Discrete stream function method for the incompressible Navier-Stokes equations with oscillatory flow past a flat plate

Rauan

January 9, 2024

Abstract

The goal of these notes is to present the detailed overview of discrete stream function method for solving incompressible Navier-Stokes equations with oscillatory flow past a flat plate. We will discuss in detail the scheme formulation, transient and spatial discretizations. Special attention will be paid to the boundary conditions and their implementation. After studying these notes one must get a coherent picture of the application of discrete stream function method to incompressible flows and be able to implement the scheme in code.

Contents

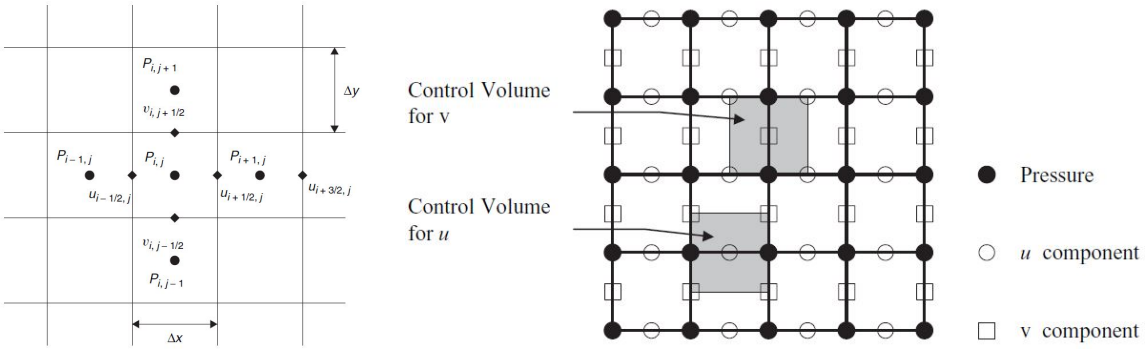
1 Introduction

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

$$\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v} \cdot \nabla \boldsymbol{v} = -\nabla p + \epsilon \nabla \cdot \nabla \boldsymbol{v} \tag{1.1a}$$

$$\nabla \cdot \boldsymbol{v} = 0, \tag{1.1b}$$

which are written here in the non-dimensional form, i.e. $\epsilon \equiv Re^{-1}$ 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 Fig. 1(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 [14]) and hence staggered grids are used since the work of Harlow and Welch [6] introducing the marker-and-cell (MAC) method, cf. Fig. 1(a).



(a) MAC (b) general

Figure 1: Staggered grid.

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 Fig. 1(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 Eqs. (1.1) 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 Eqs. (1.1) is that pressure p is not determined by a time-evolution equation, but rather is implicitly defined by the incompressibility Eq. (1.1b), 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 Eq. (1.1a) and combining the result with Eq. (1.1b). 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 Eq. (1.1b) and its solution can in principle be substituted back into Eq. (1.1a) to obtain an evolution equation for the velocity field alone. Alternatively, the pressure can be immediately eliminated by taking the curl of momentum Eq. (1.1a), 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 Eqs. (1.1) 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.

- ~~Motivation.~~ Added some, where possible!
- ~~BC discussion (analytic).~~ [4, 8, 10, 11, 12, 13].
- ~~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.~~ Waiting for the response form UofA library since 11th Dec.
- ~~Found this condition $\frac{\partial \boldsymbol{v}}{\partial t} + u\frac{\partial \boldsymbol{v}}{\partial x} + v\frac{\partial \boldsymbol{v}}{\partial y} - \epsilon\frac{\partial^2 \boldsymbol{v}}{\partial y^2} = 0$ by G. Fournier and F. Golanski (2008, but not many citations). They claim Prandtl BL solution for advective bc $\frac{\partial \boldsymbol{v}}{\partial t} + u\frac{\partial \boldsymbol{v}}{\partial x} = 0$ is wrong, while their solution matches Prandtl BL. Looks like we can express the remaining diffusive summand in terms of pressure gradient at the boundary.~~ Wrote an email to author, waiting for the response.
- ~~Change computational domain to inner part only? Boundary values are to be determined by BC, not the momentum eq'n, contrary to Hall~~ [5]. Changed unknowns to inner part only. Values at the boundary are determined using BC.

ToDo

- ~~Check algebra of diffusion and advection for v component.~~
- It feels like BC scheme which is consistent with inner discretization won't work, we will use Jin-Braza-Braza scheme.
- Outlet BC stated early in non-dim truncated part, but discussion is much later.
- Continuous analogue of C^T to be determined. Figured out that $C^T C = L$ and $C^T L C = L^2$. Some information of transpose of curl is in Vector Calculus notes on page 24 of course by TSOGTGEREL GANTUMUR, but the material is quite heavy.
- Repeating Fig. 7 from first try of Laplacian and Fig. 16 after Artificial BC discussion.
- Paraphrase [1] in Section 12 since most of it is copypaste.
- Two identities above Eq. (10.5) are missing algebra.

2 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 Eqs. (1.1) on a semi-infinite domain together with boundary conditions:

$$\text{Momentum: } \frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v}\cdot\nabla\boldsymbol{v} = -\frac{1}{\rho}\nabla p + \frac{\mu}{\rho}\nabla\cdot\nabla\boldsymbol{v}, \tag{2.1a}$$

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

$$\text{Continuity: } \nabla\cdot\boldsymbol{v} = 0, \tag{2.1b}$$

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

$$\text{Inlet and Freestream BC: } \boldsymbol{v}(t,0,y) = \boldsymbol{v}(t,x,\infty) = \left(U_0 + A\cos\left(kx - \omega t + \phi_0\right), 0\right), \tag{2.1c}$$

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

$$\text{No-slip BC: } \boldsymbol{v}(t,x,0) = (0,0). \tag{2.1d}$$

$$\text{Initial condition: } \boldsymbol{v}(0,x,y) = (0,0) \text{ or Blasius profile.} \tag{2.1e}$$

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

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

After 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:

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

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

4 Problem statement on truncated domain

The initial Eqs. (2.1) from Section 2 on semi-infinite domain now become a system of partial differential equations on a unit square domain

$$\text{Momentum: } \frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v} \cdot \nabla \boldsymbol{v} = -\nabla p + \epsilon \nabla \cdot \nabla \boldsymbol{v}, \tag{4.1a}$$

$$\epsilon = \frac{1}{\mathrm{Re}}, 0 \le x \le 1, 0 \le y \le 1, t \ge 0.$$

$$\text{Continuity: } \nabla \cdot \boldsymbol{v} = 0, \tag{4.1b}$$

$$0 \le x \le 1, 0 \le y \le 1, t \ge 0.$$

$$\text{Inlet and Freestream BC: } \boldsymbol{v}(t, 0, y) = \boldsymbol{v}(t, x, 1) = (1 + A\cos(kx - \omega t + \phi_0), 0), \tag{4.1c}$$

$$\{A \le 1, k, \omega, \phi_0\} \subset \mathbb{R}.$$

$$\text{No-slip BC: } \boldsymbol{v}(t, x, 0) = (0, 0). \tag{4.1d}$$

$$\text{Outlet BC: } \frac{\partial \boldsymbol{v}}{\partial t} + u\frac{\partial \boldsymbol{v}}{\partial x} - \epsilon\frac{\partial^2 \boldsymbol{v}}{\partial y^2} = 0, \text{ discussed in Section 7,} \tag{4.1e}$$

$$(x = 1).$$

$$\text{Initial condition: } \boldsymbol{v}(0, x, y) = (0, 0) \text{ or Blasius profile.} \tag{4.1f}$$

5 Domain discretization

Following the 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 called grid. Consider discretizing the area into M intervals horizontally and N intervals vertically forming a mesh of rectangular cells. To enhance memory access it is recommended to organize the data using vectors instead of matrices. The reason behind is memory access time; a single index i works faster than double i, j in most programming languages. In this section, however, for the sake of simplicity, we use two indices i, j (column and row as in Harlow and Welch [6]) to represent coordinates on the grid.

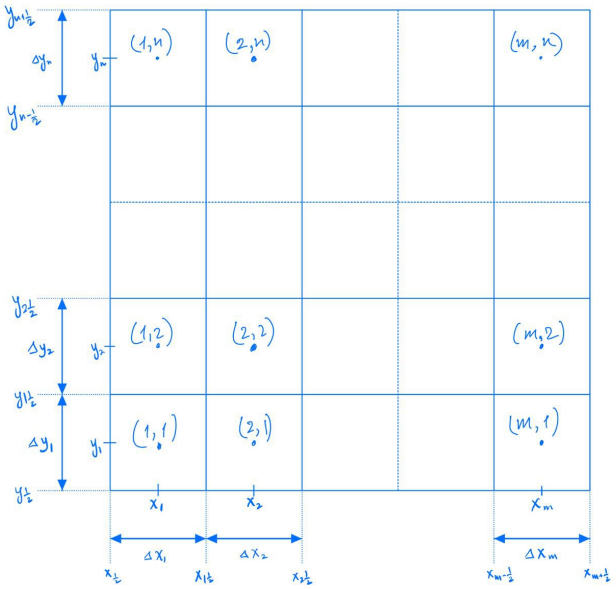


Figure 2: Domain discretization

To eliminate the necessity of solving an additional equation for pressure we will be using a staggered grid arrangement. Details are discussed in Sections 6.3, 6.4 and 11. Moreover, the pressure gradients can be evaluated directly using central differences on such grids. This calculation on staggered grids does not require any interpolation, furthermore, it is computationally cheaper and simpler to implement.

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 BC (4.1e), the values of u for the outlet are not explicitly defined. To address this, we will express the values on the boundary using BC (4.1e) in terms of data from inner part of the domain. These arrays for u and v are then concatenated into a single vector $\boldsymbol{v}$ of size $M(N-1)+(M-1)N$.

Increasing the accuracy closer to the wall and the outlet does look advantageous, hence, 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. 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.

6 Discrete operators

This subsection focuses on the specific operators used in the discretization of Eqs. (4.1), providing the mathematical tools necessary to transform these continuous equations into a form that can be solved numerically.

Denote discrete spatial operators as:

$\hat{L}$: Laplacian.

$\hat{G}$: Gradient.

$\hat{D}$: Divergence.

$\hat{\mathbf{H}}$: Non-linear advective terms.

Then initial system of Eqs. (4.1) can be approximated using such operators as

$$\begin{bmatrix} \mathbf{I} & 0 \\ 0 & 0 \end{bmatrix} \frac{\partial}{\partial t} \begin{pmatrix} \boldsymbol{v} \\ p \end{pmatrix} = \begin{bmatrix} \hat{L} & -\hat{G} \\ -\hat{D} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{v} \\ p \end{bmatrix} + \begin{pmatrix} -\hat{\mathbf{H}}(\boldsymbol{v}) \\ 0 \end{pmatrix} + \mathrm{bc}_{\boldsymbol{v},p}, \tag{6.1}$$

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

Attack Sys. (6.1) with the following schemes as in Colonius [3] (superscript denotes the time step):

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

$$\hat{L}\boldsymbol{v} = \frac{1}{2}\left(\hat{L}\boldsymbol{v}^{n+1} + \hat{L}\boldsymbol{v}^{n}\right) \tag{6.2}$$

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

$$\hat{\mathbf{H}}(\boldsymbol{v}) = \frac{3}{2}\hat{\mathbf{H}}(\boldsymbol{v}^{n}) - \frac{1}{2}\hat{\mathbf{H}}(\boldsymbol{v})^{n-1}. \tag{6.3}$$

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

$$\hat{G}p = \hat{G}p^{n+1}. \tag{6.4}$$

The above schemes result in the following time-discretized system:

$$\begin{bmatrix} \frac{1}{\Delta t}\mathbf{I} - \frac{1}{2}\hat{L} & \hat{G} \\ \hat{D} & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{v}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \left[\frac{1}{\Delta t}\mathbf{I} - \frac{1}{2}\hat{L}\right]\boldsymbol{v}^{n} - \left[\frac{3}{2}\hat{\mathbf{H}}(\boldsymbol{v}^{n}) - \frac{1}{2}\hat{\mathbf{H}}(\boldsymbol{v}^{n-1})\right] \\ 0 \end{pmatrix} + \begin{pmatrix} \hat{bc}_1 \\ \hat{bc}_2 \end{pmatrix} \tag{6.5}$$

and BC (7.2b)

$$\begin{aligned} \frac{\boldsymbol{v}^{n+1} - \boldsymbol{v}^{n}}{\Delta t} + \left[\frac{3}{2}u^{n}\left(\frac{\partial \boldsymbol{v}}{\partial x}\right)^{n} - \frac{1}{2}u^{n-1}\left(\frac{\partial \boldsymbol{v}}{\partial x}\right)^{n-1}\right] \\ = \epsilon\left[\left(\frac{\partial^2 \boldsymbol{v}}{\partial y^2}\right)^{n+1} + \left(\frac{\partial^2 \boldsymbol{v}}{\partial y^2}\right)^{n}\right], \end{aligned} \tag{6.6}$$

however, the results produced with such BC (6.6) schemes are yet to be known and should be compared with BC (6.7) from the corresponding paper of Jin-Braza [7], which was said to be

$$\begin{aligned} \frac{\boldsymbol{v}^{n+1} - \boldsymbol{v}^{n}}{\Delta t} + \frac{u^{n}}{2}\left[\left(\frac{\partial \boldsymbol{v}}{\partial x}\right)^{n+1} + \left(\frac{\partial \boldsymbol{v}}{\partial x}\right)^{n}\right] \\ = \epsilon\left(\frac{\partial^2 \boldsymbol{v}}{\partial y^2}\right)^{n}. \end{aligned} \tag{6.7}$$

We will first use discretized BC (6.7) from Jin-Braza [7] as it was shown to be stable and produced solid results. Later on we can proceed with discretized BC (6.6) which is consistent with our schemes after developing the necessary techniques.

After applying the discrete operators listed in the following Sections 6.1 to 6.4, we rewrite Sys. (6.5) as

$$\begin{bmatrix} \hat{A} & \hat{G} \\ \hat{D} & 0 \end{bmatrix} \begin{pmatrix} \boldsymbol{v}^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} \hat{r}^n \\ 0 \end{pmatrix} + \begin{pmatrix} \hat{bc}_1 \\ \hat{bc}_2 \end{pmatrix}, \tag{6.8}$$

where $\hat{r}^n$ includes viscous, non-linear and gradient terms from time steps $n, n-1$ and $\hat{bc}_i$ are explicit boundary values in terms of velocity and pressure.

6.1 Laplacian

As per Sys. (6.5) it is required to approximate implicit viscous terms spatially. In order to obtain the discrete operator acting on a velocity vector we will rewrite Laplacian as a block matrix:

$$\hat{L} = \begin{bmatrix} \hat{L}^u_{xx} + \hat{L}^u_{yy} & 0 \\ 0 & \hat{L}^v_{xx} + \hat{L}^v_{yy} \end{bmatrix}.$$

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 similarly. Below we will discuss how the Laplacian matrix is constructed at different parts of the grid.

6.1.1 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 $\hat{L}^u_{xx}$. The other three matrices can be 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

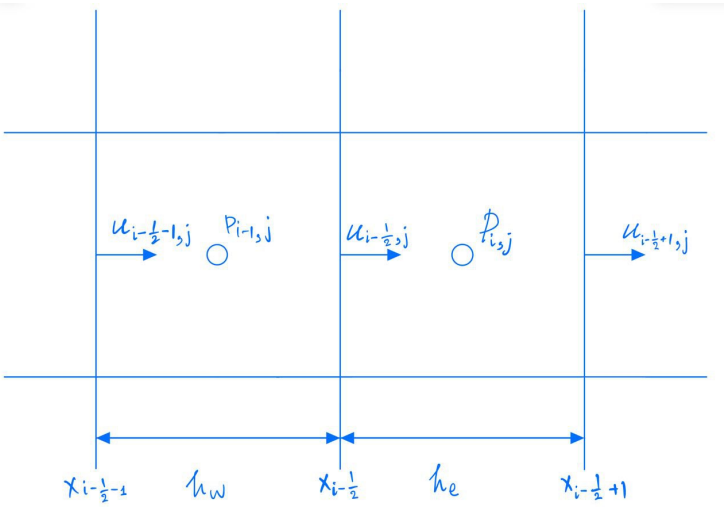


Figure 3: $\hat{L}^u_{xx}$ inner part.

$$u_{i-\frac{1}{2}+1,j} = u_{i-\frac{1}{2},j} + \left.\frac{\partial u}{\partial x}\right|_{i-\frac{1}{2},j} \left(x_{i-\frac{1}{2}+1} - x_{i-\frac{1}{2}}\right) + \frac{1}{2} \left.\frac{\partial^2 u}{\partial x^2}\right|_{i-\frac{1}{2},j} \left(x_{i-\frac{1}{2}+1} - x_{i-\frac{1}{2}}\right)^2 + O\left(\Delta x^3\right), \tag{6.9}$$

$$u_{i-\frac{1}{2}-1,j} = u_{i-\frac{1}{2},j} + \left.\frac{\partial u}{\partial x}\right|_{i-\frac{1}{2},j} \left(x_{i-\frac{1}{2}-1} - x_{i-\frac{1}{2}}\right) + \frac{1}{2} \left.\frac{\partial^2 u}{\partial x^2}\right|_{i-\frac{1}{2},j} \left(x_{i-\frac{1}{2}-1} - x_{i-\frac{1}{2}}\right)^2 + O\left(\Delta x^3\right). \tag{6.10}$$

We can combine Eqs. (6.9) and (6.10) by cancelling the first derivative, which will result in

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

$$\approx \frac{1}{h_c h_e} u_{i-\frac{1}{2}+1} - \frac{2}{h_w h_e} u_{i-\frac{1}{2}} + \frac{1}{h_c h_w} u_{i-\frac{1}{2}-1} + O(\Delta x), \tag{6.11}$$

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 Fig. 3. 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.

6.1.2 Left boundary

The exact value on the left boundary ($u_{\frac{1}{2},j}$ in a box on Fig. 4) 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 Sys. (6.5).

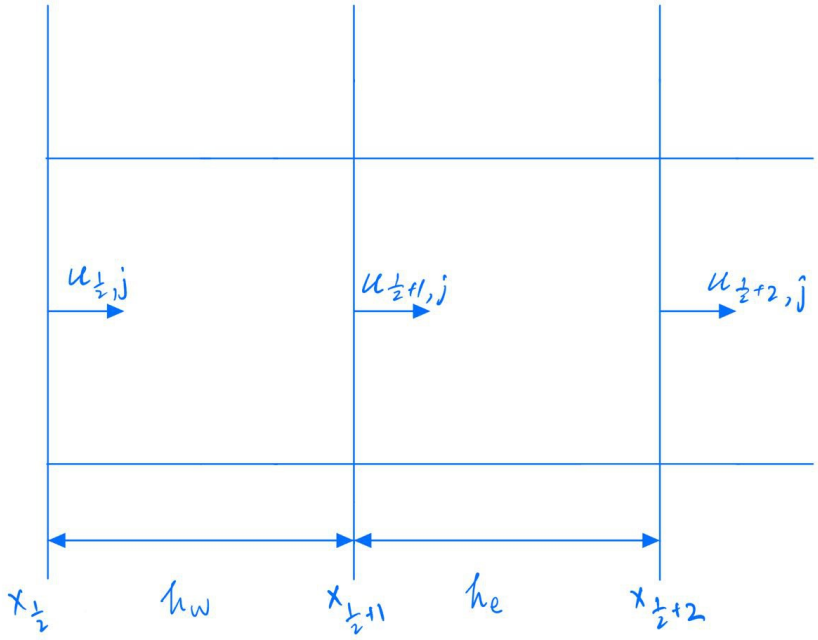


Figure 4: $\hat{L}^u_{xx}$ at the left boundary.

6.1.3 Top boundary

Most of the flow that differs much from the freestream is likely to happen closer to the wall, thus, we can incorporate symmetric boundary conditions over the top. BC (7.1) 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.

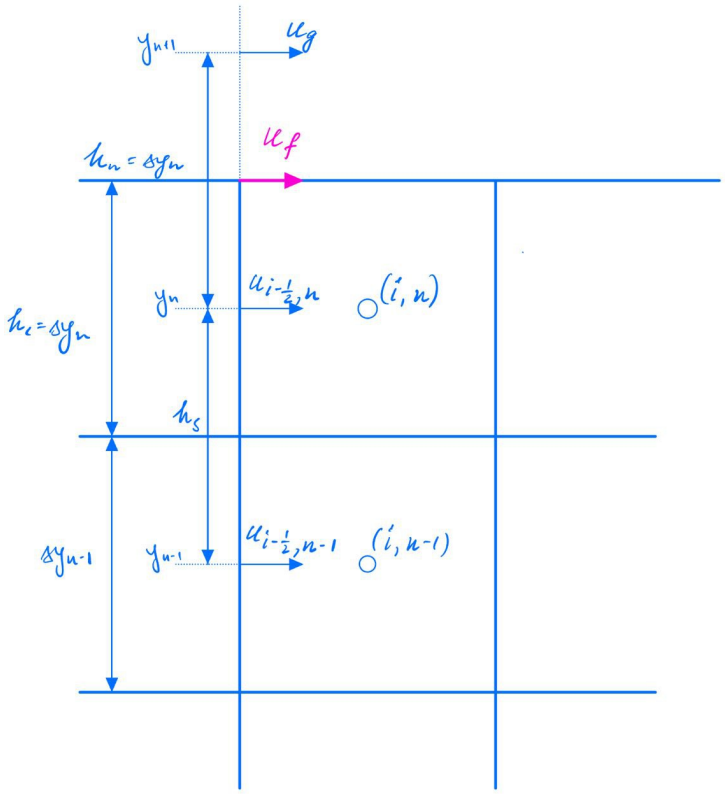


Figure 5: $\hat{L}^u_{yy}$ at top boundary.

Consider the neighbouring cells near the wall as in Fig. 5. Resultant Eq. (6.11) can be written as

$$\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},N} = \frac{u_g\,(h_s) + u_{i-\frac{1}{2},N}\,(-2h_c) + u_{i-\frac{1}{2},N-1}\,(h_n)}{h_n h_s h_c}, \tag{6.12}$$

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 Eq. (6.11) 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 freestream velocity $u_f = \frac{u_g + u_{i-\frac{1}{2},N}}{2} \implies u_g = 2u_f - u_{i-\frac{1}{2},N}$, which we can plug into Eq. (6.12) from above to obtain

$$\begin{aligned}
\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},N} &= \frac{2u_f h_s}{h_n h_s h_c} + u_{i-\frac{1}{2},N}\left(\frac{-h_s}{h_n h_s h_c} + \frac{-2h_c}{h_n h_s h_c}\right) + u_{i-\frac{1}{2},N-1}\frac{h_n}{u_c h_s h}, \\
\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},N} &= \frac{2u_f}{h_c h_c} + u_{i-\frac{1}{2},N}\left(\frac{-\left(2h_c + h_s\right)}{h_c h_s h_c}\right) + u_{i-\frac{1}{2},N-1}\frac{1}{h_s h_c}, \\
\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},N} &= \frac{2u_f}{{h_c}^2} + u_{i-\frac{1}{2},N}\left(\frac{-\left(2h_c + h_s\right)}{h_c^2 h_s}\right) + u_{i-\frac{1}{2},N-1}\frac{1}{h_s h_c}.
\end{aligned} \tag{6.13}$$

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 Sys. (6.5), while the other two coefficients are used as elements for the $\hat{L}^u_{yy}$ matrix.

6.1.4 Bottom boundary

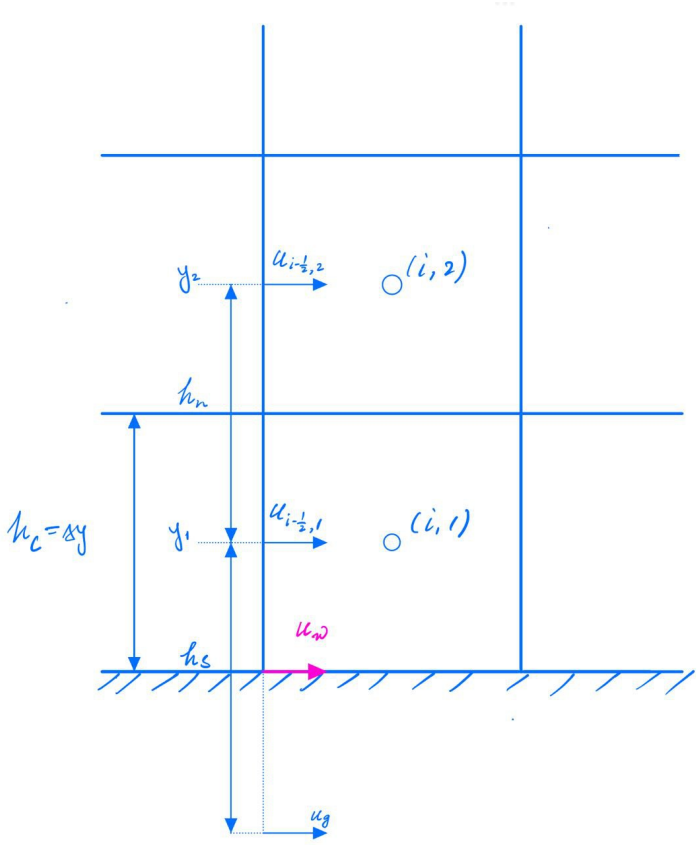


Figure 6: $\hat{L}^u_{yy}$ at the bottom boundary.

It is, in fact, possible to directly use the third velocity directly from the boundary. However, the order of such a scheme will then be reduced due to the large ratio of the distances between the neighbouring velocities. If, on the other hand, we introduce a ghost velocity u_g, we have the freedom of where to place it. Let us use Eq. (6.11) and change the distances between velocities as in Fig. 6, which leads to

$$\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},1} = u_{i-\frac{1}{2},2}\left(\frac{1}{h_n h_c}\right) + u_{i-\frac{1}{2},1}\left(\frac{-2}{h_n h_s}\right) + u_g\left(\frac{1}{h_s h_c}\right). \tag{6.14}$$

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 velocity at the wall $u_w = \frac{u_{i-\frac{1}{2},1}+u_g}{2} = 0 \Rightarrow u_g = -2u_{i-\frac{1}{2},1}$, then

$$\begin{aligned}\left.\frac{\partial^2 u}{\partial y^2}\right|_{i-\frac{1}{2},1} &= u_{i-\frac{1}{2},2}\left(\frac{1}{h_n h_c}\right) + u_{i-\frac{1}{2},1}\left(\frac{-2}{h_n h_s} + \frac{-2}{h_s h_c}\right) \\ &= u_{i-\frac{1}{2},2}\left(\frac{1}{h_n h_c}\right) + u_{i-\frac{1}{2},1}\left(\frac{-2h_c - 2h_n}{h_s h_c h_n}\right).\end{aligned} \tag{6.15}$$

The coefficients from Eq. (6.15) are distributed into the $\hat{L}^u_{yy}$ matrix similarly to previous boundary cases.

6.1.5 Right boundary.

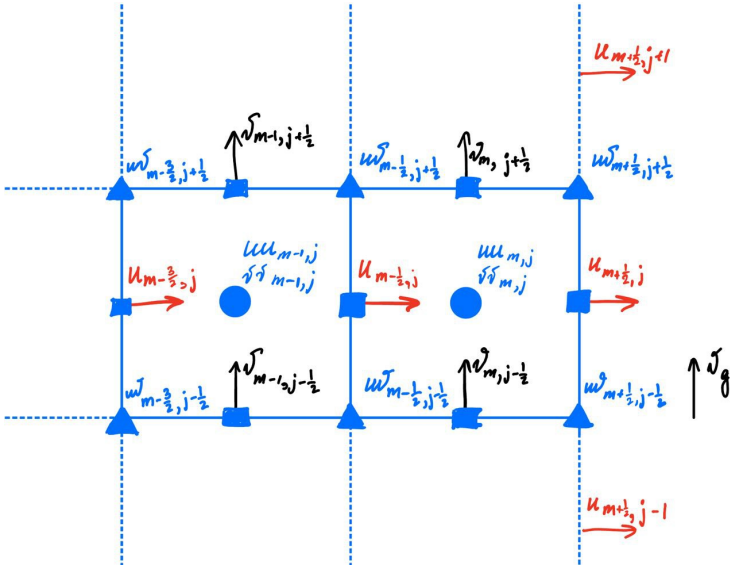


Figure 7: $\hat{L}^u_{xx}, \hat{L}^v_{xx}$ at the right boundary ($m = M$).

In order to express the rightmost unknown velocity component $u_{M-\frac{1}{2},j}$ inside the domain it is required to use the boundary values of $u_{M+\frac{1}{2},j}$ and ghost elements for v_g as in Fig. 7. The best case scenario would be to have Dirichlet velocity values $\boldsymbol{v}(t, 1, y) \subset \mathbb{R}$ or Neumann boundary condition $\frac{\partial u}{\partial x}\big|_{M+\frac{1}{2},j} = 0$ as in Colonius-Taira [3] at the outlet. However, since the solution of Eqs. (4.1) at the right boundary of the domain might not have the same behaviour, it is not possible to impose such boundary conditions for our problem statement. We will address the spatial discretization of Laplacian at the right boundary in Section 7.2.

6.2 Advection

Since Explicit Adams-Bashforth Eq. (6.3) uses information from time steps n and $n-1$, while we solve the system of equations for time step $n+1$ there is no need for upwinding or even more complicated schemes. For spatial discretization we will try to stick to the central schemes as they are more stable and less dependent on the flow direction. It is convenient to write such schemes for conservative form of the advection, moreover, this also keeps the error low, since there is no product of velocity with acceleration. Hence, let us rewrite advective derivatives in conservative form.

$$\begin{aligned} u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} &= u\frac{\partial u}{\partial x} + 0 + v\frac{\partial u}{\partial y} \\ &= u\frac{\partial u}{\partial x} + u\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}\right) + v\frac{\partial u}{\partial y} \\ &= \left(u\frac{\partial u}{\partial x} + u\frac{\partial u}{\partial x}\right) + \left(u\frac{\partial v}{\partial y} + v\frac{\partial u}{\partial y}\right) \\ &= \frac{\partial uu}{\partial x} + \frac{\partial uv}{\partial y}. \end{aligned} \tag{6.16}$$

Our goal is to compute advection components at the unknown velocity coordinates. Below we will describe the discretization schemes for Eq. (6.16).

6.2.1 Inner part

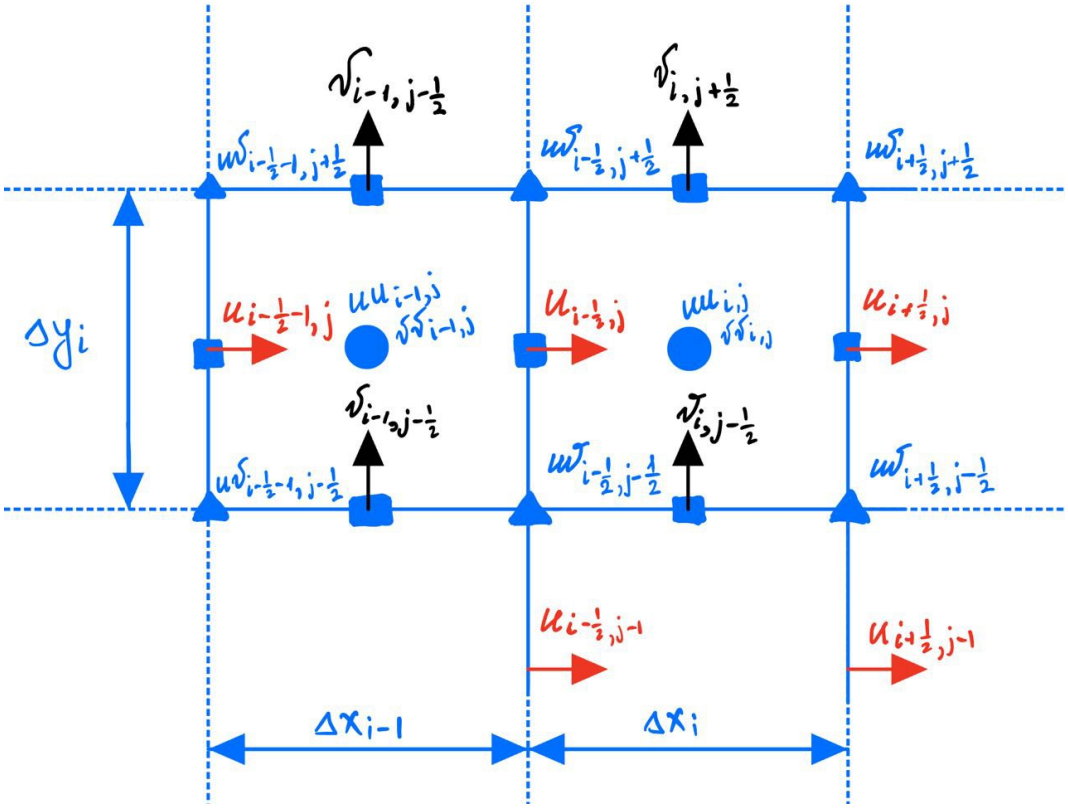


Figure 8: Advection discretization.

Advective component from x-momentum was expressed in conservative form by Eq. (6.16). It is discretized using the standard central differencing schemes [3] as

$$\left(u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}\right)_{i-\frac{1}{2},j}=\frac{(uu)_{i,j}-(uu)_{i-1,j}}{\frac{\Delta x_i+\Delta x_{i-1}}{2}}+\frac{(uv)_{i-\frac{1}{2},j+\frac{1}{2}}-(uv)_{i-\frac{1}{2},j-\frac{1}{2}}}{\Delta y_i}, \tag{6.17}$$

which is evaluated at the same points as $u_{i-\frac{1}{2},j}$ (squares in Fig. 8). We need to compute uv at the nodes and uu at cell centres, which are triangles and circles respectively in Fig. 8. Using linear interpolation [3] of velocity values leads to

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

6.2.2 Left boundary

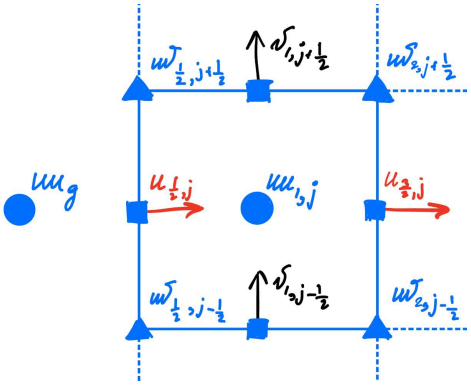


Figure 9: Left boundary advection.

The exact value of the uu_g element located outside the boundary (Fig. 9) 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.

6.2.3 Top boundary

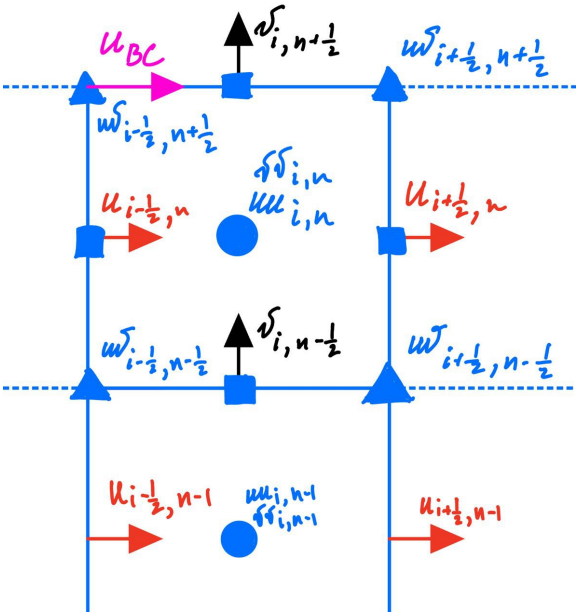


Figure 10: Top boundary advection.

Since the exact values of the velocity u_{BC} and $v_{BC} = 0$ in a freestream are known from BC (7.1), we can directly use them in our advection term computation making the second summand $uv = 0$, in particular

$$\left(\frac{\partial uu}{\partial x} + \frac{\partial uv}{\partial y}\right)_{i-\frac{1}{2},N} = 2\frac{(uu)_{i,N} - (uu)_{i-1,N}}{\Delta x_i + \Delta x_{i-1}} + 0. \tag{6.18}$$

6.2.4 Bottom boundary

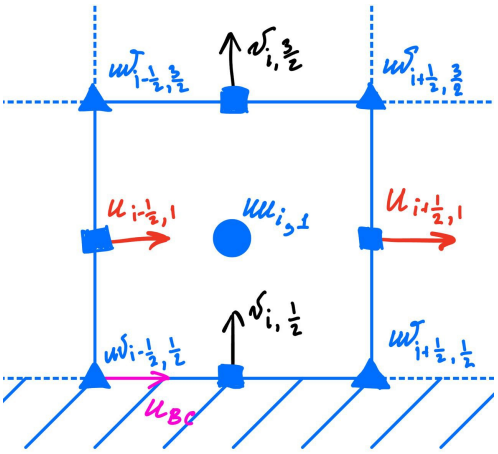


Figure 11: Bottom boundary advection

Computation of the advective terms just above the bottom boundary is the same as for the inner part, with the only exception being $uv_{i-\frac{1}{2},\frac{1}{2}} = 0$ at the boundary, due to the $u_{bc} = 0$ condition on the plate (purple on Fig. 11).

The advective terms in the y-momentum equation are computed similarly. After the derivatives are evaluated, they are moved to the right-hand side as in Sys. (6.5) and treated explicitly.

6.2.5 Right boundary

Similarly to Laplacian for the right side in Section 6.1.5 we need explicit values of u and v at and beyond the boundary. These values are yet to be known, hence, it is not possible to utilize the inner part advection formulas from Section 6.2.1 to rightmost unknown velocities at time steps $n, n-1$ directly. These values are to be determined using artificial boundary conditions in Section 7.3.

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. In this section we will show one side of why this identity is true, the other part could be found in Section 6.4. The second line of Sys. (6.1) 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

$$\Delta_{xy} = \begin{bmatrix} \frac{1}{\Delta x_1 \Delta y_1} & 0 & \dots & 0 \\ 0 & \frac{1}{\Delta x_2 \Delta y_1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{\Delta x_M \Delta y_N} \end{bmatrix} \tag{6.19}$$

is $MN \times MN$ 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 then 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 central difference schemes into

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

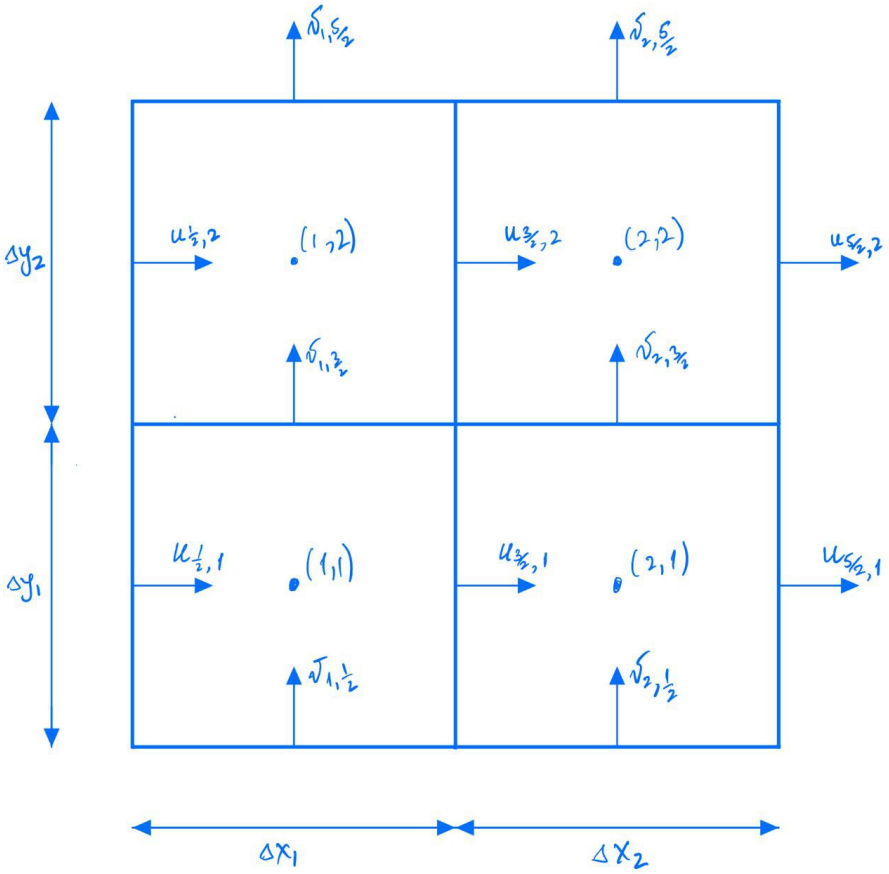


Figure 12: 2×2 grid example for divergence operator.

To provide a good visual example, consider 2×2 grid as in Fig. 12 with the same boundary conditions as in Eqs. (4.1). The partial derivatives of divergence operator evaluated at cell centres are:

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

After applying the boundary conditions to the above equations, we obtain the system

$$\begin{bmatrix} \frac{1}{\Delta x_1 \Delta y_1} & 0 & 0 & 0 \\ 0 & \frac{1}{\Delta x_2 \Delta y_1} & 0 & 0 \\ 0 & 0 & \frac{1}{\Delta x_1 \Delta y_2} & 0 \\ 0 & 0 & 0 & \frac{1}{\Delta x_2 \Delta y_2} \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 0 & -1 \end{bmatrix} \begin{bmatrix} u_{\frac{3}{2},1}\Delta y_1 \\ u_{\frac{3}{2},2}\Delta y_2 \\ v_{1,\frac{3}{2}}\Delta x_1 \\ v_{2,\frac{3}{2}}\Delta x_2 \end{bmatrix} = \begin{bmatrix} \frac{u_{\frac{1}{2},1}}{\Delta x_1} + \frac{v_{1,\frac{1}{2}}}{\Delta y_1} \\ -\frac{u_{\frac{5}{2},1}}{\Delta x_2} + \frac{v_{2,\frac{1}{2}}}{\Delta y_1} \\ \frac{u_{\frac{1}{2},2}}{\Delta x_1} - \frac{v_{1,\frac{5}{2}}}{\Delta y_2} \\ -\frac{u_{\frac{5}{2},2}}{\Delta x_2} - \frac{v_{2,\frac{5}{2}}}{\Delta y_2} \end{bmatrix}, \tag{6.20}$$

which in general case is written as $\frac{1}{\Delta_{xy}} Dq = \hat{b}c_2$. 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 Sys. (6.20).

6.4 Gradient

The pressure gradients are computed at the same coordinates as the unknown velocities according to Sys. (6.1). Below we will arrive to the conclusion that the discrete gradient and divergence operators satisfy

$$G = -D^T \tag{6.21}$$

on staggered/MAC grids.

As an illustrative example, we may consider a 2×2 grid as in Fig. 13 with the same boundary conditions as in Eqs. (4.1). It is required to determine the pressure gradients across each unknown velocity (square nodes on Fig. 13):

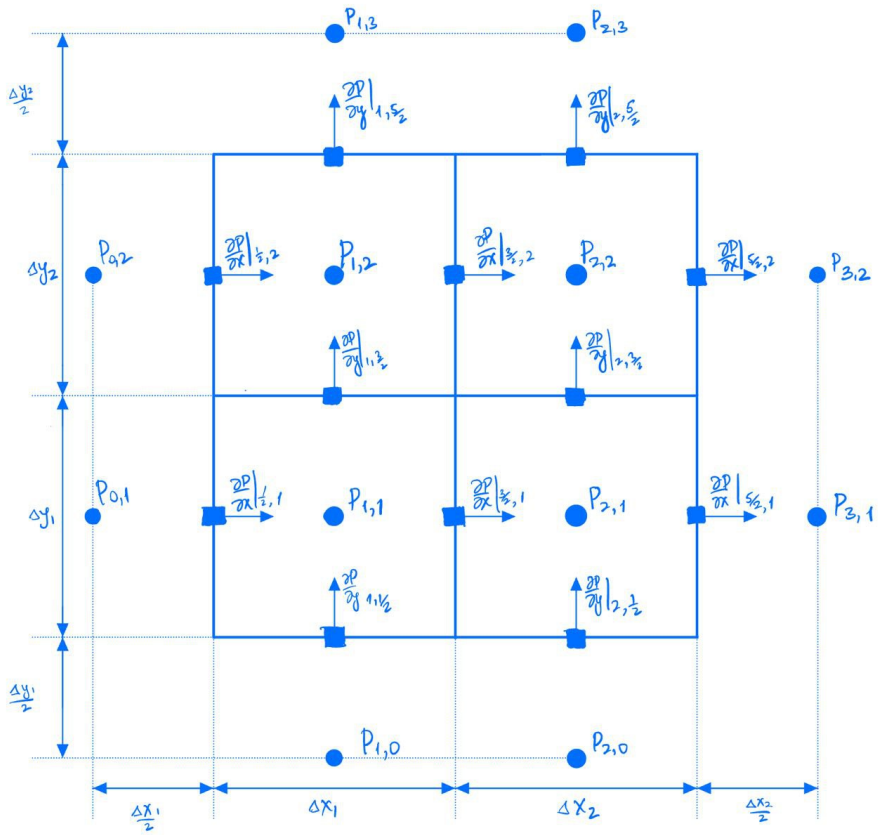


Figure 13: 2×2 grid example for gradient operator.

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

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

$$
\begin{bmatrix}
\frac{2}{\Delta x_1+\Delta x_2} & 0 & 0 & 0 \\
0 & \frac{2}{\Delta x_1+\Delta x_2} & 0 & 0 \\
0 & 0 & \frac{2}{\Delta y_1+\Delta y_2} & 0 \\
0 & 0 & 0 & \frac{2}{\Delta y_1+\Delta y_2}
\end{bmatrix}
\begin{bmatrix}
-1 & 1 & 0 & 0 \\
0 & 0 & -1 & 1 \\
-1 & 0 & 1 & 0 \\
0 & -1 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
P_{1,1} \\ P_{2,1} \\ P_{1,2} \\ P_{2,2}
\end{bmatrix}
=
\begin{bmatrix}
0 \\ 0 \\ 0 \\ 0
\end{bmatrix},
\tag{6.22}
$$

which can be rewritten as

$$
\hat{M}^{-1} G
\begin{bmatrix}
p_{1,1} \\ p_{1,2} \\ \vdots \\ p_{MN}
\end{bmatrix}
= \hat{bc}_p,
\tag{6.23}
$$

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. There are no pressure BC values in $\hat{bc}_p$ since no pressure gradient has to be computed across the boundary. It can be observed from Sys. (6.20) and (6.22) that Eq. (6.21) holds and the general case is true by construction.

7 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 [12] and Tsynkov [13] list and compare different artificial boundary conditions (ABC) applied to the various flow scenarios. Below we will discuss the options that are most applicable to Eqs. (2.1).

In solutions of fluid flow, the reflection of waves at the outlet is common. Such behaviour may lead to issues due to incompressibility of the Navier-Stokes equations, i.e. a change of solution near the outlet will affect the solution at the inner part of the domain as well. Engquist and Majda [4] introduced one of the earliest and successful tools to address this problem. The authors derive a hierarchy of highly absorbing local boundary conditions, serving as approximations to the theoretical nonlocal boundary conditions which were historically used as the only non-reflective options. Engquist and Majda admit that it is not possible to eliminate all of the reflections, however, it is possible to come up with condition such that for an arbitrary highly singular wave packet only low amplitude smooth reflections occur. The main idea of the paper is to transition from space-time domain into frequency domain, factorize the equation at the boundary (within a smooth error) and eliminate the reflected waves. The methodology outlined was employed by Engquist and Majda to construct absorbing artificial boundaries tailored for the acoustic wave equation in both Cartesian and polar coordinates, as well as for the linearized shallow water equations in two dimensions. Although the method is not fully applicable to our problem, the authors systematically addressed and resolved the reflective behaviour of the outlet.

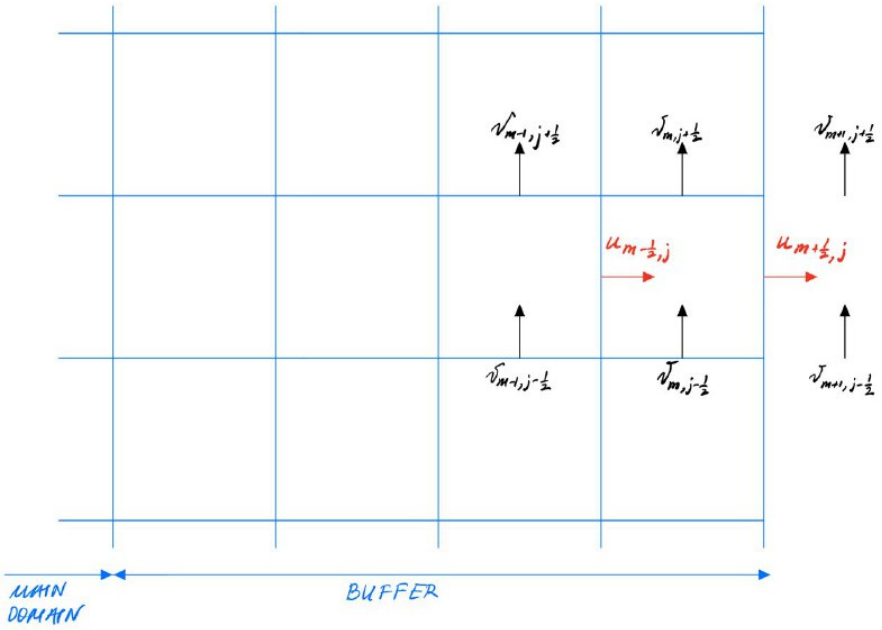


Figure 14: Buffer BC by Liu.

Another way of overcoming the reflective behaviour of the fluid flow was developed by Liu [10]. They 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. 14) at the exit that involves the use of:

1. Semi-implicit perturbed mass conservation (outlet velocity is updated using the data from inner part at previous time levels)

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

2. Linear profile of the tangential velocity component computed semi-implicitly (tangential ghost velocity is updated using the data from inner part at previous time levels) 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 [7, 8, 11] considered the shear flow (mixing layer) problem. Authors constructed a set of BCs that allows the eddies to be freely developed downstream as they leave the domain. Furthermore, the most appropriate boundary conditions at the top and bottom were those derived by considering boundaries as streamlines with

$$v = 0 \text{ and } \frac{\partial u}{\partial y} = 0, \tag{7.1}$$

while the non-reflective outlet had the following boundary conditions:

$$\text{Type I: } \frac{\partial^2 u}{\partial x^2} = 0, \frac{\partial v}{\partial x} = 0, \tag{7.2a}$$

$$\text{Type II: } \frac{\partial \boldsymbol{v}}{\partial t} + u\frac{\partial \boldsymbol{v}}{\partial x} - \nu\frac{\partial^2 \boldsymbol{v}}{\partial y^2} = 0, \tag{7.2b}$$

where ν is kinematic viscosity. Plots shown in Figs. 15(a) and 15(b) correspond to results obtained through BCs (7.2a) and (7.2b). The first BC (7.2a) made vortices hit against a "wall", while they should have left the domain. However, the second BC (7.2b) was derived similarly to Engquist and Majda [4] and displayed more non-reflective behaviour.

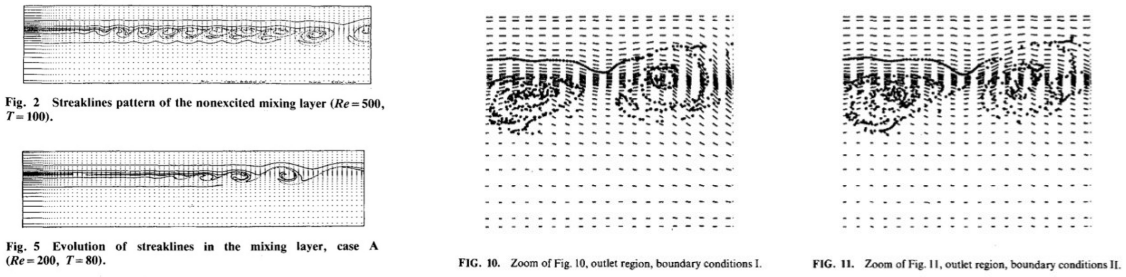


Fig. 2 Streaklines pattern of the nonexcited mixing layer ($Re = 500$, $T = 100$).

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

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

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

(a) Kourta-Braza results.

(b) Jin-Braza-Braza results.

Figure 15: Resultant streaklines of Braza and collaborators.

Taking into account successful results obtained in [7, 8, 11] and similarities between the boundary layer and mixing layer flows we will use [1] BCs (7.1) and (7.2b) as boundary conditions for top and right boundaries respectively and also truncate the domain to width and height L.

[1] G. Fournier and F. Golanski (2008) claim that Prandtl BL solution for advective BCs s.a. $\frac{\partial \boldsymbol{v}}{\partial t} + u\frac{\partial \boldsymbol{v}}{\partial x} = 0$ results in wrong solution for pressure gradient, while their condition $\frac{\partial \boldsymbol{v}}{\partial t} + u\frac{\partial \boldsymbol{v}}{\partial x} + v\frac{\partial \boldsymbol{v}}{\partial y} - \nu\frac{\partial^2 \boldsymbol{v}}{\partial y^2} = 0$ matches Prandtl BL. Such BC could be used as an alternative option if BC (7.2b) from Jin-Braza [7] will fail.

7.1 Derivation of Type II ABC (as in paper)

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

$$\frac{\partial^2 \boldsymbol{v}}{\partial t^2} - c_x^2 \frac{\partial^2 \boldsymbol{v}}{\partial x^2} - c_y^2 \frac{\partial^2 \boldsymbol{v}}{\partial y^2} = 0, \tag{7.3}$$

where c_x, c_y are the characteristic velocities of x, y wave propagation. The presence of the second time derivative seems to be an "ad-hoc guess" which worked for Jin-Braza [7].

We can rewrite Eq. (7.3) in terms of pseudo-differential operator

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

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 [4] at the boundary

$$L^+ \boldsymbol{v} = 0 \tag{7.4}$$

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

$$\sqrt{1 - \left(\frac{c_y D_y}{D_t}\right)^2} \approx 1 - \frac{1}{2}\left(\frac{c_y D_y}{D_t}\right)^2 \tag{7.5}$$

turned out to be well posed [4, 9] . It is possible to rewrite Eq. (7.4) in terms of Eq. (7.5) as

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

which can be matched to the momentum Navier-Stokes equation and nondimensionalized as

$$\frac{\partial \boldsymbol{v}}{\partial t} + u \frac{\partial \boldsymbol{v}}{\partial x} - \epsilon \frac{\partial^2 \boldsymbol{v}}{\partial y^2} = 0. \tag{7.6}$$

7.2 Laplacian at the right boundary using ABC

The rightmost unknown element of Sys. (6.5) in x direction is $u^{n+1}_{M-\frac{1}{2},j}$ on Fig. 16. Denote the vertical distances between three neighbouring u components as h_n and h_s, which stand for distances to the north $u^n_{M+\frac{1}{2},j+1}$ and south $u^n_{M+\frac{1}{2},j-1}$ neighbours from the $u^n_{M+\frac{1}{2},j}$ between them, while central h_c is the average of h_s and h_n. To apply Eq. (6.11) we need to express $u^{n+1}_{M+\frac{1}{2},j}$ using BC (6.7).

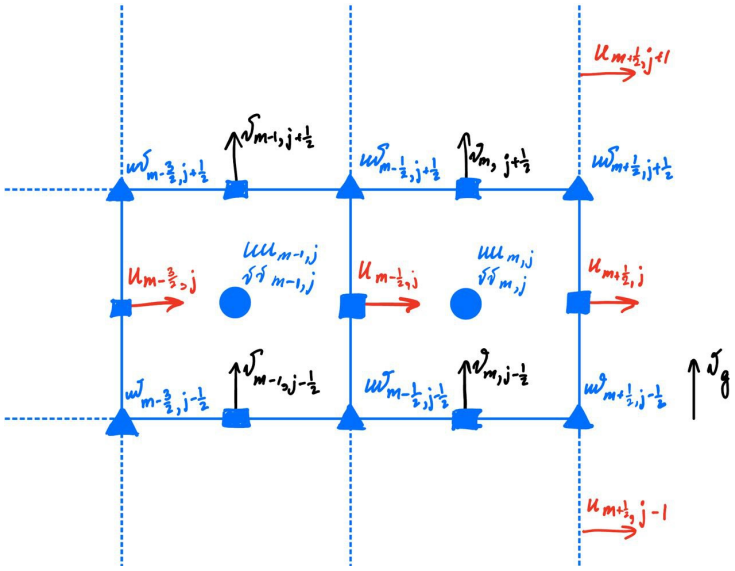


Figure 16: $\hat{L}^u_{xx}, \hat{L}^v_{xx}$ at the right boundary ($m = M$).

$$\frac{u^{n+1}_{M+\frac{1}{2},j} - u^n_{M+\frac{1}{2},j}}{\Delta t} + \frac{u^n_{M+\frac{1}{2},j}}{2}\left[\left(\frac{u^{n+1}_{M+\frac{1}{2},j} - u^{n+1}_{M-\frac{1}{2},j}}{\Delta x_M}\right) + \left(\frac{u^n_{M+\frac{1}{2},j} - u^n_{M-\frac{1}{2},j}}{\Delta x_M}\right)\right] =$$
$$= \epsilon\left(\frac{1}{h_c h_s}u^n_{M+\frac{1}{2},j-1} - \frac{2}{h_n h_s}u^n_{M+\frac{1}{2},j} + \frac{1}{h_c h_n}u^n_{M+\frac{1}{2},j+1}\right)$$

leads to

$$\left(\frac{1}{\Delta t} + \frac{u^n_{M+\frac{1}{2},j}}{2\Delta x_M}\right)u^{n+1}_{M+\frac{1}{2},j} = u^{n+1}_{M-\frac{1}{2},j}\frac{u^n_{M+\frac{1}{2},j}}{2\Delta x_M} -$$
$$-\frac{u^n_{M+\frac{1}{2},j}}{2}\frac{u^n_{M+\frac{1}{2},j} - u^n_{M-\frac{1}{2},j}}{\Delta x_M} + \frac{u^n_{M+\frac{1}{2},j}}{\Delta t} + \epsilon\left(\frac{1}{h_c h_s}u^n_{M+\frac{1}{2},j-1} - \frac{2}{h_n h_s}u^n_{M+\frac{1}{2},j} + \frac{1}{h_c h_n}u^n_{M+\frac{1}{2},j+1}\right). \tag{7.7}$$

Note that $\left(\frac{1}{\Delta t} + \frac{u^n_{M+\frac{1}{2},j}}{2\Delta x_M}\right)$ can be made to never be zero regardless of the flow direction, hence dividing by it is allowed. The second line of fully discretized BC (7.7) is treated explicitly since we posses the flow data at time steps n and earlier. Denote explicit terms as

$$bc^n_{u-\text{explicit}} = -\frac{u^n_{M+\frac{1}{2},j}}{2}\frac{u^n_{M+\frac{1}{2},j} - u^n_{M-\frac{1}{2},j}}{\Delta x_M} + \frac{u^n_{M+\frac{1}{2},j}}{\Delta t} + \epsilon\left(\frac{1}{h_c h_s}u^n_{M+\frac{1}{2},j-1} - \frac{2}{h_n h_s}u^n_{M+\frac{1}{2},j} + \frac{1}{h_c h_n}u^n_{M+\frac{1}{2},j+1}\right),$$

which simplifies BC (7.7) to

$$\left(\frac{1}{\Delta t} + \frac{u^n_{M+\frac{1}{2},j}}{2\Delta x_M}\right)u^{n+1}_{M+\frac{1}{2},j} = u^{n+1}_{M-\frac{1}{2},j}\frac{u^n_{M+\frac{1}{2},j}}{2\Delta x_M} + bc^n_{u-\text{explicit}}. \tag{7.8}$$

Next, we plug BC (7.8) into Eq. (6.11), which finally leads us to

$$\begin{aligned}
\left.\frac{\partial^2 u}{\partial x^2}\right|^{n+1}_{M-\frac{1}{2},j} &= \frac{1}{h_c h_e} u^{n+1}_{M+\frac{1}{2},j} - \frac{2}{h_w h_e} u^{n+1}_{M-\frac{1}{2},j} + \frac{1}{h_c h_w} u^{n+1}_{M-\frac{1}{2}-1,j} + h.o.t. \\
&= \frac{1}{h_c h_e} \frac{u^{n+1}_{M-\frac{1}{2},j} \frac{u^n_{M+\frac{1}{2},j}}{2h_e} + bc^n_{u-\text{explicit}}}{\frac{1}{\Delta t} + \frac{u^n_{M+\frac{1}{2},j}}{2\Delta x_M}} - \frac{2}{h_w h_e} u^{n+1}_{M-\frac{1}{2},j} + \frac{1}{h_c h_w} u^{n+1}_{M-\frac{1}{2}-1,j} + h.o.t. \\
&= \left[\frac{1}{h_c h_e}\left(\frac{u^n_{M+\frac{1}{2},j}}{2h_e\left[\frac{1}{\Delta t} + \frac{u^n_{M+\frac{1}{2},j}}{2\Delta x_M}\right]}\right) - \frac{2}{h_w h_e}\right] u^{n+1}_{M-\frac{1}{2},j} + \frac{1}{h_c h_w} u^{n+1}_{M-\frac{1}{2}-1,j} + \frac{bc^n_{u-\text{explicit}}}{h_c h_e\left(\frac{1}{\Delta t} + \frac{u^n_{M+\frac{1}{2},j}}{2\Delta x_M}\right)} + h.o.t..
\end{aligned} \tag{7.9}$$

Condition for v component is not as trivial as for u since there is no v directly at the boundary due to the staggered grid arrangement. We need to express $v_g = v_{M+1,j-\frac{1}{2}}$ in terms of the right-most unknown $v_{M,j-\frac{1}{2}}$ component for the diffusion operator. We can use more than one value of v from inner part of the domain to express the ghost velocity outside. However, this will lead to change of non-diagonal elements in diffusion operator, which will cause Laplacian matrix to be asymmetric. Therefore, we are restricted to use the elements of the main diagonal to express the ghost velocities outside the computational domain. Hence, interpolation leads to

$$v_{\text{BC}} = v_{M+\frac{1}{2},j-\frac{1}{2}} = \frac{v_{M+1,j-\frac{1}{2}} + v_{M,j-\frac{1}{2}}}{2},$$

where $v_{M+1,j-\frac{1}{2}}$ is ghost component outside the domain. It is also possible to apply central difference to the advection terms at the boundary $\left(\frac{\partial v}{\partial x}\right)_{M+\frac{1}{2},j-\frac{1}{2}} = \frac{v_{M+1,j-\frac{1}{2}} - v_{M,j-\frac{1}{2}}}{\Delta x_{M+\frac{1}{2}}}$.

BC (6.7) using the approximations from above can be rewritten for $v_{M+\frac{1}{2},j-\frac{1}{2}}$ component at the boundary as

$$\begin{aligned}
&\frac{\left(\frac{v^{n+1}_{M+1,j-\frac{1}{2}} + v^{n+1}_{M,j-\frac{1}{2}}}{2}\right) - \left(\frac{v^n_{M+1,j-\frac{1}{2}} + v^n_{M,j-\frac{1}{2}}}{2}\right)}{\Delta t} + \\
&+ \frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2}\left[\left(\frac{v^{n+1}_{M+1,j-\frac{1}{2}} - v^{n+1}_{M,j-\frac{1}{2}}}{\Delta x_{M+\frac{1}{2}}}\right) + \left(\frac{v^n_{M+1,j-\frac{1}{2}} - v^n_{M,j-\frac{1}{2}}}{\Delta x_{M+\frac{1}{2}}}\right)\right] = \\
= \epsilon\left[\frac{1}{h_c h_s}\left(\frac{v^n_{M+1,j-\frac{1}{2}-1} + v^n_{M,j-\frac{1}{2}-1}}{2}\right) - \frac{2}{h_n h_s}\left(\frac{v^n_{M+1,j-\frac{1}{2}} + v^n_{M,j-\frac{1}{2}}}{2}\right) + \frac{1}{h_c h_n}\left(\frac{v^n_{M+1,j+\frac{1}{2}} + v^n_{M,j+\frac{1}{2}}}{2}\right)\right],
\end{aligned}$$

which is used to express ghost velocity $v^{n+1}_{M+1,j-\frac{1}{2}}$ after rearranging

$$\begin{aligned}
&\left(\frac{1}{2\Delta t} + \frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2\Delta x_{M+\frac{1}{2}}}\right) v^{n+1}_{M+1,j-\frac{1}{2}} = \left(-\frac{1}{2\Delta t} + \frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2\Delta x_{M+\frac{1}{2}}}\right) v^{n+1}_{M,j-\frac{1}{2}} + \\
&+ \left(\frac{v^n_{M+1,j-\frac{1}{2}} + v^n_{M,j-\frac{1}{2}}}{2\Delta t}\right) + \frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2}\left(\frac{v^n_{M+1,j-\frac{1}{2}} - v^n_{M,j-\frac{1}{2}}}{\Delta x_{M+\frac{1}{2}}}\right) + \\
&+\epsilon\left[\frac{1}{h_c h_s}\left(\frac{v^n_{M+1,j-\frac{1}{2}-1} + v^n_{M,j-\frac{1}{2}-1}}{2}\right) - \frac{2}{h_n h_s}\left(\frac{v^n_{M+1,j-\frac{1}{2}} + v^n_{M,j-\frac{1}{2}}}{2}\right) + \frac{1}{h_c h_n}\left(\frac{v^n_{M+1,j+\frac{1}{2}} + v^n_{M,j+\frac{1}{2}}}{2}\right)\right].
\end{aligned} \tag{7.10}$$

Let us denote explicit terms from last two rows as

$$\begin{aligned}
bc^n_{v-\text{explicit}} &= \left(\frac{v^n_{M+1,j-\frac{1}{2}} + v^n_{M,j-\frac{1}{2}}}{2\Delta t}\right) + \frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2}\left(\frac{v^n_{M+1,j-\frac{1}{2}} - v^n_{M,j-\frac{1}{2}}}{\Delta x_{M+\frac{1}{2}}}\right) + \\
&+\epsilon\left[\frac{1}{h_c h_s}\left(\frac{v^n_{M+1,j-\frac{1}{2}-1} + v^n_{M,j-\frac{1}{2}-1}}{2}\right) - \frac{2}{h_n h_s}\left(\frac{v^n_{M+1,j-\frac{1}{2}} + v^n_{M,j-\frac{1}{2}}}{2}\right) + \frac{1}{h_c h_n}\left(\frac{v^n_{M+1,j+\frac{1}{2}} + v^n_{M,j+\frac{1}{2}}}{2}\right)\right].
\end{aligned}$$

Then BC (7.10) simplifies to

$$\left(\frac{1}{2\Delta t}+\frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2\Delta x_{M+\frac{1}{2}}}\right)v^{n+1}_{M+1,j-\frac{1}{2}}=\left(-\frac{1}{2\Delta t}+\frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2\Delta x_{M+\frac{1}{2}}}\right)v^{n+1}_{M,j-\frac{1}{2}}+bc^n_{v-\text{explicit}}. \tag{7.11}$$

The ultimate step is to plug BC (7.11) into Eq. (6.11) which will result in diffusion approximation of v in x direction of the right-most unknown component

$$\begin{aligned}\left.\frac{\partial^2 v}{\partial x^2}\right|^{n+1}_{M,j-\frac{1}{2}} &= \frac{1}{h_c h_e}v^{n+1}_{M+1,j-\frac{1}{2}}-\frac{2}{h_w h_e}v^{n+1}_{M,j-\frac{1}{2}}+\frac{1}{h_c h_w}v^{n+1}_{M-1,j-\frac{1}{2}}+h.o.t.\\ &= \frac{1}{h_c h_e}\left[\frac{\left(-\frac{1}{2\Delta t}+\frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2h_e}\right)v^{n+1}_{M,j-\frac{1}{2}}+bc^n_{v-\text{explicit}}}{\frac{1}{2\Delta t}+\frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2h_e}}\right]-\frac{2}{h_w h_e}v^{n+1}_{M,j-\frac{1}{2}}+\frac{1}{h_c h_w}v^{n+1}_{M-1,j-\frac{1}{2}}+h.o.t.\\ &= \left[\frac{1}{h_c h_e}\left(\frac{-\frac{1}{2\Delta t}+\frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2h_e}}{\frac{1}{2\Delta t}+\frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2h_e}}\right)-\frac{2}{h_w h_e}\right]v^{n+1}_{M,j-\frac{1}{2}}+\frac{1}{h_c h_w}v^{n+1}_{M-1,j-\frac{1}{2}}+\frac{bc^n_{v-\text{explicit}}}{h_c h_e\left(\frac{1}{2\Delta t}+\frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2h_e}\right)}+h.o.t..\end{aligned} \tag{7.12}$$

Let us now try to apply BC (6.6) which is consistent with schemes for the inner part to the boundary values at the outlet

$$\begin{aligned}&\frac{u^{n+1}_{M+\frac{1}{2},j}-u^n_{M+\frac{1}{2},j}}{\Delta t}+\left[\frac{3}{2}u^n_{M+\frac{1}{2},j}\left(\frac{\partial u}{\partial x}\right)^n_{M+\frac{1}{2},j}-\frac{1}{2}u^{n-1}_{M+\frac{1}{2},j}\left(\frac{\partial u}{\partial x}\right)^{n-1}_{M+\frac{1}{2},j}\right]\\ &\qquad=\epsilon\left[\left(\frac{\partial^2 u}{\partial y^2}\right)^{n+1}_{M+\frac{1}{2},j}+\left(\frac{\partial^2 u}{\partial y^2}\right)^{n}_{M+\frac{1}{2},j}\right].\end{aligned}$$

The above equation after discretizing spatially becomes

$$\begin{aligned}&\frac{\epsilon}{h_c h_n}u^{n+1}_{M+\frac{1}{2},j+1}-\left(\frac{2\epsilon}{h_s h_n}-\frac{1}{\Delta t}\right)u^{n+1}_{M+\frac{1}{2},j}+\frac{\epsilon}{h_c h_s}u^{n+1}_{M+\frac{1}{2},j-1}\\ &=-\frac{u^n_{M+\frac{1}{2},j}}{\Delta t}+\left[\frac{3}{2}u^n_{M+\frac{1}{2},j}\left(\frac{u^n_{M+\frac{1}{2},j}-u^n_{M-\frac{1}{2},j}}{\Delta x_M}\right)-\frac{1}{2}u^{n-1}_{M+\frac{1}{2},j}\left(\frac{u^{n-1}_{M+\frac{1}{2},j}-u^{n-1}_{M-\frac{1}{2},j}}{\Delta x_M}\right)\right]\\ &\quad-\left[\frac{\epsilon}{h_c h_n}u^n_{M+\frac{1}{2},j+1}-\frac{2\epsilon}{h_s h_n}u^n_{M+\frac{1}{2},j}+\frac{\epsilon}{h_c h_s}u^n_{M+\frac{1}{2},j-1}\right].\end{aligned} \tag{7.13}$$

The fully discretized BC (7.13) from above is a system of linear equations at time step $n+1$ and only has to be solved for the outlet boundary values of u. We know both topmost velocity $u_{\text{top}}=u_{\text{freestream}}$ and bottommost $u_{\text{bot}}=u_{\text{wall}}$ which play roles of boundary values for this system of equations. However, there is an issue that no velocity values at time step $n+1$ are passed from the inner part of the domain, the whole equation is self-sufficient and takes only the inner data from previous time steps n and $n-1$. Thus, BC (7.13) behaves as an independent equation from main Sys. (6.5).

7.3 Advection at the right boundary using ABC

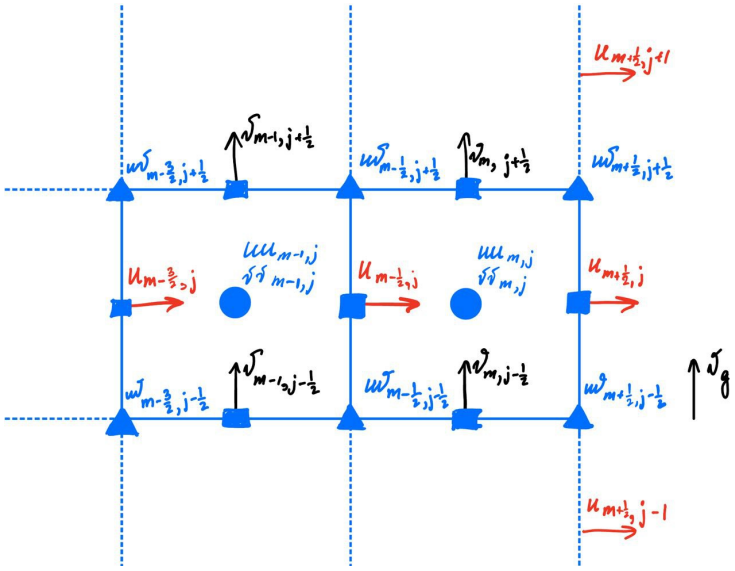


Figure 17: Right boundary advection.

BC (7.8) rewritten for the time step n is

$$u^n_{M+\frac{1}{2},j}\left(\frac{1}{\Delta t}+\frac{u^{n-1}_{M+\frac{1}{2},j}}{2\Delta x_M}\right)=u^n_{M-\frac{1}{2},j}\frac{u^{n-1}_{M+\frac{1}{2},j}}{2\Delta x_M}+bc^{n-1}_e.$$

This can be used to obtain the value $u^n_{M+\frac{1}{2},j}$ at the right boundary. Then we use the Eq. (6.17) for inner part as now all the necessary values are known. Similarly, BC (7.11) at time step n

$$\left(\frac{1}{2\Delta t}+\frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2\Delta x_{M+\frac{1}{2}}}\right)v^n_{M+1,j-\frac{1}{2}}=\left(-\frac{1}{2\Delta t}+\frac{u^n_{M+\frac{1}{2},j-\frac{1}{2}}}{2\Delta x_{M+\frac{1}{2}}}\right)v^n_{M,j-\frac{1}{2}}+bc^{n-1}_{ve}$$

is used to express ghost velocity $v^n_{M+1,j-\frac{1}{2}}$, which we use to compute Eq. (6.17) for inner part.

8 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

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

Define matrices

$$\Delta y_j = \text{diag}([\underbrace{\Delta y_1; \Delta y_1; \dots; \Delta y_1}_{M-1 \text{ times}}; \underbrace{\Delta y_2; \Delta y_2; \dots; \Delta y_2}_{M-1 \text{ times}}; \dots; \underbrace{\Delta y_N; \Delta y_N; \dots; \Delta y_N}_{M-1 \text{ times}}]),$$
$$\Delta x_i = \text{diag}([\underbrace{\Delta x_1; \Delta x_2; \dots \Delta x_M; \Delta x_1; \Delta x_2; \dots \Delta x_M; \dots; \Delta x_1; \Delta x_2; \dots; \Delta x_M}_{N-1 \text{ blocks of } 1,\dots,M}]),$$

which are then combined into

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

By moving to new variable $q = R\boldsymbol{v}$, we use matrix R, which inverse R^{-1} performs half of the symmetrization process for $\hat{L}$. Here, R is a diagonal matrix of size $(M-1)N \times M(N-1)$. Using the R matrix we can express mass flux $q^{n+1} = R\boldsymbol{v}^{n+1} \implies \boldsymbol{v}^{n+1} = R^{-1}q^{n+1}$. It is required that the difference matrices cancel out $\frac{x_{i+1}-x_{i-1}}{2}$ central term in diffusion discretization to make Laplacian partially symmetric. Define the elements of diagonal mass matrix as

$$\Delta x_i + \Delta x_{i-1} = \text{diag}[\underbrace{\Delta x_2 + \Delta x_1; \Delta x_3 + \Delta x_2; \dots; \Delta x_M + \Delta x_{M-1}; \dots; \Delta x_2 + \Delta x_1; \Delta x_3 + \Delta x_2; \dots \Delta x_M + \Delta x_M}_{N \text{ times for each block of } \Delta x_2+\Delta x_1 \text{ to } \Delta x_M+\Delta x_{M-1}}]$$

and

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

which are then combined into

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

which both removes the M^{-1} in $\hat{G} = \hat{M}^{-1}G$ if left multiplied and completes the symmetrization process of $\hat{L}$.

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

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

premultiplying by $\hat{M}$ we obtain

$$\hat{M}\hat{L}R^{-1}q,$$

where $\hat{M}LR^{-1}$ is symmetric by construction, hence, $\hat{M}\hat{A}R^{-1} = \hat{M}(\frac{1}{\Delta t}\mathbf{I} - \frac{1}{2}L)R^{-1}$ is also symmetric.

Using the above transformations we can modify Sys. (6.8) into

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

which can be rewritten as

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{pmatrix} q^{n+1} \\ p^{n+1} \end{pmatrix} = \begin{pmatrix} r^n \\ 0 \end{pmatrix} + \begin{pmatrix} bc_1 \\ bc_2 \end{pmatrix}, \tag{8.1}$$

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 Sys. (6.22). It is also possible to transform the divergence operator $\hat{D}$ with non-integer coefficients into D with integer coefficients in continuity part of Sys. (8.1) using Sys. (6.20). Multiplying both sides of continuity equation by Δ_{xy} matrix from Eq. (6.19) leads to

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

9 Vorticity-stream function formulation

The algorithm to be used is a combination of classical formulation as in Eqs. (1.1) and vorticity-stream function Eqs. (9.3) and (9.4). Consider:

1. Stream function $\psi(x, y, t) : \boldsymbol{v} = \nabla \times \psi$ of an incompressible two-dimensional flow:

$$u = \frac{\partial \psi}{\partial y}, \quad v = -\frac{\partial \psi}{\partial x}, \tag{9.1}$$

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

   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

$$\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}. \tag{9.2}$$

Importantly, continuity Eq. (1.1b) is satisfied immediately by taking spatial derivatives of stream function Eq. (9.1) components and adding them up. The governing Eqs. (1.1) can now be transformed. The new equations are changed to the following:

1. Transport equation for vorticity.

   Application of $(\nabla\times)$ to momentum and taking into account continuity Eq. (1.1b) 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

$$\boxed{\frac{\partial \omega}{\partial t} + u\frac{\partial \omega}{\partial x} + v\frac{\partial \omega}{\partial y} = \epsilon\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right).} \tag{9.3}$$

2. Vorticity-Stream function equation.

   After substituting stream function Eq. (9.1) into vorticity Eq. (9.2) we obtain

$$\boxed{\nabla^2 \psi = -\omega.} \tag{9.4}$$

These two equations above form a coupled system, and the pressure field does not explicitly appear in neither Eq. (9.3) nor Eq. (9.4) and, in principle, is not needed in the solution.

The system (Eqs. (9.3) and (9.4)) requires boundary conditions on ψ and ω. For the stream function, imposing physically plausible conditions is not difficult. One has to write the proper boundary conditions for the velocity components and use vorticity Eq. (9.2) 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 Eq. (9.4) at the boundary. The process of finding boundary conditions of ω typically results in expressions containing second derivatives, therefore, special numerical treatment is required.

10 Prelude to the algorithm using continuous operators

Consider continuous operators div, curl and grad on bounded domain Ω with boundary $\partial\Omega$. Let vector X, Y and scalar ϕ, ψ fields vanish at boundary $\partial\Omega$. To extend the transpose to the operator grad, we need to have inner products in SF (scalar field) and in VF (vector field) [1]. We use

$$\langle \phi, \psi \rangle = \int_\Omega \phi\psi,$$

for scalar fields ϕ and ψ in Ω, and

$$\langle X, Y \rangle = \int_\Omega X \cdot Y,$$

for vector fields X and Y in Ω. Then the transpose of grad is an operator $\operatorname{grad}^* : VF \to SF$ such that

$$\langle X, \operatorname{grad} \phi \rangle = \langle \phi, \operatorname{grad}^* X \rangle ,$$

for all vector fields X and scalar fields ϕ vanishing on the boundary of Ω. To identify this operator, intergate the identity

$$\nabla \cdot (\phi X) = \phi \nabla \cdot X + X \cdot \nabla\phi,$$

and apply the divergence theorem, to get

$$\int_{\partial\Omega} \phi X \cdot n = \int_\Omega \nabla \cdot (\phi X) = \int_\Omega \phi \nabla \cdot X + \int_\Omega X \cdot \nabla\phi.$$

Since $\phi = 0$ on $\partial\Omega$, we infer

$$\langle \phi, \nabla \cdot X \rangle + \langle X, \nabla\phi \rangle = 0,$$

implying that the transpose of the gradient is the negative divergence, and the transpose of the divergence is the negative gradient:

$$\operatorname{grad}^* = -\operatorname{div} , \qquad \operatorname{div}^* = -\operatorname{grad} . \tag{10.1}$$

This result is dimension-independent, because the divergence theorem holds in all dimensions. To compute the transpose of the 2-dimensional curl, we proceed as

$$\int_\Omega \phi \operatorname{curl} X = \int_\Omega \phi \nabla \cdot X^\perp = -\int_\Omega X^\perp \cdot \nabla\phi = \int_\Omega X \cdot (\nabla\phi)^\perp = \int_\Omega X \cdot \nabla^\perp \phi,$$

yielding

$$\operatorname{curl}^* = \operatorname{grad}^\perp, \quad \left(\operatorname{grad}^\perp\right)^* = \operatorname{curl}.$$

This establishes the first two connections in Diagram (10.2). Moving onto 3-dimensions, Eq. (10.1) already populates most of the diagram

$$SF \underset{\text{-div}}{\overset{\text{grad}}{\rightleftarrows}} VF \underset{\text{curl}}{\overset{\text{curl}}{\rightleftarrows}} VF \underset{\text{-grad}}{\overset{\text{div}}{\leftrightarrows}} SF, \tag{10.2}$$

leaving only the transpose of curl. To this end, recall the identity

$$\nabla \cdot (X \times Y) = (\nabla \times X) \cdot Y - X \cdot (\nabla \times Y).$$

Since we consider only those vector fields vanishing at the boundary $\partial\Omega$, the integration by parts yields

$$0 = \int_{\partial\Omega} (X \times Y) \cdot n = \int_\Omega \nabla \cdot (X \times Y) = \int_\Omega X \cdot \nabla \times Y - \int_\Omega Y \cdot \nabla \times X = \langle X, \operatorname{curl} Y \rangle - \langle X, \operatorname{curl}^* Y \rangle,$$

which identifies the curl as the transpose of itself in 3-dimensions:

$$\operatorname{curl}^* = \operatorname{curl}. \tag{10.3}$$

For 3-dimensional vector fields we can write curl operator in a symmetric matrix form

$$\operatorname{curl} := \begin{bmatrix} 0 & -\frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & -\frac{\partial}{\partial x} \\ -\frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \end{bmatrix} \equiv \begin{bmatrix} 0 & -\frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & -\frac{\partial}{\partial x} \\ -\frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \end{bmatrix}^* =: \operatorname{curl}^*, \tag{10.4}$$

which agrees with Eq. (10.3). When we transpose curl operator, the transpose of each derivative changes it's sign. Thus, we get a symmetric matrix $\operatorname{curl}^* = \operatorname{curl}$, since integrating by parts with respect to variable s of vector fields $X(s), Y(s)$ leads to

$$\langle \frac{d}{ds} X(s), Y(s) \rangle = \int_\Omega \frac{dX}{ds} Y(s) ds = X(s)Y(s)|_{\partial\Omega} - \int_\Omega X(s) \frac{dY}{ds} ds = \int_\Omega X(s) \left(-\frac{d}{ds} Y(s) \right) ds = \langle X(s), -\frac{d}{ds} Y(s) \rangle,$$

where $X(s)Y(s)|_{\partial\Omega}$ vanishes at the boundary $\partial\Omega$.

We can proceed to the algorithm now. To overcome the issue of stating boundary conditions for ω, ψ as discussed in Section 9 we first apply boundary conditions in terms of velocity and pressure and keep them explicit as in Sys. (6.5).

Assume that the solution to Eqs. (1.1) can be decomposed into sum of homogeneous and inhomogeneous velocity vector fields

$$\boldsymbol{v} = \boldsymbol{v}_p + \boldsymbol{v}_h,$$

where $\boldsymbol{v}_h$ satisfies momentum Eq. (1.1a) with homogeneous continuity $\nabla \cdot \boldsymbol{v}_h = 0$, whereas the particular solution satisfies $\nabla \cdot \boldsymbol{v}_p = \hat{\text{bc}}_2$. Identity $\nabla \cdot v_h = 0$ means that the vector field $\boldsymbol{v}_h$ vanishes at the boundary $\partial\Omega$, hence it is permissible use all identities with curl, grad and div derived in this section.

Our goal is to modify the implicit (left) part of Sys. (6.5) by transforming it into vorticity transport Eq. (9.3) together with vorticity stream function substitution Eq. (9.4). We first keep transient, gradient and viscous terms on the left hand side, whereas non-linear terms are linearized to become a vector and moved to the right. Next, we apply curl^* to momentum Eq. (1.1a), which eliminates the pressure gradient ($\operatorname{curl}\operatorname{grad} = 0$), then use $\boldsymbol{v} = \operatorname{curl}\psi$. Since, transient derivative and curl operators are commutative and $\nabla \cdot \boldsymbol{v}_h = 0$, we get

$$\left(\frac{\partial}{\partial t} \right) \operatorname{curl}^* \operatorname{curl} \psi = -\frac{\partial}{\partial t} \nabla^2 \psi, \tag{??}$$

and

$$\operatorname{curl}^* \nabla^2 \operatorname{curl} \psi = -\nabla^4 \psi. \tag{???}$$

Hence, momentum Eq. (1.1a) after the above steps results in

$$\boxed{-\frac{\partial \nabla^2 \psi}{\partial t} + \nabla^4 \psi = \text{RHS}_{\boldsymbol{v},\text{p}},} \tag{10.5}$$

where the right hand side terms were initially stated in terms of $\boldsymbol{v}, p$ and transformed using the same operations as the left hand side of Eq. (10.5).

11 Nullspace method and pressure elimination in terms of discrete operators

The goal of this subsection is to show how unknown pressure variables can be eliminated from Sys. (8.1) similarly to Section 9. Publications of Chang [2] and Hall [5] use the idea that in Sys. (8.1) matrix D is wider than tall for grids larger than 2×2, hence it defines a nullspace. The nullspace of matrix D 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 [2].

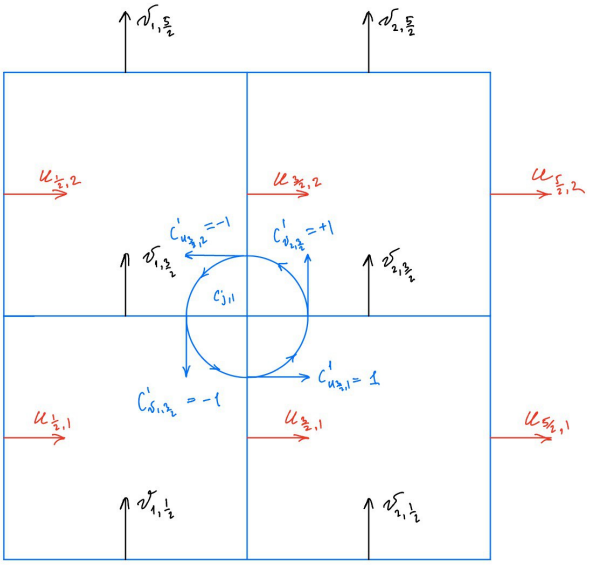


Figure 18: 2×2 example for C matrix.

A more intuitive way of constructing the matrix C relies on the utilization of counterclockwise vorticity around the nodes within the domain (Fig. 18). 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

$$C = \begin{bmatrix} 1 \\ -1 \\ -1 \\ 1 \end{bmatrix}.$$

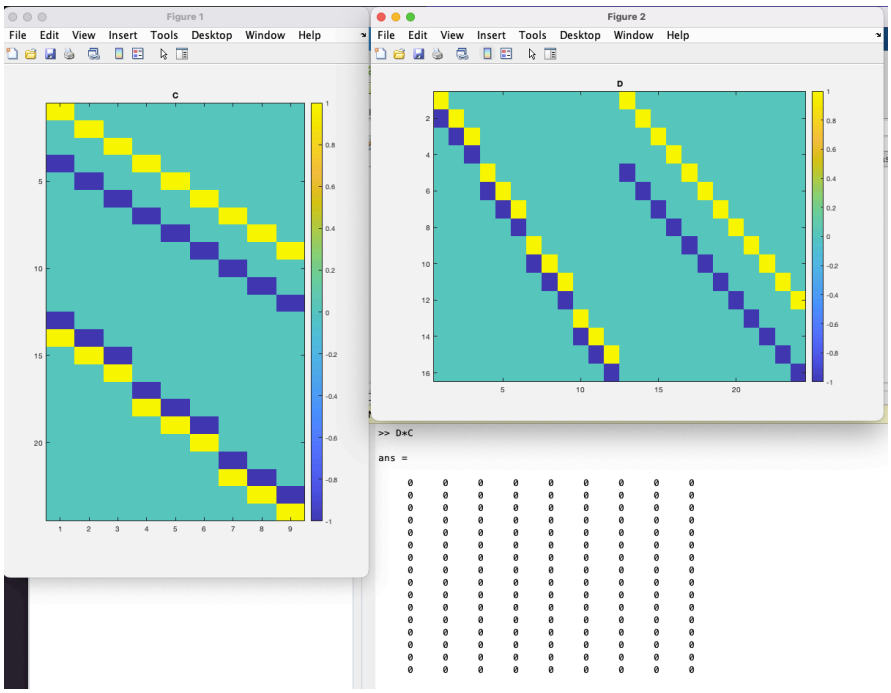


Figure 19: Divergence and Curl matrices.

The matrix C has dimensions corresponding to unknown velocities times the number of nodes around which these velocities revolve. Figure 19 illustrates matrices D and C for a 4×4 grid with an open boundary on the right side of the domain. The desired product then becomes $DC = 0$. $D = -G^T$ (derived in Sections 6.3 and 6.4) leads to important property $(DC)^T = C^T D^T = -C^T G = 0$. Premultiplying momentum equation in Sys. (8.1) by C^T creates $C^T G p = 0$ term, which completely eliminates the pressure from our system.

In order to make use of efficient solvers, the matrix $C^T A$ can be made symmetric if multiplied by C from the right. We use the property of product $C^T C = -L$ being symmetric Laplacian and $C^T L C = -L^2$ being symmetric biharmonic operator as in Eq. (10.5). Hence, the final chord of this method is the introduction of new variable called discrete streamfunction $\psi : q_h = C\psi$, where q_h is a homogeneous solution to the Sys. (8.1) and $q = q_h + q_p$ decomposition is discussed in the following Section 12.

12 Resulting algorithm

Let us consider the solution to discretized Sys. (8.1)

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

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

$$Dq_h^{n+1} = 0, \tag{12.1}$$

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

$$Dq_p^{n+1} = bc_2. \tag{12.2}$$

We assume that momentum equation from Sys. (8.1) is modified to streamfunction Eq. (10.5) with biharmonic operator satisfies homogeneous Eq. (12.1) as in Colonius-Taira [3]. The difference between [3] and the method described below lies in boundary conditions. Colonius and Taira used homogeneous conditions at all boundaries (e.g. $\frac{\partial u}{\partial x} = 0$ at inlet-outlet). Therefore, it was not needed for them to solve an additional non-homogeneous Eq. (12.2), whereas, in our problem statement such equation has to be solved due to non-zero Dirichlet and Neumann boundary conditions.

Taking into account continuous operator identities in Section 10, the resulting algorithm for discrete Navier-Stokes Sys. (8.1) 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 Eq. (12.2) using the method described in next Section 13.
4. Split the solution q^{n+1} into homogeneous and particular, then eliminate the pressure terms in the momentum equation.

$$\begin{aligned}
Aq^{n+1} &= -Gp^{n+1} + bc_1, && \\
A(q_h^{n+1} + q_p^{n+1}) &= D^T p^{n+1} + bc_1, && \text{premultiply by } C^T, \\
C^T A q_h^{n+1} &= C^T(bc_1 - Aq_p^{n+1}), && \text{use } q_h^{n+1} = C\psi^{n+1}, \\
C^T AC\psi^{n+1} &= C^T(bc_1 - Aq_p^{n+1}). &&
\end{aligned} \tag{12.3}$$

   The above Sys. (12.3) is solving Eq. (10.5) using Euler scheme $\frac{\partial \psi}{\partial t} = \frac{\psi^{n+1} - \psi^n}{\Delta t}$.
5. Solve resulting Sys. (12.3) for ψ^{n+1}.
6. Obtain $q_h^{n+1} = C\psi^{n+1}$.
7. Compute $q^{n+1} = q_p^{n+1} + q_h^{n+1}$ and convert it to velocity vector field $\boldsymbol{v}^{n+1} = R^{-1}q^{n+1}$.
8. Transition to the next time instance by repeating Steps (3) to (7) using the new boundary values from q^{n+1}.

Steps (3) and (7) and changing values at the boundary of Step (8) were not necessary in Colonius-Taira [3], whereas Step (2) used homogeneous property $bc_2 = 0$ of continuity Eq. (12.2), making $q^{n+1} = q_h^{n+1}$.

13 Particular solution using Lagrange multipliers

It is possible to find one particular solution to the discrete non-homogeneous continuity Eq. (12.2) using the method of Lagrange multipliers as follows

$$Dq_p^{n+1} = bc_2, \implies \mathcal{L}(q_p, \lambda) = \|q_p\|^2 + \lambda^T (bc_2 - Dq_p). \tag{13.1}$$

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

$$2q_p - D^T\lambda = 0. \tag{13.2}$$

We may premultiply by D to obtain

$$2Dq_p - DD^T\lambda = 0,$$

then the substitution of $bc_2 = Dq_p$ will lead to

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

plugging the above into Eq. (13.2) results in

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

where $D^\dagger = D^T\left(DD^T\right)^{-1}$ 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.

References

[1] MATH 248: Honours Vector Calculus Fall 2019. https://www.math.mcgill.ca/gantumur/math248f19/.

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

[3] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

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

[10] 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.

[11] 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.

[12] 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.

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

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

A Appendix

A.1 Transient schemes