1. Summary

As an Undergraduate Research Assistant in the Computational Fluid Dynamics Group under Dr. Georgios Matheou, one of my main contributions was the development of a spectral interpolation algorithm for 1-3 spatial dimensions, that is portable to any common scientific computing language. Tested versions in Python and MATLAB were completed, as well as proofs of concept in C and Julia.

Additionally, I implemented a numerical solution of Perturbed Taylor-Green Turbulence using the Vorticity-Streamfunction method, where I am currently implementing particle tracking. Dr. Matheou was helpful in giving advice and direction, but all implementations were solely my work.

2. Spectral Interpolation Algorithm

It is often advantageous to perform operations in frequency space rather than physical space. By using a spectral interpolation algorithm on the frequency-space variables, we can obtain physical-space values at intermediate positions.

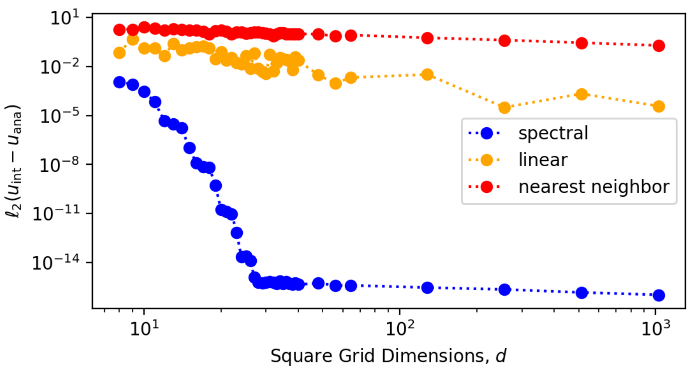

Figure 1. Comparison of Interpolation Error Convergence by Method

This method's accuracy was shown to converge to within an order of machine epsilon for smooth functions on a discrete grid larger than roughly 30×30, much more rapidly than nearest-neighbor or linear interpolation methods, which can be seen in Fig. 1, which was tested on the function $u(x, y) = \exp\Big[\sin(x)\cos(y)\Big]$, which is representative of any smooth periodic function without a finite Fourier series.

3. Perturbed Taylor-Green Turbulence

In order to test this algorithm in a real application, a 2D turbulence model was developed with the Spectral Voriticity-Streamfunction Method. The equation for incompressible (i.e. $\nabla \cdot \vec{u} \equiv 0$) vorticity evolution is as follows:

$$\partial_t [\omega] = \nu \left(\partial_x^2 + \partial_y^2\right) [\omega] - \Big(u \cdot \partial_x [\omega] + v \cdot \partial_y [\omega]\Big) \tag{1}$$

By using discrete spectral methods, where we assume periodicity and smoothness, we obtain the following:

$$\begin{aligned}\partial_t [\Omega_{pq}] = &- \nu \left(k_p^2 + k_q^2\right) [\Omega_{pq}] \\ &- \text{fft2}\Big(u_{ij} \cdot \partial_x [\omega_{ij}] + v_{ij} \cdot \partial_y [\omega_{ij}]\Big)\end{aligned} \tag{2}$$

This equation is transformed with an integrating factor to be:

$$\Xi_{pq} = \exp\left[\nu \left(k_p^2 + k_q^2\right) t\right] \tag{3}$$

$$\partial_t \Big[\Xi_{pq}\Omega_{pq}\Big] = -\underline{i}\,\Xi_{pq}\Big[k_p \cdot \text{fft2}(u_{ij}\omega_{ij}) + k_q \cdot \text{fft2}(v_{ij}\omega_{ij})\Big] \tag{4}$$

Equation (4) was then numerically integrated using a 3rd-order Runge-Kutta method. $\Omega_{pq} = \text{fft2}\Big[\nabla \times \vec{u}_{ij}\Big]$ was initialized with the following:

$$u_{ij} = +\cos(\beta x_i) \cdot \sin(\beta y_j) + \gamma_u \tag{5}$$

$$v_{ij} = -\sin(\beta x_i) \cdot \cos(\beta y_j) + \gamma_v \tag{6}$$

where γ_u and γ_v represent `rng`-generated perturbations to u and v, respectively. Turbulence was then sustained by modifying the Ξ_{pq} variable to be negative for a thin range of higher wavenumbers and boosted for very low wavenumbers. The resultant turbulence can be seen in Fig. 2.

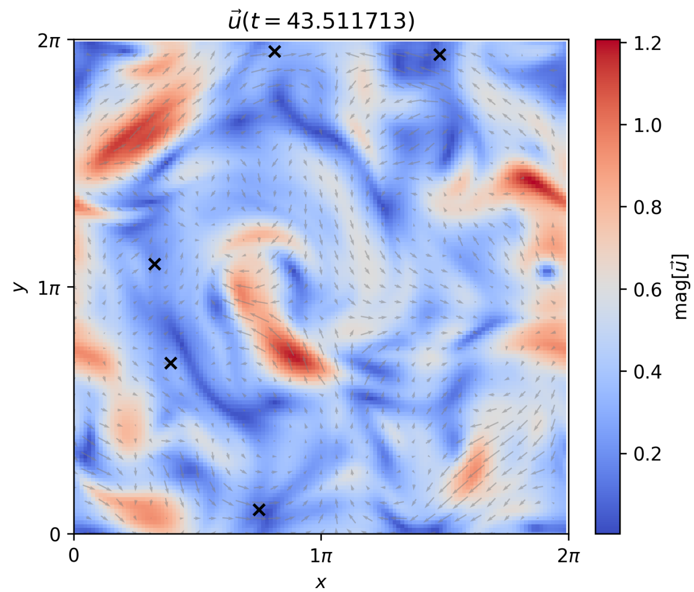

Figure 2. Perturbed Taylor-Greene Turbulence with Particle Tracking

Particle tracking was implemented with a simple Euler integration of the spectrally interpolated velocities at the particle point, and a Predictor-Corrector method is currently in development.

4. Conclusion

In conclusion, through my research with Dr. Matheou, which I believe to be the best example of my work, I was able to develop a broad understanding of numerical and computational methods as they apply to fluid dynamics. I believe the skills that I have developed have prepared me for graduate-level research in the fields of super/hypersonic propulsion.