Machine Learning for Turbulent Fluid Dynamics

Machine Learning for Science [ML4SCI]
Google Summer of Code [GSoC]

Amey Varhade

Mentors:
Prof. Brad Marston, Brown University
Chenyu Zhang, Brown University
Prof. Steve Tobias, University of Leeds

Table of contents

Introduction

Initial Setup

The Newtonian and in-compressible fluids are described by the famous Navier–Stokes equations, which are nonlinear differential equations

$$\frac{\partial \rho}{\partial t} + \overrightarrow{\nabla} \cdot (\rho \overrightarrow{u}) = 0 \tag{1}$$

$$\frac{\partial (\rho \overrightarrow{u})}{\partial t} + \overrightarrow{\nabla} \cdot [\rho \overline{\overline{u \otimes u}}] = -\overrightarrow{\nabla p} + \overrightarrow{\nabla} \cdot \overline{\overline{\tau}} + \rho \overrightarrow{f} \tag{2}$$

$$\frac{\partial (\rho e)}{\partial t} + \overrightarrow{\nabla} \cdot ((\rho e + p) \overrightarrow{u}) = \overrightarrow{\nabla} \cdot (\overline{\overline{\tau}} \cdot \overrightarrow{u}) + \rho \overrightarrow{f} \overrightarrow{u} + \overrightarrow{\nabla} \cdot (\overrightarrow{\dot{q}}) + r \tag{3}$$

The data for the fluid description is generated from the above equation by finding solutions through spectral methods.

We run both the nonlinear and the reduced/quasi-linear simulations through the Dedalus Python package to generate the data varying the conditions according to the state of flow.

Data

The data consists of 4 fields, pressure and velocity P, V_r, v_θ, v_z, at each point in space and time of the cylindrical system of pipe.

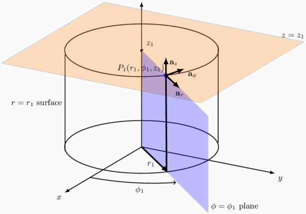


Figure 1: Pipe flow of fluids

According to the Reynolds' number of flow we populate the corresponding data into large HDF5 files

Problem Description

The entire simulation takes a huge time to generate the data, much of it it is very less useful and also redundant. Further treatment and analysis for specialised tasks becomes cumbersome.

Even half a million CPU-hours are not enough to generate the final data for 100 time stamps for a specific state of flow.

The project explores dimensional reduction techniques to help reconstruct the data and to build a statistical theory of the fluid motion in the pipe and further explain turbulence and transition to turbulence.

Proper Orthogonal Decomposition (POD)

Representation

We setup the data as a co-variance matrix including all the 4 pressure and velocity fields together.
We convert the data in spectral domain along azimuthal (θ) direction initially through FFT. In another variation, we apply 2D FFT to convert into spectral domain by transforming along both the azimuthal (θ) and axial (z) directions.

Applying the *POD* we calculate the eigenvalue corresponding to each mode, as we would have a maximum of 1 non-zero eigenvalue corresponding to each mode.

Computations

Only along θ direction.

$$\begin{aligned} c_{ij}(r_1, z_1, r_2, z_2, \theta_1 - \theta_2) &= \overline{q_i(r_1, \theta_1, z_1, t)\ q_j(r_2, \theta_2, z_2, t)} \\ &= \frac{1}{2\pi}\int_0^{2\pi} d\overline{\theta}\ q_i(r_1, \overline{\theta} + \frac{\delta\theta}{2}, z_1, t)\ q_j(r_2, \overline{\theta} - \frac{\delta\theta}{2}, z_2, t) \end{aligned}$$

$$\langle q_m(t) | q_m(t) \rangle = \sum_i \int_{R_1}^{R_2} r\ dr \int_0^L dz\ |q_{im}(r, z, t)|^2$$

Along θ and z direction.

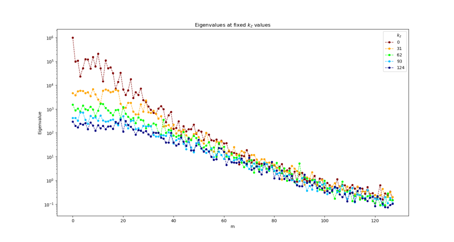
$$c_{ij}(r_1, r_2, \theta_1 - \theta_2, z_1 - z_2, t) = \frac{1}{2\pi}\int_0^{2\pi}\int_0^L d\overline{\theta} d\overline{z}\ q_i(r_1, \overline{\theta} + \frac{\delta\theta}{2}, \overline{z} + \frac{\delta t}{z}, t) q_j(r_2, \overline{\theta} - \frac{\delta\theta}{2}, \overline{z} - \frac{\delta z}{2}, t) \tag{20}$$

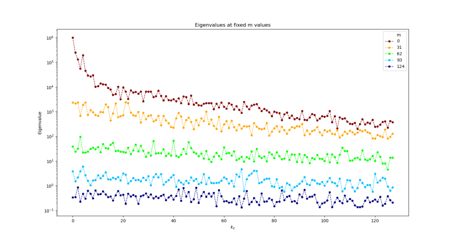
$$\langle q_{m,k}(t) | q_{m,k}(t) \rangle = \sum_i \int_{R_1}^{R_2} r\ dr |q_{imk}(r, t)|^2$$

Plots

Computations

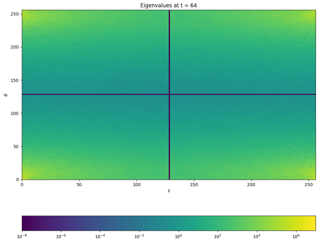
Quasilinear Simulations Data.



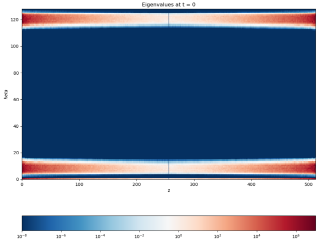


Eigen Heatmaps

Nonlinear

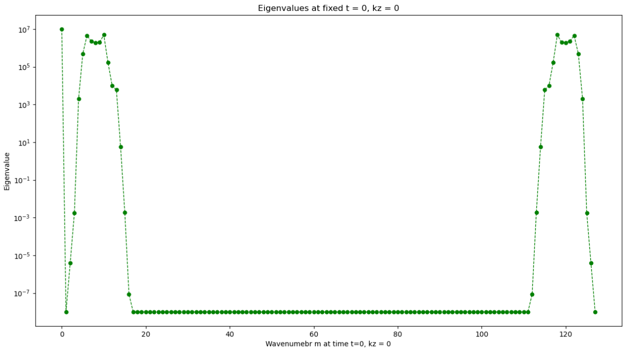


Quasilinear



Computations

Nonlinear Simulations Data.



Conclusion

Further Work

We are in the final stages of concretize and quantify the results from the POD. Work on other techniques has started and will be continue much beyond the GSoC period

Exploring other dimensional reduction techniques including but limited to Kernel methods on POD, PCA.
Implementing various types of Autoencoder architectures and a thorough comparison between the results obtained by them.
Another extension could be to explore Generative models to understand the motion further.

End

Thank You!

Questions?