# Discrete stream function method for the incompressible Navier-Stokes equations with simple boundary conditions

Rauan

March 5, 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 simple boundary conditions. We will discuss in detail the scheme formulation, transient and spatial discretizations. Special attention will be paid to the change of unknown variables. After studying these notes one must get a coherent picture of the application of discrete stream function method to incompressible flows with simple Boundary Conditions (BCs) and be able to implement the scheme in code.

## Contents

**1 Introduction** **1**

**2 Discretization** **2**

**3 Nullspace method** **3**

**4 Summary** **5**

**A Appendix** **7**

A.1 Transient schemes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 7

## 1 Introduction

In these notes we will be concerned with the discretization of the Navier-Stokes equations describing the flow of an incompressible fluid past the 2-dimensional cylinder with a singular boundary force f added to the momentum equation as a continuous analog of the immersed boundary formulation:

$$\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{v} \cdot \nabla \boldsymbol{v} = -\nabla p + \epsilon \nabla \cdot \nabla \boldsymbol{v} + \int_s \boldsymbol{f}(\xi(s,t))\delta(\xi - x)ds \tag{1.1a}$$

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

$$\boldsymbol{v}(\xi(s,t)) = \int_x \boldsymbol{v}(x)\delta(x-\xi)dx = \boldsymbol{v}_B(\xi(x,t)), \tag{1.1c}$$

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. The boundary conditions are assumed to be simple, i.e. Dirichlet and Neumann for the normal and tangential velocities, respectively. Spatial variable $x$ represents position in the flow field, $\mathscr{D}$, and $\xi$ denotes coordinates along the immersed boundary, $\partial\mathscr{B}$ having a velocity of $\boldsymbol{v}_B$. The geometry of the immersed object $\mathscr{B}$ is considered to be of arbitrary shape.

# 2 Discretization

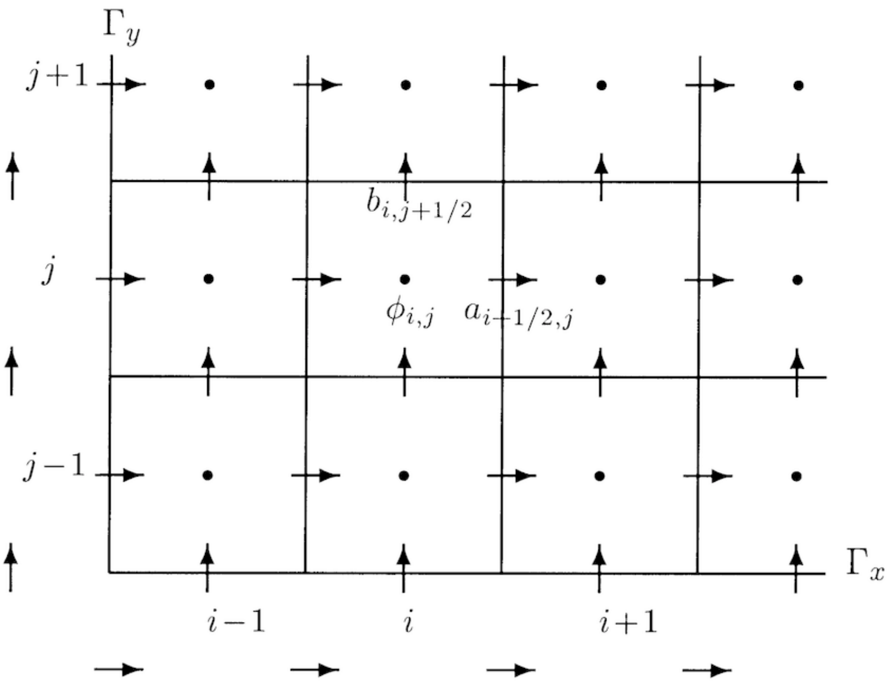


Figure 1: Staggered grid discretization of a two-dimensional computational domain $\mathscr{D}$ and immersed boundary formulation for a body $\mathscr{B}$ depicted by a shaded object. The horizontal and vertical arrows ($\rightarrow$, $\uparrow$) represent the discrete $u_i$ and $v_i$ velocity components, respectively. Pressure $p_i$ is positioned at the center of each cell ($\times$). Lagrangian points, $\xi_k = (\xi_k, \eta_k)$, along $\partial\mathscr{B}$ are shown with filled squares ($\blacksquare$) where boundary forces $\boldsymbol{f}_k = (f_{x,k}, f_{y,k})$ are applied ($\Rightarrow, \Uparrow$).

The above system is discretized with a standard staggered Cartesian grid finite volume method. The mesh and variable locations are depicted in Fig. 1. The computational domain, $\mathscr{D}$, is represented by a Cartesian grid, $(x_i, y_i)$, and the immersed boundary, $\mathscr{B}$ is described by a set of Lagrangian points, $(\xi_k, \eta_k)$, which can be a function of time.

We can rewrite Eqs. (1.1) using discrete differential operators as

$$\begin{aligned} M\frac{dq}{dt} + Gp - Hf &= \mathbf{N}(q) + Lq + bc_1 && \text{(momentum)}, \\ Dq &= 0 + bc_2 && \text{(continuity)}, \\ Eq &= \boldsymbol{v}_B^{n+1} && \text{(no-slip condition)}, \end{aligned} \tag{2.1}$$

where $q = (u\Delta y, v\Delta x)$, $p$ and $f$ are the discrete velocity flux vector, pressure, and boundary force. Discretized non-linear advective term $\boldsymbol{v}\cdot\nabla\boldsymbol{v}$ is denoted as $\mathbf{N}(q)$, whereas operators $M$, $L$ are mass matrix and discrete Laplacian respectively. Operators $G$ and $D$ are discrete gradient and divergence matrices s.t. $D = -G^T$ and [1,4]. The rest operators $E$ and $H$ are the interpolation and regularization operators resulting from the regularization of the Dirac delta functions in momentum and no-slip equations, which are constructed in such a way that $E = -H^T$. No-slip constraint is enforced by equating the boundary velocity, $\boldsymbol{v}_B$, to the velocity value along $\partial\mathscr{B}$ interpolated by $E$ from the neighbouring cells. Regularization operator diffuses the singular boundary force along $\partial\mathscr{B}$ to the Cartesian grid [2].

For computational convenience Eqs. (2.1) can represented as system of linear equations

$$\begin{bmatrix} A & G & -H \\ D & 0 & 0 \\ E & 0 & 0 \end{bmatrix} \begin{pmatrix} q \\ p \\ f \end{pmatrix} = \begin{pmatrix} r^n \\ 0 \\ \boldsymbol{v}_B^{n+1} \end{pmatrix} + \begin{pmatrix} bc_1 \\ bc_2 \\ 0 \end{pmatrix}, \tag{2.2}$$

where $A = \frac{1}{\Delta t}M - \frac{1}{2}L$ comes from implicit treatment using trapezoid method of viscous terms. Explicit term $r^n = \left(\frac{1}{\Delta t}M + \frac{1}{2}L\right)q^n + \frac{3}{2}\mathbb{N}(q^n) - \frac{1}{2}\mathbf{N}(q^{n-1})$ is also obtained from applying Adams-Bashforth scheme to advective term. We can use the properties of operators discussed above to rewrite Sys. (2.2) as

$$\begin{bmatrix} A & G & E^T \\ G^T & 0 & 0 \\ E & 0 & 0 \end{bmatrix} \begin{pmatrix} q \\ p \\ \tilde{f} \end{pmatrix} = \begin{pmatrix} r^n \\ 0 \\ \boldsymbol{v}_B^{n+1} \end{pmatrix} + \begin{pmatrix} bc_1 \\ -bc_2 \\ 0 \end{pmatrix}, \tag{2.3}$$

making the above system of linear algebraic equations symmetric. Here, $\tilde{f}$ is the boundary force with an incorporated scaling factor. This form of the equation is known as the Karush-Kahn-Tucker (KKT) system where $(p, \tilde{f})^T$ appear as a set of Lagrange multiplier to satisfy a set of kinematic constraints. In the discretized set of equations, the constraints are purely numerical and it is no longer necessary to distinguish the pressure and boundary force. Instead we can define a combined variable $\lambda = (p, \tilde{f})^T$ for the Lagrange multipliers and group the submatrices as $Q = [G, E^T]$. Note that by removing the boundary force and no-slip condition along $\partial\mathscr{B}$, the traditional discretization of the incompressible Navier-Stokes equations can be retrieved, i.e. being equivalent to applying divergence to momentum and obtaining pressure-Poisson.

Next, we apply the classical projection (fractional-step) algorithm to Sys. (2.3), which can be expressed as an approximate $LU$ decomposition of the left-hand side matrix, to produce the immersed boundary projection method [2]:

1. $Aq^* = r_1$, (Solve for intermediate velocity).
2. $Q^{\mathrm{T}} A_j^{\dagger} Q\lambda = Q^{\mathrm{T}} q^* - r_2$, (Solve a modified Poisson equation).
3. $q^{n+1} = q^* - A_j^{\dagger} Q\lambda$, (Projection step).

Here $A_j^{\dagger}$ denotes the $j$-th order Taylor series expansion of $A^{-1}$ with respect to $\Delta t$. The explicit terms on the right-hand side have been grouped into $r_1$ and $r_2$. Matrices $A$ and $Q^T A_j^{\dagger} Q$ are constructed to be symmetric positive definite operators in order to use the conjugate-gradient method to efficiently solve for the intermediate velocity and the Lagrange multipliers. All boundary conditions are set to uniform flow $(U_\infty, 0, 0)$ in the streamwise direction ($x$-direction) except for the outflow boundary where a convective boundary condition:

$$\frac{\partial \boldsymbol{v}}{\partial t} + U_\infty \frac{\partial \boldsymbol{v}}{\partial x} = 0, \tag{2.4}$$

where backward first order difference scheme has to be used (without ghost velocity components) to preserve the symmetric property of matrices $A$ and $Q^T A_j^{\dagger} Q$.

# 3 Nullspace method

The nullspace or discrete streamfunction approach [1,3] is a method for solving the Sys. (2.3) without the immersed boundary formulation. In this case, the flow only needs to satisfy the incompressibility constraint, which leads us to the use of discrete streamfunction, $s$, such that

$$q = C\psi, \tag{3.1}$$

where $C$ is a discrete curl operator. This operator is constructed with column vectors corresponding to the basis of the nullspace of divergence matrix $D$.

In this section we will show how unknown pressure variables can be eliminated from Sys. (2.3). Publications of Chang [1] and Hall [3] use the idea that matrix $D$ is wider than tall for grids larger than $2 \times 2$, 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 [1].

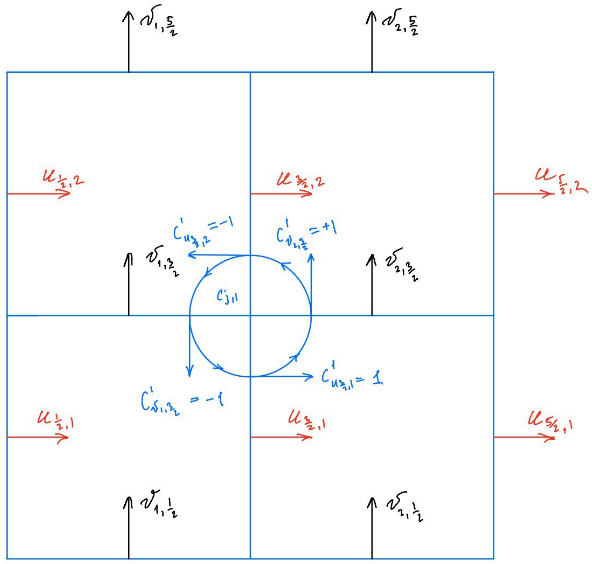


Figure 2: $2 \times 2$ example for $C$ matrix.

A more intuitive way of constructing the matrix $C$ relies on the utilization of counterclockwise vorticity around the nodes within the domain (Fig. 2). 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

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

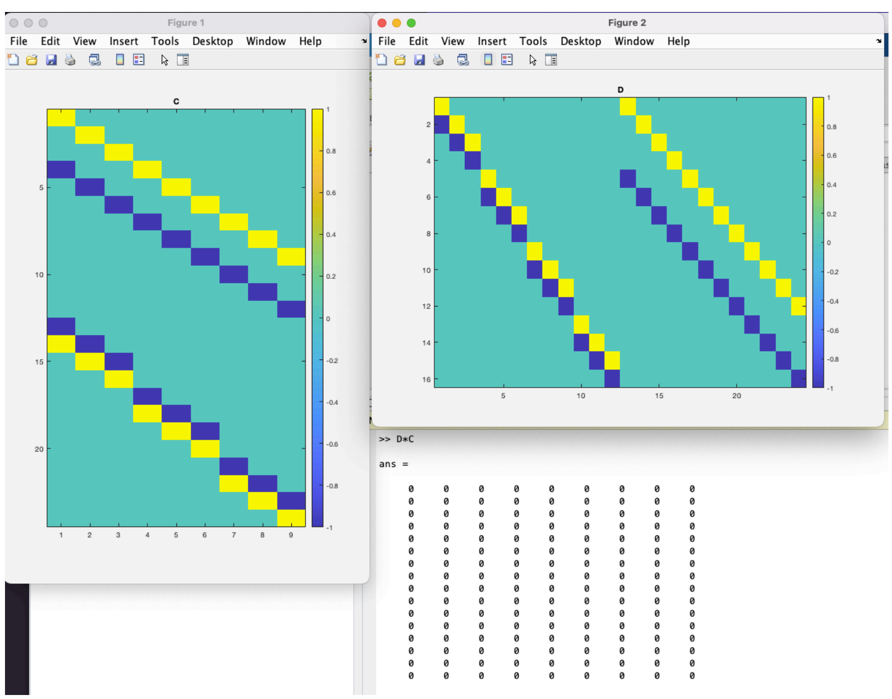


Figure 3: Divergence and Curl matrices.

The matrix $C$ has dimensions corresponding to unknown velocities times the number of nodes around which these velocities revolve. Figure 3 illustrates matrices $D$ and $C$ for a $4 \times 4$ grid with an open boundary on the right side of the domain. The desired product then becomes $DC = 0$. $D = -G^T$ (from Section 2) leads to important property $(DC)^T = C^T D^T = -C^T G = 0$. Premultiplying momentum equation in Sys. (2.3) 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 with Dirichlet boundary conditions. Matrix $C^T C = -L^2$ has zero boundary conditions for the vector it is applied to. Hence, the final chord of this method is the introduction of new variable called discrete streamfunction $\psi : q = C\psi$ as discussed at the beginning of this section.

# 4 Summary

# References


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

[2] 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.

[3] 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.

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

# A Appendix

## A.1 Transient schemes