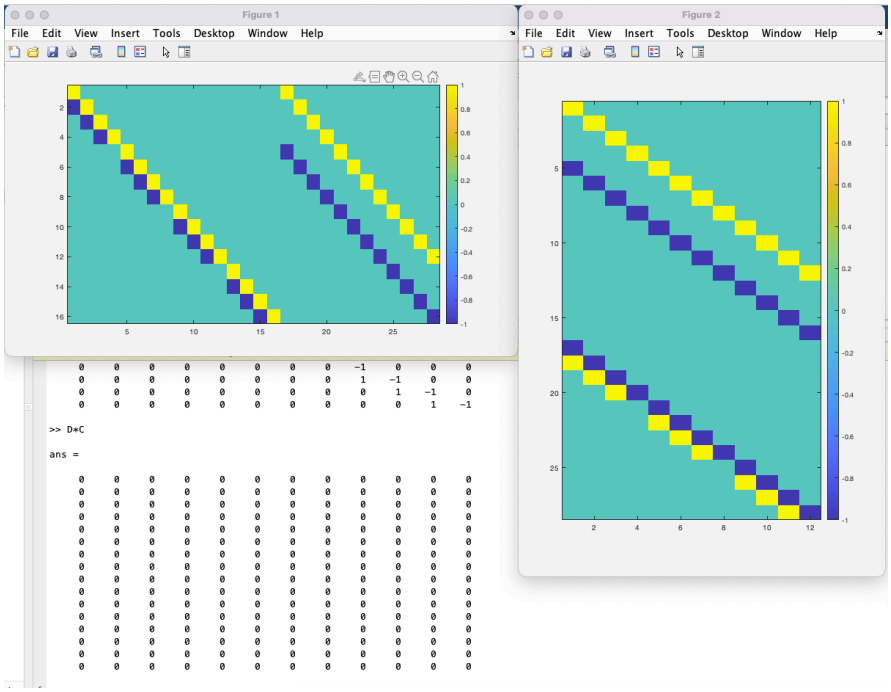


Figure 18: Divergence and Curl matrices.

The algorithm to solve the discrete Navier-Stokes system of equations (70) can be described as follows:

1. Construct matrix C, such that $DC = 0$ and $q_h = C\psi$.
2. Rewrite $q^{n+1} = q_p^{n+1} + q_h^{n+1}$.
3. Find q_p^{n+1} from the non-homogeneous continuity equation using the methods described in the next subsection 7.10.
4. Split the solution q^{n+1} into homogeneous and particular, then eliminate the pressure terms in the momentum equation.

$$\begin{aligned} Aq^{n+1} &= -Gp^{n+1} + S_1 + bc_1, && \text{where } S_1 \text{ is momentum source term,} \\ A(q_h^{n+1} + q_p^{n+1}) &= D^T p^{n+1} + S_1 + bc_1, && \text{premultiply by } C^T, \\ C^T A q_h^{n+1} &= C^T(S_1 + bc_1 - Aq_p^{n+1}), && \text{use } q_h^{n+1} = C\psi^{n+1}, \\ C^T AC\psi^{n+1} &= C^T(S_1 + bc_1 - Aq_p^{n+1}). \end{aligned}$$

5. Solve the resulting system from above for ψ^{n+1}.
6. Obtain $q_h^{n+1} = C\psi^{n+1}$.
7. Compute $q^{n+1} = q_p^{n+1} + q_h^{n+1}$.
8. Repeat the steps (3-7) since the boundary values in q^{n+1} were changed.

7.10 A particular solution to the discrete continuity equation

7.10.1 Particular solution using Lagrange multipliers

It is possible to find one particular solution to the discrete non-homogeneous continuity equation (74) using the method of Lagrange multipliers as follows

$$Dq_p^{n+1} = bc_2, \implies \mathcal{L}(q_p, \lambda) = \|q_p\|^2 + \lambda^T(bc_2 - Dq_p). \tag{75}$$

Differentiating w.r.t q_p and finding the minimum (derivative equal to zero) yields to

$$2q_p - D^T\lambda = 0. \tag{76}$$

We may premultiply by D to obtain

$$2Dq_p - DD^T\lambda = 0, \tag{77}$$

the substitution of $bc_2 = Dq_p$ will lead to

$$\begin{aligned} 2bc_2 &= DD^T\lambda, \\ \lambda &= 2\left(DD^T\right)^{-1} bc_2, \end{aligned} \tag{78}$$

plugging the above into (76) results in

$$q_p = D^T\left(DD^T\right)^{-1} bc_2 = D^\dagger bc_2, \tag{79}$$

where $D^\dagger = D^T\left(DD^T\right)$ is called pseudo-inverse. In the method above we minimize the square of the mass flux, which is equivalent to the minimization of kinetic energy.

8 Other interesting methods for solving Naver Stokes (may include algorithms from articles [2, 6, 14, 20])

Citing Colonius [5] The second type of error associated with vorticity advecting or diffusing through the boundary is typically handled by posing outflow boundary conditions. For incompressible flow, these are usually called convective boundary conditions, whereas for compressible flow the term non-reflecting boundary condition is often used. Multidomain boundary conditions used on 3 domains: fine, medium, and coarse.

9 Vorticity-Streamfunction formulation

Consider:

1. Streamfunction $\psi(x, y, t)$ of an incompressible two-dimensional flow:

$$u = \frac{\partial\psi}{\partial y}, \quad v = -\frac{\partial\psi}{\partial x}, \tag{80}$$

where $(u, v) = \boldsymbol{v}$.

The velocity vector at every point of space and every moment of time is tangential to the line $\psi = const$ and such lines represent the streamlines of the flow.

2. Vorticity $\omega = \nabla \times \boldsymbol{v}$, in two-dimensional case (x-y-plane) the only non-zero component of ω is z, which leads to

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}. \tag{81}$$

Importantly, continuity equation (1b) is satisfied immediately by taking spatial derivatives of streamfunction (80) components and adding them up. The governing equations (1) can now be transformed. The new set of equations becomes:

1. Transport equation for vorticity.

Application of $(\nabla\times)$ to momentum and taking into account continuity (1b) together with the fact

$$\frac{\partial}{\partial y}\left(\frac{\partial p}{\partial x}\right) - \frac{\partial}{\partial x}\left(\frac{\partial p}{\partial y}\right) = 0$$

results in

$$\boxed{\frac{\partial\omega}{\partial t} + u\frac{\partial\omega}{\partial x} + v\frac{\partial\omega}{\partial y} = \nu\left(\frac{\partial^2\omega}{\partial x^2} + \frac{\partial^2\omega}{\partial y^2}\right).} \tag{82}$$

2. Vorticity-Streamfunction equation.

   After substituting streamfunction definition (80) into vorticity (81) we obtain

$$\boxed{\nabla^2\psi = -\omega.} \tag{83}$$

These two equations above form a coupled system, and the pressure field (as promised at the end of Section 4) does not explicitly appear in neither equations (82) nor (83) and, in principle, is not needed in the solution.

The system of equations ((82),(83)) requires boundary conditions on ψ and ω. For the streamfunction, imposing physically plausible conditions is not difficult. One has to write the proper boundary conditions for the velocity components and use vorticity (81) to represent them as conditions for ψ and its derivatives. The situation is more difficult in the case of vorticity. There are no natural boundary conditions on ω, but they can be derived from the conditions on ψ by application of the equation (83) at the boundary. The process of finding boundary conditions of ω typically results in expressions containing second derivatives, therefore, special numerical treatment is required.

References

[1] S. Biringen and C.-Y. Chow. *An Introduction to Computational Fluid Mechanics by Example*. John Wiley & Sons, Inc., 2011.

[2] D. L. Brown, R. Cortez, and M. L. Minion. Accurate projection methods for the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 168:464–499, 2001.

[3] Wang Chang, Francis Giraldo, and Blair Perot. Analysis of an exact fractional step method. *Journal of Computational Physics*, 180(1):183–199, 2002.

[4] A. J. Chorin. Numerical solution of the Navier–Stokes equations. *Mathematics of Computation*, 22:745–762, 1968.

[5] Tim Colonius and Kunihiko Taira. A fast immersed boundary method using a nullspace approach and multi-domain far-field boundary conditions. *Computer Methods in Applied Mechanics and Engineering*, 197:2131–2146, April 2008.

[6] J. Dukowicz and A. Dvinsky. Approximate factorisation as a high order splitting for the incompressible flow equations. *Journal of Computational Physics*, 102:336–347, 1992.

[7] Björn Engquist and Andrew Majda. Absorbing boundary conditions for numerical simulation of waves. *Proceedings of the National Academy of Sciences*, 74(5):1765–1766, May 1977.

[8] Philip M. Gresho and Robert L. Sani. On pressure boundary conditions for the incompressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 7(10):1111–1145, 1987.

[9] C. Hall, T. Porsching, R. Dougall, R. Amit, A. Cha, C. Cullen, George Mesina, and Samir Moujaes. Numerical methods for thermally expandable two-phase flow-computational techniques for steam generator modeling. *NASA STI/Recon Technical Report N*, 1980.

[10] F. Harlow and E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids*, 8:2182–2189, 1965.

[11] R. I. Issa. Solution of the implicitly discretized fluid flow equations by operator splitting. 62:40–65, 1985.

[12] R. I. Issa, A. D. Gosman, and A. P. Watkins. The computation of compressible and incompressible recirculating flows by a non-iterative implicit scheme. 62:66–82, 1986.

[13] G. Jin and M. Braza. A Nonreflecting Outlet Boundary Condition for Incompressible Unsteady Navier-Stokes Calculations. *Journal of Computational Physics*, 107(2):239–253, August 1993.

[14] J. Kim and P. Moin. Application of a fractional-step method to incompressible Navier-Stokes equations. *Journal of Computational Physics*, 59:308–323, 1985.

[15] A. Kourta, M. Braza, P. Chassaing, and H. Haminh. Numerical analysis of a natural and excited two-dimensional mixing layer. *AIAA Journal*, 25(2):279–286, February 1987.

[16] Heinz-Otto Kreiss. Initial boundary value problems for hyperbolic systems. *Communications on Pure and Applied Mathematics*, 23(3):277–298, May 1970.

[17] C. Liu and Z. Liu. High Order Finite Difference and Multigrid Methods for Spatially Evolving Instability in a Planar Channel. *Journal of Computational Physics*, 106(1):92–100, May 1993.

[18] S. V. Patankar. *Numerical Heat Transfer and Fluid Flow*. Hemisphere, Washington, D.C., 1980.

[19] S. V. Patankar and D. B. Spalding. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal of Heat and Mass Transfer*, 15:1787–1806, 1972.

[20] J. B. Perot. An analysis of the fractional step method. *Journal of Computational Physics*, 108:249, 1993.

[21] Hélène Persillon and Marianna Braza. Physical analysis of the transition to turbulence in the wake of a circular cylinder by three-dimensional Navier–Stokes simulation. *Journal of Fluid Mechanics*, 365:23–88, June 1998.

[22] C. M. Rhie and W. L. Chow. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal*, 21(11):1525–1532, 1983.

[23] R.Témam. Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (I). *Archive for Rational Mechanics and Analysis*, 32:135–153, 1969.

[24] R.Témam. Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (II). *Archive for Rational Mechanics and Analysis*, 33:377–385, 1969.

[25] R. L. Sani and P. M. Gresho. Résumé and remarks on the open boundary condition minisymposium. *International Journal for Numerical Methods in Fluids*, 18(10):983–1008, 1994.

[26] Semyon V. Tsynkov. Numerical solution of problems on unbounded domains. A review. *Applied Numerical Mathematics*, 27(4):465–532, August 1998.

[27] N. N. Yanenko. *The Method of Fractional Steps*. Springer-Verlag, New York, 1971.

[28] O. Zikanov. *Essential Computational Fluid Dynamics*. Wiley, Hoboken, N.J., 2010.

A Appendix

A.1 Transient schemes

A.2 Laplacian symmetrization in depth

In order to get a symmetric diffusion matrix we need to move to the new variables.

Let us have $N_x = m$ intervals in x direction and $N_y = n$ intervals in y direction. The number of unknowns then will be $N_u = (N_x - 1)N_y + N_y = mn$ and $N_v = N_x(N_y - 1) = m(n-1)$. The number of unknowns is equal to the number of face velocities in the interior of the domain plus the outlet boundary (extra N_y in N_u). Rewrite $\boldsymbol{v}$ as a vector of unknowns

$$\boldsymbol{v} = [\underbrace{u_1; u_2; \dots; u_m; u_1; u_2; \dots; u_m; \dots; u_1; u_2; \dots; u_m}_{n \text{ times for each } 1 \dots m \text{ block}}, \underbrace{v_1; v_2; \dots; v_m; v_1; v_2; \dots; v_m; \dots; v_1; v_2; \dots; v_m}_{n-1 \text{ of each } 1,\dots,m \text{ block}}].$$

To maintain consistency - indexation goes left to right from bottom to top.

Define matrices

$$\Delta y_j = diag([\underbrace{\Delta y_1; \Delta y_1; \dots; \Delta y_1}_{m \text{ times}}; \underbrace{\Delta y_2; \Delta y_2; \dots; \Delta y_2}_{m \text{ times}}; \dots; \underbrace{\Delta y_n; \Delta y_n; \dots; \Delta y_n}_{m \text{ times}}]),$$
$$\Delta x_i = diag([\underbrace{\Delta x_1; \Delta x_2; \dots \Delta x_m; \Delta x_1; \Delta x_2; \dots \Delta x_m; \dots; \Delta x_1; \Delta x_2; \dots; \Delta x_m}_{n-1 \text{ times}}])$$

$$R = \begin{bmatrix} \Delta y_j & 0 \\ 0 & \Delta x_i \end{bmatrix}.$$

Let the new vector variable of unknowns (something like mass flux) be defined as

$$\boldsymbol{q} = R\boldsymbol{v},$$

where R is a diagonal matrix of size $mn \times m(n-1)$ defined above.

Next, we need to define the difference matrices to cancel out the $\frac{x_{i+1}x_{i-1}}{2}$ central term in diffusion discretization. Define the diagonal matrices

$$\Delta x_i + \Delta x_{i-1} = diag[\underbrace{\Delta x_2 + \Delta x_1; \Delta x_3 + \Delta x_2; \dots \Delta x_{m+1} + \Delta x_m; ...; \Delta x_2 + \Delta x_1; \Delta x_3 + \Delta x_2; \dots \Delta x_{m+1} + \Delta x_m}_{n \text{ times for each block of } \Delta x_2+\Delta x_1 \text{ to } \Delta x_{m+1}+\Delta x_m}],$$

where the $x_{m+1} = x_m$ is a ghost interval lying outside the domain.

$$\Delta y_j + \Delta y_{j-1} = diag[\underbrace{\Delta y_2 + \Delta y_1; \dots \Delta y_2 + \Delta y_1}_{m \text{ times}}; \underbrace{\Delta y_3 + \Delta y_2; \dots \Delta y_3 + \Delta y_2}_{m \text{ times}}; \underbrace{\Delta y_n + \Delta y_{n-1} \dots \Delta y_n + \Delta y_{n-1}}_{m \text{ times}}],$$

$$\hat{M} = \begin{bmatrix} \frac{1}{2}(\Delta x_i + \Delta x_{i-1}) & 0 \\ 0 & \frac{1}{2}(\Delta y_j + \Delta y_{j-1}) \end{bmatrix}.$$

The identity $\boldsymbol{q} = R\boldsymbol{v}$ implies $\boldsymbol{v} = R^{-1}\boldsymbol{q}$, which is used with Laplacian operator

$$L\boldsymbol{v} = LR^{-1}\boldsymbol{q},$$

premultiply by $\hat{M}$, the expression will become

$$\hat{M}LR^{-1}\boldsymbol{q},$$

where $\hat{M}LR^{-1}$ is symmetric by construction.