Theory Manual for CaNS 2.0

This document explains the numeric in CaNS 2.0. The solver uses the projection method on staggered grids.

1. Staggered grid

We show the calculation of convective H and viscous L terms on a staggered grid,

$$\frac{u_{i+1/2,j}^{n+1}-u_{i+1/2,j}}{\delta t}=H_{i+1/2,j}+L_{i+1/2,j} \tag{1}$$

The following scheme can be derived via finite difference/volume. Here, we regard it as a finite volume scheme. Taking 2D problems as an example, the two terms in the x direction,

$$H_{i+1/2,j}=-\frac{(uu)_{i+1,j}-(uu)_{i+1,j}}{\delta x}-\frac{(uv)_{i+1/2,j+1/2}-(uv)_{i+1/2,j-1/2}}{\delta y}-\frac{\varphi_{i+1,j}-\varphi_{i,j}}{\delta x} \tag{2}$$

$$\begin{aligned} L_{i+1/2,j} &= \frac{1}{\mathrm{Re}}\left(\frac{u_{i+3/2,j}-2u_{i+1/2,j}+u_{i-1/2,j}}{\delta x^2}+\frac{u_{i+1/2,j+1}-2u_{i+1/2,j}+u_{i+1/2,j-1}}{\delta y^2}\right) \\ &= \frac{1}{\mathrm{Re}}\left(\frac{1}{\delta x}\left(\frac{\left(u_{i+3/2,j}-u_{i+1/2,j}\right)}{\delta x}-\frac{\left(u_{i+1/2,j}-u_{i-1/2,j}\right)}{\delta x}\right)+\frac{1}{\delta y}\left(\frac{\left(u_{i+1/2,j+1}-u_{i+1/2,j}\right)}{\delta y}-\frac{\left(u_{i+1/2,j}-u_{i+1/2,j-1}\right)}{\delta y}\right)\right) \end{aligned} \tag{3}$$

The two terms in the y direction.

$$H_{i,j+1/2}=-\frac{(uv)_{i+1/2,j+1/2}-(uv)_{i-1/2,j+1/2}}{\delta x}-\frac{(vv)_{i,j+1}-(vv)_{i,j}}{\delta y}-\frac{\varphi_{i,j+1}-\varphi_{i,j}}{\delta y} \tag{4}$$

$$\begin{aligned} L_{i,j+1/2} &= \frac{1}{\mathrm{Re}}\left(\frac{v_{i+1,j+1/2}-2v_{i,j+1/2}+v_{i-1,j+1/2}}{\delta x^2}+\frac{v_{i,j+3/2}-2v_{i,j+1/2}+v_{i,j-1/2}}{\delta y^2}\right) \\ &= \frac{1}{\mathrm{Re}}\left(\frac{1}{\delta x}\left(\frac{\left(v_{i+1,j+1/2}-v_{i,j+1/2}\right)}{\delta x}-\frac{\left(v_{i,j+1/2}-v_{i-1,j+1/2}\right)}{\delta x}\right)+\frac{1}{\delta y}\left(\frac{\left(v_{i,j+3/2}-v_{i,j+1/2}\right)}{\delta y}-\frac{\left(v_{i,j+1/2}-v_{i,j-1/2}\right)}{\delta y}\right)\right) \end{aligned} \tag{5}$$

In the projection method, we also need the divergence of velocity and the gradient of ϕ

$$D_i u_i=\frac{u_{i+1/2,j}-u_{i-1/2,j}}{\Delta x}+\frac{v_{i,j+1/2}-v_{i,j-1/2}}{\Delta y} \tag{6}$$

$$\partial_1\phi = \frac{\phi_{i+1/2,j} - \phi_{i-1/2,j}}{\Delta x}$$
$$\partial_2\phi = \frac{\phi_{i,j+1/2} - \phi_{i,j-1/2}}{\Delta y} \tag{7}$$

Figure 1 shows the stencils for solving u, v and p.

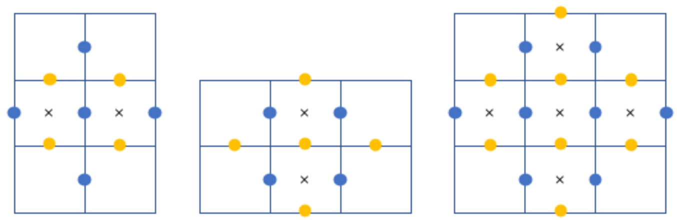
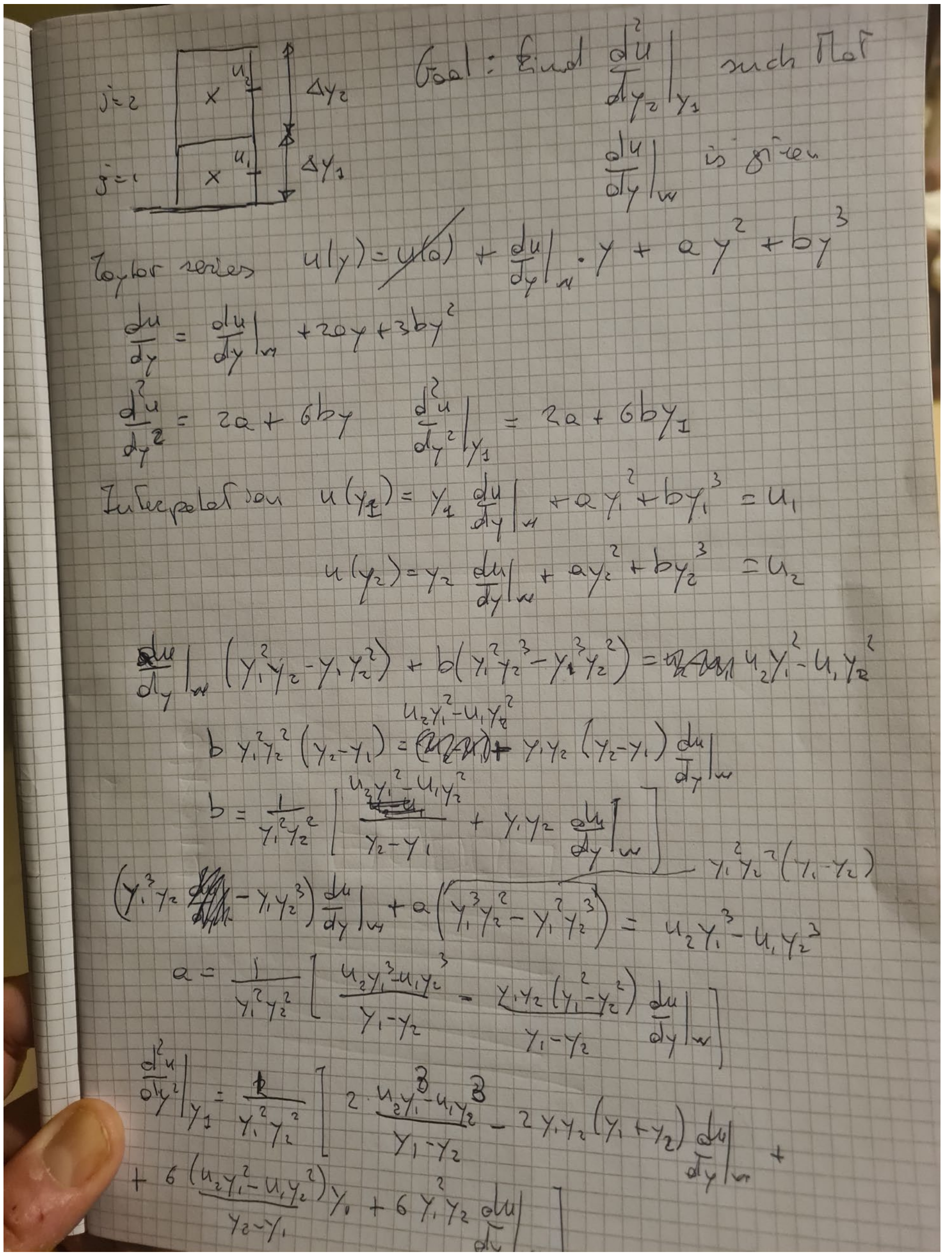


Figure 1 Stencil points for solving u, v and p

We assume a discontinuity between the wall and the first off-wall u location. Across the discontinuity, $\frac{\partial u}{\partial y}$ is large, while $\frac{\partial v}{\partial y}$, $\frac{\partial p}{\partial y}$ and $\frac{\partial \phi}{\partial y}$ are not large. Consequently, we should avoid the interpolation (reconstruction) $(u\varphi)_{i+1/2,j-1/2} = \frac{1}{2}\left((u\varphi)_{i+1/2,j} + (u\varphi)_{i+1/2,j-1}\right)$ or the differencing operator $\left.\frac{\partial(u\varphi)}{\partial y}\right|_{i+1/2,j-1/2} = \frac{1}{\delta y}\left((u\varphi)_{i+1/2,j} - (u\varphi)_{i+1/2,j-1}\right)$, where φ is 1, u or v. Instead, one-sided operation biased towards the domain should be adopted. Table 1 lists the relevant terms in Eq. (2) -(7). The conclusion is that no special treatments are needed near the wall for the projection method on a staggered grid,

Table 1 Terms that contain $(u\varphi)_{i+1/2,j-1/2}$ or $\left.\frac{\partial(u\varphi)}{\partial y}\right|_{i+1/2,j-1/2}$ in Eq. (2)-(7)

Eq.	Related term	Explanation
(2)	$(uv)_{i+1/2,j-1/2}$	$(uv)_{i+1/2,j-1/2} \equiv 0$ due to $v_{i+1/2,j-1/2} \equiv 0$ on the wall
(3)	$\frac{\left(u_{i+1/2,j} - u_{i+1/2,j-1}\right)}{\delta y}$	$\frac{\left(u_{i+1/2,j} - u_{i+1/2,j-1}\right)}{\delta y} \equiv \tau_w$, computed from wall model
(4)-(7)	No	



j=2, j=1, u_2, u_1, Δy_2, Δy_1

Goal: find $\left.\frac{d^2u}{dy^2}\right|_{y_1}$ such that $\left.\frac{du}{dy}\right|_w$ is given

Taylor series $u(y) = \cancel{u(0)} + \left.\frac{du}{dy}\right|_w \cdot y + a y^2 + b y^3$

$\frac{du}{dy} = \left.\frac{du}{dy}\right|_w + 2ay + 3by^2$

$\frac{d^2u}{dy^2} = 2a + 6by \qquad \left.\frac{d^2u}{dy^2}\right|_{y_1} = 2a + 6by_1$

Interpolation $u(y_1) = y_1 \left.\frac{du}{dy}\right|_w + a y_1^2 + b y_1^3 = u_1$

$u(y_2) = y_2 \left.\frac{du}{dy}\right|_w + a y_2^2 + b y_2^3 = u_2$

$\left.\frac{du}{dy}\right|_w (y_1^2 y_2 - y_1 y_2^2) + b(y_1^2 y_2^3 - y_1^3 y_2^2) = u_2 y_1^2 - u_1 y_2^2$

$b\, y_1^2 y_2^2 (y_2 - y_1) = u_2 y_1^2 - u_1 y_2^2 + y_1 y_2 (y_2 - y_1) \left.\frac{du}{dy}\right|_w$

$b = \frac{1}{y_1^2 y_2^2}\left[\frac{u_2 y_1^2 - u_1 y_2^2}{y_2 - y_1} + y_1 y_2 \left.\frac{du}{dy}\right|_w\right]$

$(y_1^3 y_2 - y_1 y_2^3)\left.\frac{du}{dy}\right|_w + a(y_1^3 y_2^2 - y_1^2 y_2^3) = u_2 y_1^3 - u_1 y_2^3$ $\quad y_1^2 y_2^2 (y_1 - y_2)$

$a = \frac{1}{y_1^2 y_2^2}\left[\frac{u_2 y_1^3 - u_1 y_2^3}{y_1 - y_2} - \frac{y_1 y_2 (y_1^2 - y_2^2)}{y_1 - y_2}\left.\frac{du}{dy}\right|_w\right]$

$\left.\frac{d^2u}{dy^2}\right|_{y_1} = \frac{1}{y_1^2 y_2^2}\left[2 \cdot \frac{u_2 y_1^3 - u_1 y_2^3}{y_1 - y_2} - 2 y_1 y_2 (y_1 + y_2)\left.\frac{du}{dy}\right|_w + 6\frac{(u_2 y_1^2 - u_1 y_2^2)}{y_2 - y_1} y_1 + 6 y_1^2 y_2 \left.\frac{du}{dy}\right|_w\right]$

I think there are basically two mistakes in the solution.

(1) Both velocity and its gradient are used at the wall, which makes the problem ill-posed (overdetermined). We should use only the velocity gradient as the wall boundary condition in WMLES; the velocity at the wall is non-zero.

(2) We should avoid differencing operator $\frac{\partial u}{\partial y}$ across the layer between the wall and the first off-wall point. As Larsson reminded, that layer is under-resolved (discontinuous) in WMLES, so the wall-normal derivative computed using the wall point and the first off-wall point is incorrect.

However, this solution does an interpolation using the velocity values at positions 1 and 2 and the velocity gradient at the wall, which implies that it assumes a smooth profile of velocity from the wall to position 1 to position 2. The assumption does not hold true, so this treatment can cause large errors.

In contrast, I still believe my current implementation is correct. It guarantees that the term $\frac{\left(u_{i+1/2,j}-u_{i+1/2,j-1}\right)}{\delta y}=\left.\frac{\partial u}{\partial y}\right|_{\text{model}}$ in Eq. (3). As for using ghost cells, it is just for convenience to guarantee $\frac{\left(u_{i+1/2,j}-u_{i+1/2,j-1}\right)}{\delta y}=\left.\frac{\partial u}{\partial y}\right|_{\text{model}}$, and for consistency with CaNS using ghost points for all kinds of boundary conditions. The size of the ghost cells is not important, as long as $\frac{\left(u_{i+1/2,j}-u_{i+1/2,j-1}\right)}{\delta y}=\left.\frac{\partial u}{\partial y}\right|_{\text{model}}$ can be exactly yielded. In CaNS, the ghost cells have the same size as the first off-wall cells Δy_1, and $u_0=u_1-\left.\frac{\partial u}{\partial y}\right|_{\text{model}}\Delta y_1$. This exactly yields $\frac{u_{i+1/2,j}-u_{i+1/2,j-1}}{\delta y}=\frac{u_1-u_0}{\Delta y_1}=\left.\frac{\partial u}{\partial y}\right|_{\text{model}}$. As for $\frac{\left(u_{i+1/2,j+1}-u_{i+1/2,j}\right)}{\delta y}$, it is just normally computed as $\frac{u_2-u_1}{\delta y}=\frac{u_2-u_1}{\frac{1}{2}\left(\Delta y_1+\Delta y_2\right)}$, using only the velocity information at positions 1 and 2.

Figure 2 shows the wall gradients provided by a wall model. In essence, only four locations need to be assigned values by the wall model, which uses, as input, the information at four interior locations.

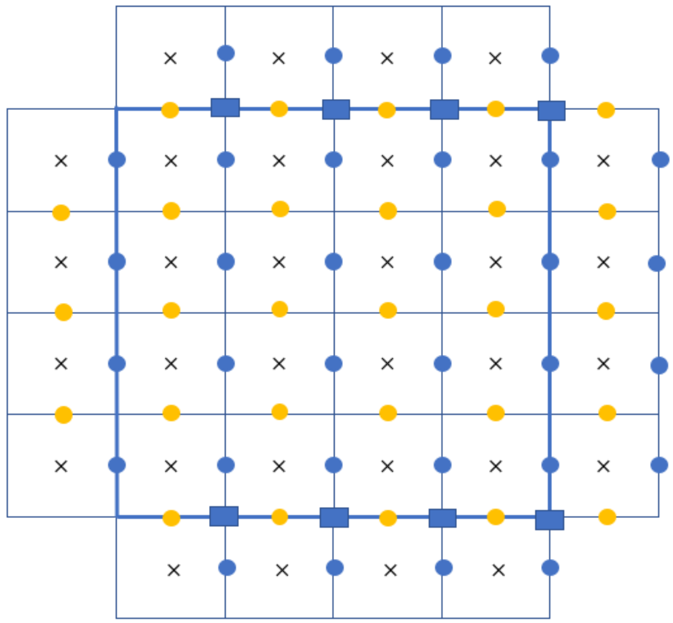


Figure 2 Gradients provided by a wall model

Figure 3 shows a splitting of the domain into two subdomains. The calculation of wall gradients by the wall model only uses, as input, the information at interior points. Hence, it is confirmed that wall model is allowed to be called before updating ghost cell information.

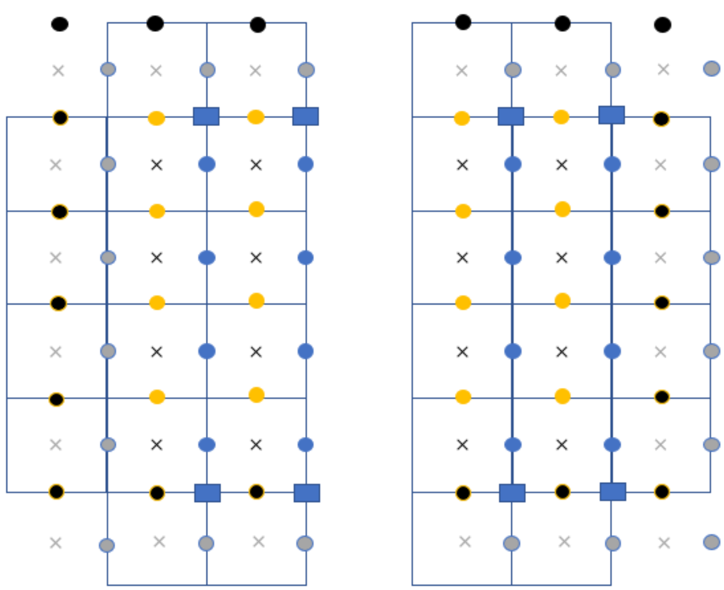


Figure 3 Splitting of domain

2. Poisson equation solver

The elliptic Poisson equation can be solved via iterative methods, like Jacobi, Gauss-Seidel (with successive overrelaxation). ADI can be used, but it also needs iterations for a second-order accuracy in unsteady flows. This is different from AF method, whereby no iteration is involved if accurate A and b in $Ax=b$ (assuming A is a constant matrix). However, there is no factorization of Poission equation. Fourier transform can be exploited for obtaining tridiagonal matrices. For a 3D problem with two periodic (homogeneous) directions, namely x and y, and one non-homogeneous direction, z, it requires two consecutive FFTs in the homogeneous directions. In the following, $\Delta x = L_x / N_x$ and $\Delta y = L_y / N_y$; N_x and N_y denote the number of cells,

$$\frac{\phi_{i-1,j,k} - 2\phi_{i,j,k} + \phi_{i+1,j,k}}{\Delta x^2} + \frac{\phi_{i,j-1,k} - 2\phi_{i,j,k} + \phi_{i,j+1,k}}{\Delta y^2} + \frac{\phi_{i,j,k-1} - 2\phi_{i,j,k} + \phi_{i,j,k+1}}{\Delta z^2} = f_{i,j,k} \tag{8}$$

Fourier series in the x direction,

$$\phi_{i,j,k} = \frac{1}{N_x} \sum_{k_x=0}^{N_x-1} \hat{\phi}_{k_x,j,k} e^{\frac{\imath 2\pi k_x i}{N_x}} \tag{9}$$

Fourier series in the y direction,

$$\hat{\phi}_{k_x,j,k} = \frac{1}{N_y} \sum_{k_y=0}^{N_y-1} \hat{\hat{\phi}}_{k_x,k_y,k} e^{\frac{\imath 2\pi k_y j}{N_y}} \tag{10}$$

Hence,

$$\phi_{i,j,k} = \frac{1}{N_x N_y} \sum_{k_x=0}^{N_x-1} \sum_{k_y=0}^{N_y-1} \hat{\hat{\phi}}_{k_x,k_y,k} e^{\frac{\imath 2\pi k_y j}{N_y}} e^{\frac{\imath 2\pi k_x i}{N_x}} = \frac{1}{N_x N_y} \sum_{k_x=0}^{N_x-1} \sum_{k_y=0}^{N_y-1} \hat{\hat{\phi}}_{k_x,k_y,k} e^{\imath\left(\frac{2\pi k_x i}{N_x} + \frac{2\pi k_y j}{N_y}\right)} \tag{11}$$

Inserting Eq. (11) into Eq. (8),

$$
\begin{aligned}
&\frac{1}{N_x N_y}\sum_{k_x=0}^{N_x-1}\sum_{k_y=0}^{N_y/2-1} e^{\iota\left(\frac{2\pi k_x i}{N_x}+\frac{2\pi k_y j}{N_y}\right)} \frac{2\left(\cos\left(\frac{2\pi k_x}{N_x}\right)-1\right)}{\Delta x^2}\hat{\hat{\phi}}_{k_x,k_y,k} \\
&+\frac{1}{N_x N_y}\sum_{k_x=0}^{N_x-1}\sum_{k_y=0}^{N_y-1} e^{\iota\left(\frac{2\pi k_x i}{N_x}+\frac{2\pi k_y j}{N_y}\right)} \frac{2\left(\cos\left(\frac{2\pi k_y}{N_y}\right)-1\right)}{\Delta y^2}\hat{\hat{\phi}}_{k_x,k_y,k} \\
&+\frac{1}{N_x N_y}\sum_{k_x=0}^{N_x-1}\sum_{k_y=0}^{N_y-1} e^{\iota\left(\frac{2\pi k_x i}{N_x}+\frac{2\pi k_y j}{N_y}\right)} \frac{\hat{\hat{\phi}}_{k_x,k_y,k-1}-2\hat{\hat{\phi}}_{k_x,k_y,k}+\hat{\hat{\phi}}_{k_x,k_y,k+1}}{\Delta z^2} \\
&=\frac{1}{N_x N_y}\sum_{k_x=0}^{N_x-1}\sum_{k_y=0}^{N_y-1} e^{\iota\left(\frac{2\pi k_x i}{N_x}+\frac{2\pi k_y j}{N_y}\right)} \hat{\hat{f}}_{k_x,k_y,k}
\end{aligned}
\tag{12}
$$

A modified wavenumber is defined as

$$
k_x'^2=\frac{2\left(1-\cos\left(\frac{2\pi k_x}{N_x}\right)\right)}{\Delta x^2}=\frac{4}{\Delta x^2}\sin^2\left(\frac{\pi k_x}{N_x}\right)=-\frac{\lambda_x}{\Delta x^2} \tag{13}
$$

$$
k_y'^2=\frac{2\left(1-\cos\left(\frac{2\pi k_y}{N_y}\right)\right)}{\Delta y^2}=\frac{4}{\Delta y^2}\sin^2\left(\frac{\pi k_y}{N_y}\right)=-\frac{\lambda_y}{\Delta y^2} \tag{14}
$$

Strictly speaking, the definition of eigenvalue λ includes Δx^2, but it is left out here for consistency with Pedro's article. Inserting Eq. (13) into Eq. (12),

$$
\left(\frac{\lambda_x}{\Delta x^2}-\frac{\lambda_y}{\Delta y^2}+\delta_z^2\right)\hat{\hat{\phi}}_{k_x,k_y,k}=\hat{\hat{f}}_{k_x,k_y,k} \tag{15}
$$

For a given set of $\left(k_x,k_y\right)$, Eq. (15) is a tridiagonal system of equations; there are in total $N_x \times N_y$ systems of equations. If uniform grid spacing in the z direction,

$$
\left(\frac{\lambda_x}{\Delta x^2}-\frac{\lambda_y}{\Delta y^2}-\frac{\lambda_z}{\Delta z^2}\right)\hat{\hat{\phi}}_{k_x,k_y,k_z}=\hat{\hat{f}}_{k_x,k_y,k_z} \tag{16}
$$

This is not as efficient as solving a tridiagonal system of equations in the z direction. Hence, Eq. (15) is the set of equations solved in channel flow. For square ducts with the Neumann boundaries in the y direction, IDCT-II is performed in that direction. Please refer to Numerical Recipes for the relationship between DFT and DCT. On a staggered grid where f_i is stored at the cell centers from 0 to $N-1$, Eq. (50) and Eq. (51) provide the formulations of DCT-II and IDCT-II. Performing cosine series expansion of in the y direction after Fourier series in the x direction (Eq. (9)),

$$\phi_{i,j,k} = \frac{1}{2N_x N_y} \sum_{k_x=0}^{N_x-1} \left\{ \hat{\hat{\phi}}_{k_x,0,k} + 2\sum_{k_y=1}^{N_y-1} \hat{\hat{\phi}}_{k_x,k_y,k} \cos\left[\frac{\pi}{N_y}\left(j+\frac{1}{2}\right)k_y \right] \right\} e^{\frac{\imath 2\pi k_x i}{N_x}} \tag{17}$$

Inserting Eq. (17) into Eq. (8),

$$\begin{aligned} &\frac{2}{\Delta x^2}\left(\cos\left(\frac{2\pi k_x}{N_x}\right)-1\right)\hat{\hat{\phi}}_{k_x,k_y,k} + \frac{2}{\Delta y^2}\left(\cos\left(\frac{\pi k_y}{N_y}\right)-1\right)\hat{\hat{\phi}}_{k_x,k_y,k} \\ &+\frac{1}{\Delta z^2}\left(\hat{\hat{\phi}}_{k_x,k_y,k-1} - 2\hat{\hat{\phi}}_{k_x,k_y,k} + \hat{\hat{\phi}}_{k_x,k_y,k+1}\right) - \hat{\hat{f}}_{k_x,k_y,k} = 0 \end{aligned} \tag{18}$$

Modified wavenumber is defined as

$$k_x'^2 = \frac{2\left(1-\cos\left(\frac{2\pi k_x}{N}\right)\right)}{\Delta x^2} = \frac{4}{\Delta x^2}\sin^2\left(\frac{\pi k_x}{N_x}\right) = -\frac{\lambda_x}{\Delta x^2} \tag{19}$$

$$k_y'^2 = \frac{2\left(1-\cos\left(\frac{\pi k_y}{N_y}\right)\right)}{\Delta y^2} = \frac{4}{\Delta y^2}\sin^2\left(\frac{\pi k_y}{2N_y}\right) = -\frac{\lambda_y}{\Delta y^2} \tag{20}$$

Hence,

$$\left(\frac{\lambda_x}{\Delta x^2} - \frac{\lambda_y}{\Delta y^2} + \delta_z^2\right)\hat{\hat{\phi}}_{k_x,k_y,k} = \hat{\hat{f}}_{k_x,k_y,k} \tag{21}$$

It has the same form as Eq. (15), which includes $N_x \times N_y$ equations. Note that the index of input data points is always $0,1,\ldots,N-1$, and N is the number of those input points.

3. Low-storage Runge-Kutta scheme

We use symbols in Orlandi's book. In CaNS, the low-storage Runge-Kutta scheme is combined with the projection method (Harlow and Welch, 1965; Chorin, 1968; Kim and Moin, 1985)

$$
\begin{aligned}
&\hat{u}_i^k = u_i^{k-1} + \Delta t\left(\gamma^k\left(H_i^{k-1} + \frac{1}{\mathrm{Re}} L_{jj} u_i^{k-1}\right) + \rho^k\left(H_i^{k-2} + \frac{1}{\mathrm{Re}} L_{jj} u_i^{k-2}\right) - \alpha^k \partial_i p^{k-1}\right)\\
&L_{jj}\phi = \frac{D_i \hat{u}_i^k}{\gamma^k \Delta t}\\
&u_i^k = \hat{u}_i^k - \gamma^k \Delta t \partial_i \phi\\
&p^k = p^{k-1} + \phi
\end{aligned}
\tag{22}
$$

where $k = 1,2,3$, $u_i^0 = u_i^n$ and $u_i^3 = u_i^{n+1}$. The variable u_i^{-1} is not needed due to $\rho_1 = 0$. The operator H_i denotes the convection term, and $L_{ij} = \dfrac{\partial^2}{\partial x_i \partial x_j}$ is second-order derivative. The coefficients

$$
\begin{aligned}
&\gamma_1 = \frac{8}{15}, \gamma_2 = \frac{5}{12}, \gamma_3 = \frac{3}{4}\\
&\rho_1 = 0, \rho_2 = -\frac{17}{60}, \rho_3 = -\frac{5}{12}\\
&\alpha_1 = \frac{8}{15}, \alpha_2 = \frac{2}{15}, \alpha_3 = \frac{1}{3}
\end{aligned}
\tag{23}
$$

If the viscous term is treated implicitly in all the three directions,

$$
\hat{u}_i^k = u_i^{k-1} + \Delta t\left(\gamma^k H_i^{k-1} + \rho^k H_i^{k-2} + \frac{\alpha^k}{2\,\mathrm{Re}} L_{jj}\left(\hat{u}_i^k + u_i^{k-1}\right) - \alpha^k \partial_i p^{k-1}\right) \tag{24}
$$

In a computer code, $\Delta u_i^k = \hat{u}_i^k - u_i^{k-1}$,

$$
\Delta u_i^k - \frac{\Delta t \alpha^k}{2\,\mathrm{Re}} L_{jj} \Delta u_i^k = \Delta t\left(\gamma^k H_i^{k-1} + \rho^k H_i^{k-2} + \frac{\alpha^k}{\mathrm{Re}} L_{jj} u_i^{k-1} - \alpha^k \partial_i p^{k-1}\right) = R_i \tag{25}
$$

The right-hand side of Eq. (25) is similar to that of Eq. (22). Let $\beta^k = \dfrac{\Delta t \alpha^k}{2\,\mathrm{Re}}$,

$$
\left(1 - \beta^k L_{jj}\right)\Delta u_i^k = \Delta t\left(\gamma^k H_i^{k-1} + \rho^k H_i^{k-2} - \alpha^k \partial_i p^{k-1}\right) + 2\beta^k L_{jj} u_i^{k-1} = R_i \tag{26}
$$

That is,

$$
\left(1 - \beta^k\left(L_{11} + L_{22} + L_{33}\right)\right)\Delta u_i^k = R_i \tag{27}
$$

If L_{ij} is discretized via second-order central differencing, the left-hand side of Eq. (27) is a heptadiagonal matrix. In 2D problems, it is a pentadiagonal matrix (see Apeendix A). The modified Helmholtz equation Eq. (27) can be solved via approximate factorization (AF) or transform method. In CaNS, the transform method is used.

If the viscous term is treated implicitly only in the z direction,

$$\hat{u}_i^k = u_i^{k-1} + \Delta t \left(\begin{array}{l} \gamma^k \left(H_i^{k-1} + \frac{1}{\mathrm{Re}} L_{11} u_i^{k-1} + \frac{1}{\mathrm{Re}} L_{22} u_i^{k-1} \right) + \rho^k \left(H_i^{k-2} + \frac{1}{\mathrm{Re}} L_{11} u_i^{k-2} + \frac{1}{\mathrm{Re}} L_{22} u_i^{k-2} \right) \\ + \frac{\alpha^k}{2\,\mathrm{Re}} L_{33} \left(\hat{u}_i^k + u_i^{k-1} \right) - \alpha^k \partial_i p^{k-1} \end{array} \right) \tag{28}$$

Re-expressed using $\Delta u_i^k = \hat{u}_i^k - u_i^{k-1}$,

$$\Delta u_i^k - \frac{\Delta t \alpha^k}{2\,\mathrm{Re}} L_{33} \Delta u_i^k = \Delta t \left(\begin{array}{l} \gamma^k \left(H_i^{k-1} + \frac{1}{\mathrm{Re}} L_{11} u_i^{k-1} + \frac{1}{\mathrm{Re}} L_{22} u_i^{k-1} \right) + \rho^k \left(H_i^{k-2} + \frac{1}{\mathrm{Re}} L_{11} u_i^{k-2} + \frac{1}{\mathrm{Re}} L_{22} u_i^{k-2} \right) \\ + \frac{\alpha^k}{\mathrm{Re}} L_{33} u_i^{k-1} - \alpha^k \partial_i p^{k-1} \end{array} \right) = R_i \tag{29}$$

Let $\beta^k = \frac{\Delta t \alpha^k}{2\,\mathrm{Re}}$,

$$\Delta u_i^k - \beta^k L_{33} \Delta u_i^k = \Delta t \left(\begin{array}{l} \gamma^k \left(H_i^{k-1} + \frac{1}{\mathrm{Re}} L_{11} u_i^{k-1} + \frac{1}{\mathrm{Re}} L_{22} u_i^{k-1} \right) \\ + \rho^k \left(H_i^{k-2} + \frac{1}{\mathrm{Re}} L_{11} u_i^{k-2} + \frac{1}{\mathrm{Re}} L_{22} u_i^{k-2} \right) - \alpha^k \partial_i p^{k-1} \end{array} \right) + 2\beta^k L_{33} u_i^{k-1} = R_i \tag{30}$$

That is,

$$\left(1 - \beta^k L_{33} \right) \Delta u_i^k = R_i \tag{31}$$

The left-hand side of Eq. (31) is a tridiagonal matrix. Note that R_i has a different form in Eq. (31) from that in Eq. (27).

4. Approximate factorization

Equation (27) with 3D implicit viscous diffusion can be solved via approximate factorization. Introducing matrix symbols (not tensors) A_{i1}, A_{i2} and A_{i3},

$$\left(1 - A_{i1} - A_{i2} - A_{i3} \right) \Delta u_i^k = R_i \tag{32}$$

Note that each of A_{i1}, A_{i2} and A_{i3} is a tridiagonal matrix. Using approximate factorization (AF),

$$\left(1 - A_{i1} \right) \left(1 - A_{i2} \right) \left(1 - A_{i3} \right) \Delta u_i^k = R_i \tag{33}$$

Re-expressed as

$$\begin{aligned}
\left(1-A_{i1}\right)\Delta u_i^{**} &= R_i \\
\left(1-A_{i2}\right)\Delta u_i^{*} &= \Delta u_i^{**} \\
\left(1-A_{i3}\right)\Delta u_i^{k} &= \Delta u_i^{*}
\end{aligned} \tag{34}$$

The sweep procedure is described in Anderson's book (P245). In the i direction, $\left(i,j,k\right)=\left(1:n,1,1\right)$ is first solved, then $\left(i,j,k\right)=\left(1:n,2,1\right)$, etc. In summary, hybrid Runge-Kutta/Crank-Nicolson finally solves tridiagonal systems of equations, and there are no constraints on the kinds of boundary conditions. The three-step explicit Runge-Kutta helps improve the accuracy, and the implicit Crank-Nicolson helps increase Δt.

5. Transform method

Equation (27) with 3D implicit viscous diffusion can be solved via transform method,

$$\left(1-\beta\left(\frac{\lambda_x}{\Delta x^2}-\frac{\lambda_y}{\Delta y^2}+\delta_z^2\right)\right)\Delta\hat{\hat{u}}_{k_x,k_y,k}=\hat{\hat{R}}_{k_x,k_y,k} \tag{35}$$

The formulations of the eigenvalues are given in Table 6 and Table 7 in Appendix C. Note that both staggered and non-staggered transforms are used. In a square duct, u and v use staggered and non-staggered transforms in the y direction, respectively.

In CaNS, transform method (non-staggered) is also used in the x and y direction when the diffusion term is treated implicitly in all the directions. Table 2 lists the theoretical limitations of CaNS, and Table 3 lists the actual limitations. The same constraints apply to both the implicit and implicit 1D scheme. Consequently, the wall modeling capability always applies in the z direction, and only in explicit (or implicit 1D) form, applies to x and y directions. Also note that homogenous Neumann boundary condition always applies for walls with/without wall models, due to the no-penetration wall-normal boundary condition.

Table 2 Limitations of time-advancement schemes

	Explicit RK	Hybrid RK/CN	Hybrid RK/CN (1D, z)
Mesh-x	Uniform	Uniform	Uniform
Mesh-y	Uniform	Uniform	Uniform
Mesh-z	Non-uniform	Non-uniform	Non-uniform

cbcvel-x	PP, DD, NN, DN	PP, D0D0, N0N0, D0N0	PP, DD, NN, DN
cbcvel-y	PP, DD, NN, DN	PP, D0D0, N0N0, D0N0	PP, DD, NN, DN
cbcvel-z	PP, DD, NN, DN	PP, DD, NN, DN	PP, DD, NN, DN

Table 3 Limitations of CaNS

	Explicit RK	Hybrid RK/CN	Hybrid RK/CN (1D, z)
Mesh-x	Uniform	Uniform	Uniform
Mesh-y	Uniform	Uniform	Uniform
Mesh-z	Non-uniform	Non-uniform	Non-uniform
bcvel-x	PP, DD, NN, DN	PP, D0D0	PP, DD, NN, DN (PP, D0D0)
bcvel-y	PP, DD, NN, DN	PP, D0D0	PP, DD, NN, DN (PP, D0D0)
bcvel-z	PP, DD, NN, DN	PP, DD, NN, DN	PP, DD, NN, DN

6. Wall model

The nondimensionalized log law reads as

$$\frac{U}{u_\tau} = \frac{1}{\kappa} \ln\left(y u_\tau \operatorname{Re}_b \right) + B \tag{36}$$

We use Newton-Raphson to solve u_τ in Eq. (36). We construct

$$f\left(u_\tau\right) = \frac{U}{u_\tau} - \frac{1}{\kappa} \ln\left(y u_\tau \operatorname{Re}_b \right) - B \tag{37}$$

$$f'\left(u_\tau\right) = -\frac{1}{u_\tau}\left(\frac{U}{u_\tau} + \frac{1}{\kappa} \right) \tag{38}$$

The Newton-Raphson iteration formula is

$$u_{\tau,k+1} = u_{\tau,k} - \frac{f\left(u_\tau\right)}{f'\left(u_\tau\right)} \tag{39}$$

7. Coupling of wall model to LES

The current coupling is though modifying the ghost cell wall-parallel velocity values to match the wall stress yielded from a wall model. No-penetration boundary condition is applied on the walls, so zero convective flux is guaranteed; this implementation strategy ensures no convective flux on the walls. Of course, there is slip velocity on the wall.

8. Verification and validation

Verification is to demonstrate that a wall model has been correctly implemented into CaNS. It is done via comparison to other codes or analytical solutions (if existing). Validation is to demonstrate whether a model is good or not, in terms of representing/modeling the true physics. It is done via comparing to experiments, or analytical solutions (if existing). Here, verification is needed to show if the implementation of wall model is correct.

Table 4 Selected verification cases

	Re_τ	Time advancement
Laminar channel2D		Explicit
Laminar channel2D		Implicit 1D
Laminar channel3D		Explicit
Laminar channel3D		Implicit 1D
Turbulent channel2D	180-10^10	Explicit
Turbulent channel2D	180-10^10	Implicit 1D
Turbulence channel3D	180-10^10	Explicit
Turbulence channel3D	180-10^10	Implicit 1D

Table 5 Selected validation cases

	Re	Time advancement
Square duct		Explicit

Appendix A

For an interior u, v and p control volumes, we have

$$\begin{aligned}\frac{u_{i+1/2,j}^{n+1}-u_{i+1/2,j}}{\delta t}=&-\frac{\left(u_{i+1,j}\right)^2-\left(u_{i,j}\right)^2}{\delta x}-\frac{\left(u_{i+1/2,j+1/2}\right)\left(v_{i+1/2,j+1/2}\right)-\left(u_{i+1/2,j-1/2}\right)\left(v_{i+1/2,j-1/2}\right)}{\delta y}\\&+\nu\left(\frac{u_{i+3/2,j}-2u_{i+1/2,j}+u_{i-1/2,j}}{\delta x^2}+\frac{u_{i+1/2,j+1}-2u_{i+1/2,j}+u_{i+1/2,j-1}}{\delta y^2}\right)-\frac{\varphi_{i+1,j}-\varphi_{i,j}}{\delta x}\end{aligned}\tag{40}$$

$$\begin{aligned}\frac{v_{i,j+1/2}^{n+1}-v_{i,j+1/2}}{\delta t}=&-\frac{\left(u_{i+1/2,j+1/2}\right)\left(v_{i+1/2,j+1/2}\right)-\left(u_{i-1/2,j+1/2}\right)\left(v_{i-1/2,j+1/2}\right)}{\delta x}-\frac{\left(v_{i,j+1}\right)^2-\left(v_{i,j}\right)^2}{\delta y}\\&+\nu\left(\frac{v_{i+1,j+1/2}-2v_{i,j+1/2}+v_{i-1,j+1/2}}{\delta x^2}+\frac{v_{i,j+3/2}-2v_{i,j+1/2}+v_{i,j-1/2}}{\delta y^2}\right)-\frac{\varphi_{i,j+1}-\varphi_{i,j}}{\delta y}\end{aligned}\tag{41}$$

In the following, we derive the Poisson equation of pressure. We define

$$D_{i,j}=\frac{u_{i+1/2,j}-u_{i-1/2,j}}{\delta x}+\frac{v_{i,j+1/2}-v_{i,j-1/2}}{\delta y}\tag{42}$$

We have

$$\begin{aligned}\frac{D_{i,j}^{n+1}-D_{i,j}}{\delta t}=&-Q_{i,j}-\frac{\varphi_{i+1,j}-2\varphi_{i,j}+\varphi_{i-1,j}}{\delta x^2}-\frac{\varphi_{i,j+1}-2\varphi_{i,j}+\varphi_{i,j-1}}{\delta y^2}\\&+\nu\left(\frac{D_{i+1,j}-2D_{i,j}+D_{i-1,j}}{\delta x^2}+\frac{D_{i,j+1}-2D_{i,j}+D_{i,j-1}}{\delta y^2}\right)\end{aligned}\tag{43}$$

where

$$\begin{aligned}Q_{i,j}=&\frac{\left(u_{i+1,j}\right)^2-2\left(u_{i,j}\right)^2+\left(u_{i-1,j}\right)^2}{\delta x^2}\\&+\frac{\left(v_{i,j+1}\right)^2-2\left(v_{i,j}\right)^2+\left(v_{i,j-1}\right)^2}{\delta y^2}\\&+\frac{2}{\delta x\delta y}\left\{\begin{array}{l}\left(u_{i+1/2,j+1/2}\right)\left(v_{i+1/2,j+1/2}\right)-\left(u_{i+1/2,j-1/2}\right)\left(v_{i+1/2,j-1/2}\right)\\-\left(\left(u_{i-1/2,j+1/2}\right)\left(v_{i-1/2,j+1/2}\right)\right)-\left(u_{i-1/2,j-1/2}\right)\left(v_{i-1/2,j-1/2}\right)\Big)\end{array}\right\}\end{aligned}\tag{44}$$

We should have $D_{i,j}^{n+1}=0$ for every cell at the end of the time cycle. Hence,

$$\frac{\varphi_{i+1,j}-2\varphi_{i,j}+\varphi_{i-1,j}}{\delta x^2}+\frac{\varphi_{i,j+1}-2\varphi_{i,j}+\varphi_{i,j-1}}{\delta y^2}=-R_{i,j}\tag{45}$$

$$R_{i,j}=Q_{i,j}-\frac{D_{i,j}}{\delta t}-\nu\left(\frac{D_{i+1,j}-2D_{i,j}+D_{i-1,j}}{\delta x^2}+\frac{D_{i,j+1}-2D_{i,j}+D_{i,j-1}}{\delta y^2}\right)\tag{46}$$

Eq. (40), (41), (45), (46) are the set of equations used for solving the Navier-Stokes equations. When there is variable eddy viscosity, the viscous term is

$$\begin{aligned}\mathrm{VD}_u = &\frac{1}{\delta x_c}\left[\left(\nu+\nu_t\right)_{i+1,j}\frac{u_{i+3/2,j}-u_{i+1/2,j}}{\delta x_f}-\left(\nu+\nu_t\right)_{i,j}\frac{u_{i+1/2,j}-u_{i-1/2,j}}{\delta x_f}\right]\\ &+\frac{1}{\delta y_f}\left[\left(\nu+\nu_t\right)_{i+1/2,j+1/2}\frac{u_{i+1/2,j+1}-u_{i+1/2,j}}{\delta y_c}-\left(\nu+\nu_t\right)_{i+1/2,j-1/2}\frac{u_{i+1/2,j}-u_{i+1/2,j-1}}{\delta y_c}\right]\end{aligned} \tag{47}$$

$$\begin{aligned}\mathrm{VD}_v = &\frac{1}{\delta x_f}\left[\left(\nu+\nu_t\right)_{i+1/2,j+1/2}\frac{v_{i+1,j+1/2}-v_{i,j+1/2}}{\delta x_c}-\left(\nu+\nu_t\right)_{i-1/2,j+1/2}\frac{v_{i,j+1/2}-v_{i-1,j+1/2}}{\delta x_c}\right]\\ &+\frac{1}{\delta y_c}\left[\left(\nu+\nu_t\right)_{i,j+1}\frac{v_{i,j+3/2}-v_{i,j+1/2}}{\delta y_f}-\left(\nu+\nu_t\right)_{i,j}\frac{v_{i,j+1/2}-v_{i,j-1/2}}{\delta y_f}\right]\end{aligned} \tag{48}$$

Appendix B

Pentadiagonal matrix yielded from Eq. (27) in 2D problems.

$$\left[\begin{array}{ccccccccccccccccccccccccc}
 & & & & & & & & & & & & & & & & & & & & & & & & \\
 & -b & & & & -a & c & -a & & & & -b & & & & & & & & & & & & & \\
 & & -b & & & & -a & c & -a & & & & -b & & & & & & & & & & & & \\
 & & & -b & & & & -a & c & -a & & & & -b & & & & & & & & & & & \\
 & & & & & & & & & & & & & & & & & & & & & & & & \\
 & & & & & & & & & & & & & & & & & & & & & & & & \\
 & & & & & & -b & & & & -a & c & -a & & & & -b & & & & & & & & \\
 & & & & & & & -b & & & & -a & c & -a & & & & -b & & & & & & & \\
 & & & & & & & & -b & & & & -a & c & -a & & & & -b & & & & & & \\
 & & & & & & & & & & & & & & & & & & & & & & & & \\
 & & & & & & & & & & & & & & & & & & & & & & & & \\
 & & & & & & & & & & & -b & & & & -a & c & -a & & & & -b & & & \\
 & & & & & & & & & & & & -b & & & & -a & c & -a & & & & -b & & \\
 & & & & & & & & & & & & & -b & & & & -a & c & -a & & & & -b &
\end{array}\right]
\begin{bmatrix}\Delta u_{0,0}=0\\ \Delta u_{1,0}=0\\ \Delta u_{2,0}=0\\ \Delta u_{3,0}=0\\ \Delta u_{4,0}=0\\ \Delta u_{0,1}=0\\ \Delta u_{1,1}\\ \Delta u_{2,1}\\ \Delta u_{3,1}\\ \Delta u_{4,1}=0\\ \Delta u_{0,2}=0\\ \Delta u_{1,2}\\ \Delta u_{2,2}\\ \Delta u_{3,2}\\ \Delta u_{4,2}=0\\ \Delta u_{0,3}=0\\ \Delta u_{1,3}\\ \Delta u_{2,3}\\ \Delta u_{3,3}\\ \Delta u_{4,3}=0\\ \Delta u_{0,4}=0\\ \Delta u_{1,4}=0\\ \Delta u_{2,4}=0\\ \Delta u_{3,4}=0\\ \Delta u_{4,4}=0\end{bmatrix}
=
\begin{bmatrix}R_{0,0}\\ R_{1,0}\\ R_{2,0}\\ R_{3,0}\\ R_{4,0}\\ R_{0,1}\\ R_{1,1}\\ R_{2,1}\\ R_{3,1}\\ R_{4,1}\\ R_{0,2}\\ R_{1,2}\\ R_{2,2}\\ R_{3,2}\\ R_{4,2}\\ R_{0,3}\\ R_{1,3}\\ R_{2,3}\\ R_{3,3}\\ R_{4,3}\\ R_{0,4}\\ R_{1,4}\\ R_{2,4}\\ R_{3,4}\\ R_{4,4}\end{bmatrix}$$

Appendix C

The eigenvalues are given by $\lambda_k = -4\sin^2(\theta_k)$. Table 6 and Table 7 list the transforms and inverse transforms for different combinations of boundary conditions. N is the number of input points, $0,1,\ldots,N-1$, instead of the number of cells. Zero-value points are always left out. The current index of input points is consistent with scipy and wiki. The factors in the inverse transforms are consistent with Pedro's paper and scipy.

Table 6 Transforms on a staggered grid

BC	θ_k	Transform	Inverse transform
PP		DFT	IDFT
NN	$\frac{\pi k}{2N}$	DCT-II	$\frac{1}{2\mathrm{N}}$DCT-III
DD	$\frac{\pi(k+1)}{2N}$	DST-II	$\frac{1}{2N}$DST-III
ND	$\frac{\pi(2k+1)}{4N}$	DCT-IV	$\frac{1}{2N}$DCT-IV
DN		DST-IV	$\frac{1}{2N}$DST-IV

Table 7 Transforms on a non-staggered grid

BC	θ_q	Transform	Inverse transform
PP		DFT	IDFT
NN	$\frac{\pi k}{2(N-1)}$	DCT-I	$\frac{1}{2(N-1)}$DCT-I
DD	$\frac{\pi(k+1)}{2(N+1)}$	DST-I	$\frac{1}{2(N+1)}$DST-I
ND	$\frac{\pi(2k+1)}{4N}$	DCT-III	$\frac{1}{2N}$DCT-II
DN		DST-III	$\frac{1}{2N}$DST-II

We list the transforms and inverse transforms for the above 8 types. The formulations below have been checked using Python.

1) staggered NN

$$X_k = 2\sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N} k\left(n+\frac{1}{2}\right)\right], k = 0,...,N-1 \tag{49}$$

$$x_n = \frac{1}{2N}\left\{X_0 + 2\sum_{k=1}^{N-1} X_k \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right]\right\}, n = 0,...,N-1 \tag{50}$$

2) staggered DD

$$X_k = 2\sum_{n=0}^{N-1} x_n \sin\left[\frac{\pi}{N}(k+1)\left(n+\frac{1}{2}\right)\right], k = 0,...,N-1 \tag{51}$$

$$x_n = \frac{1}{2N}\left\{(-1)^n X_{N-1} + 2\sum_{k=0}^{N-2} X_k \sin\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)(k+1)\right]\right\}, n = 0,...,N-1 \tag{52}$$

3) staggered ND

$$X_k = 2\sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(k+\frac{1}{2}\right)\left(n+\frac{1}{2}\right)\right], k = 0,...,N-1 \tag{53}$$

$$x_n = \frac{1}{2N}\left\{2\sum_{k=0}^{N-1} X_k \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right]\right\}, n = 0,...,N-1 \tag{54}$$

4) staggered DN

$$X_k = 2\sum_{n=0}^{N-1} x_n \sin\left[\frac{\pi}{N}\left(k+\frac{1}{2}\right)\left(n+\frac{1}{2}\right)\right], k = 0,...,N-1 \tag{55}$$

$$x_n = \frac{1}{2N}\left\{2\sum_{k=0}^{N-1} X_k \sin\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right]\right\}, n = 0,...,N-1 \tag{56}$$

5) Non-staggered NN

$$X_k = x_0 + (-1)^k x_{N-1} + 2\sum_{n=1}^{N-2} x_n \cos\left(\frac{\pi}{N-1} kn\right), k=0,...,N-1 \tag{57}$$

$$x_n = \frac{1}{2(N-1)}\left\{\left(X_0 + (-1)^n X_{N-1}\right) + 2\sum_{k=1}^{N-2} X_k \cos\left(\frac{\pi}{N-1} nk\right)\right\}, n = 0,...,N-1 \tag{58}$$

6) Non-staggered DD

$$X_k = 2\sum_{n=0}^{N-1} x_n \sin\left[\frac{\pi}{N+1}(k+1)(n+1)\right], k = 0,\ldots,N-1 \tag{59}$$

$$x_n = \frac{1}{2(N+1)}\left\{2\sum_{k=0}^{N-1} X_k \sin\left[\frac{\pi}{N+1}(n+1)(k+1)\right]\right\}, n = 0,\ldots,N-1 \tag{60}$$

7) Non-staggered ND

$$X_k = x_0 + 2\sum_{n=1}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(k+\frac{1}{2}\right)n\right], k = 0,\ldots,N-1 \tag{61}$$

$$x_n = \frac{1}{2N}\left\{2\sum_{k=0}^{N-1} X_k \cos\left[\frac{\pi}{N}n\left(k+\frac{1}{2}\right)\right]\right\}, n = 0,\ldots,N-1 \tag{62}$$

8) Non-staggered DN

$$X_k = (-1)^k x_{N-1} + 2\sum_{n=0}^{N-2} x_n \sin\left[\frac{\pi}{N}\left(k+\frac{1}{2}\right)(n+1)\right], k = 0,\ldots,N-1 \tag{63}$$

$$x_n = \frac{1}{2N}\left\{2\sum_{k=0}^{N-1} X_k \sin\left[\frac{\pi}{N}(n+1)\left(k+\frac{1}{2}\right)\right]\right\}, n = 0,\ldots,N-1 \tag{64}$$

Appendix D

We show that on rectangular meshes, FVM and FDM are equivalent. The momentum equation is

$$\frac{\partial u}{\partial t} + \frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} + \frac{\partial \varphi}{\partial x} = \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \tag{65}$$

$$\frac{\partial v}{\partial t} + \frac{\partial uv}{\partial x} + \frac{\partial v^2}{\partial y} + \frac{\partial \varphi}{\partial y} = \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \tag{66}$$

Volume integral,

$$\int_{\mathrm{CV}} \frac{\partial u}{\partial t}\,\mathrm{d}\Omega + \int_{\mathrm{CV}} \left(\frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y}\right)\mathrm{d}\Omega + \int_{\mathrm{CV}} \frac{\partial \varphi}{\partial x}\,\mathrm{d}\Omega = \nu \int_{\mathrm{CV}} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)\mathrm{d}\Omega \tag{67}$$

$$\int_{\mathrm{CV}} \frac{\partial v}{\partial t}\mathrm{d}\Omega + \int_{\mathrm{CV}} \left(\frac{\partial uv}{\partial x} + \frac{\partial v^2}{\partial y}\right)\mathrm{d}\Omega + \int_{\mathrm{CV}} \frac{\partial \varphi}{\partial y}\mathrm{d}\Omega = \nu \int_{\mathrm{CV}} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right)\mathrm{d}\Omega \tag{68}$$

Gaussian theorem,

$$\frac{\partial}{\partial t}\int_{\mathrm{CV}} u\,\mathrm{d}\Omega + \int_{\mathrm{CS}} \left(u^2, uv\right)\cdot \mathrm{d}\vec{S} + \int_{\mathrm{CS}} \left(\varphi, 0\right)\cdot \mathrm{d}\vec{S} = \nu \int_{\mathrm{CS}} \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right)\cdot \mathrm{d}\vec{S} \tag{69}$$

$$\frac{\partial}{\partial t}\int_{\mathrm{CV}} v\,\mathrm{d}\Omega + \int_{\mathrm{CS}} \left(uv, v^2\right)\cdot \mathrm{d}\vec{S} + \int_{\mathrm{CS}} \left(0, \varphi\right)\cdot \mathrm{d}\vec{S} = \nu \int_{\mathrm{CS}} \left(\frac{\partial v}{\partial x}, \frac{\partial v}{\partial y}\right)\cdot \mathrm{d}\vec{S} \tag{70}$$

Here, we assume the cell spacings are non-uniform in the y direction. Due to second-order spatial accuracy, the volume average of u is equal to the cell-center value. The FDM discretization multiplied by volume

$$\begin{aligned}\frac{u_{i+1/2,j}^{n+1} - u_{i+1/2,j}}{\delta t}\delta x \delta y_j &= -\delta y_j \left[\left(u_{i+1,j}\right)^2 - \left(u_{i,j}\right)^2\right] - \delta x\left[\left(u_{i+1/2,j+1/2}\right)\left(v_{i+1/2,j+1/2}\right) - \left(u_{i+1/2,j-1/2}\right)\left(v_{i+1/2,j-1/2}\right)\right] \\ &+\nu\left(\delta y_j\left(\frac{u_{i+3/2,j} - u_{i+1/2,j}}{\delta x} - \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\delta x}\right) + \delta x\left(\frac{u_{i+1/2,j+1} - u_{i+1/2,j}}{\delta y_{j+1/2}} - \frac{u_{i+1/2,j} - u_{i+1/2,j-1}}{\delta y_{j-1/2}}\right)\right) - \delta y_j\left(\varphi_{i+1,j} - \varphi_{i,j}\right)\end{aligned} \tag{71}$$

$$\begin{aligned}\frac{v_{i,j+1/2}^{n+1} - v_{i,j+1/2}}{\delta t}\delta x \delta y_{j+1/2} &= -\delta y_{j+1/2}\left[\left(u_{i+1/2,j+1/2}\right)\left(v_{i+1/2,j+1/2}\right) - \left(u_{i-1/2,j+1/2}\right)\left(v_{i-1/2,j+1/2}\right)\right] - \delta x\left[\left(v_{i,j+1}\right)^2 - \left(v_{i,j}\right)^2\right] \\ &+\nu\left(\delta y_{j+1/2}\left(\frac{v_{i+1,j+1/2} - v_{i,j+1/2}}{\delta x} - \frac{v_{i,j+1/2} - v_{i-1,j+1/2}}{\delta x}\right) + \delta x\left(\frac{v_{i,j+3/2} - v_{i,j+1/2}}{\delta y_{j+1}} - \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\delta y_j}\right)\right) - \delta x\left(\varphi_{i,j+1} - \varphi_{i,j}\right)\end{aligned} \tag{72}$$

This form can be interpreted as FVM if

1) u and v is interpreted as volume average (or volume-center value). Since the grid spacings are non-uniform in the y direction, the location of v is different from the control volume center. Hence, it is better to interpret it as control volume average.

References