Navier-Stokes Generative Adversarial Network

1 Mathematical model

We consider residual of Navier-Stokes equation that

$$r_{cty} = \nabla \cdot \boldsymbol{u} \tag{1}$$

$$r_u = \boldsymbol{u}_t + \sigma \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} - \mu \Delta \boldsymbol{u} + \nabla p \tag{2}$$

Denote the velocity as $\boldsymbol{u} = [u_1, u_2]^T$ and pressure p. In discrete form, we have the residuals $\boldsymbol{r} = [r_{cty}, r_{u_1}, r_{u_2}]$ at point (x_i, y_j, t_n) that

$$\begin{aligned} r_{cty}(x_i, y_j, t_n) \quad = \quad & \frac{u_1(x_{i+1}, y_j, t_n) - u_1(x_{i-1}, y_j, t_n)}{2dx} \\ & + \frac{u_2(x_i, y_{j+1}, t_n) - u_2(x_i, y_{j-1}, t_n)}{2dy} \end{aligned} \tag{3}$$

$$\begin{aligned} r_{u_1}(x_i, y_j, t_n) \quad = \quad & \rho \frac{u_1(x_i, y_j, t_{n+1}) - u_1(x_i, y_j, t_n)}{dt} + \sigma u_1(x_i, y_j, t_n) \\ & + \rho u_1 \frac{u_1(x_{i+1}, y_j, t_n) - u_1(x_{i-1}, y_j, t_n)}{2dx} \\ & + \rho u_2 \frac{u_1(x_i, y_{j+1}, t_n) - u_1(x_i, y_{j-1}, t_n)}{2dy} \\ & + \mu \frac{-u_1(x_{i+1}, y_j, t_n) + 2u_1(x_i, y_j, t_n) - u_1(x_{i-1}, y_j, t_n)}{dx^2} \\ & + \mu \frac{-u_1(x_i, y_{j+1}, t_n) + 2u_1(x_i, y_j, t_n) - u_1(x_i, y_{j-1}, t_n)}{dy^2} \\ & + \frac{p(x_{i+1}, y_j, t_n) - p(x_{i-1}, y_j, t_n)}{2dx} \end{aligned} \tag{4}$$

$$
\begin{aligned}
r_{u_2}(x_i, y_j, t_n) \quad = \quad & \rho \frac{u_2(x_i, y_j, t_{n+1}) - u_2(x_i, y_j, t_n)}{dt} + \sigma u_2(x_i, y_j, t_n) \\
& + \rho u_1 \frac{u_2(x_{i+1}, y_j, t_n) - u_2(x_{i-1}, y_j, t_n)}{2dx} \\
& + \rho u_2 \frac{u_2(x_i, y_{j+1}, t_n) - u_2(x_i, y_{j-1}, t_n)}{2dy} \\
& + \mu \frac{-u_2(x_{i+1}, y_j, t_n) + 2u_2(x_i, y_j, t_n) - u_2(x_{i-1}, y_j, t_n)}{dx^2} \\
& + \mu \frac{-u_2(x_i, y_{j+1}, t_n) + 2u_2(x_i, y_j, t_n) - u_2(x_i, y_{j-1}, t_n)}{dy^2} \\
& + \frac{p(x_i, y_{j+1}, t_n) - p(x_i, y_{j-1}, t_n)}{2dy} \qquad (5)
\end{aligned}
$$

2 MLP

2.1 Proper orthogonal decomposition

Since the cost to train the network with the data of velocity $\boldsymbol{u}$ and pressure p directly is expensive, we apply the proper orthogonal decomposition (POD) to represent $\boldsymbol{u}$ and p on grid. And with the help of the basis functions, we can denote $\boldsymbol{u}$ and p as a linear combination that

$$
\boldsymbol{u}(t_n) \approx \sum_{i=1}^{N_1} c1_i(t_n)\boldsymbol{\phi}_i, p \approx \sum_{i=1}^{N_2} c2_i(t_n)\psi_i \qquad (6)
$$

where $c1$ and $c2$ are the coefficients of velocity and pressure.

2.2 Network Structure

Then, we set the network $\mathcal{N}(c)$ as (blank).

3 Physics loss functions

Here, the network is given the coefficients at time t_n and predicts the coefficients at time t_{n+1}. Consider a pair of coefficents for velocity and pressure $[\boldsymbol{c}1_n, \boldsymbol{c}2_n]$ at time t_n. Then, we have the predict labels as $[\hat{\boldsymbol{c}}1_{n+1}, \hat{\boldsymbol{c}}2_{n+1}]$. By the basis functions $\boldsymbol{\phi}(t_{n+1})$ and $\boldsymbol{\psi}(t_{n+1})$ from POD, we can recover the predict velocity $\hat{\boldsymbol{u}}(t_{n+1})$ and pressure $\hat{p}(t_{n+1})$. Based on the equations (3, 4, 5), we can obtain the residuals $\hat{\boldsymbol{r}}$ of $\hat{\boldsymbol{u}}(t_{n+1})$ and $\hat{p}(t_{n+1})$ accordingly.

References