Pseudospectral scheme for 2D turbulence simulation

Valdemar Skou Knudsen
Technical University of Denmark

Introduction and Motivation

The $2D$ vorticity equations can be derived from the Navier-Stokes equations for a flow with constant angular momentum, and are given by the system

$$\begin{cases} w_t + \psi_y w_x - \psi_x w_y = \frac{1}{Re}\Delta w, \\ w = -\Delta\psi, \end{cases}$$

where ψ is the scalar stream function, Δ is the Laplacian in the spacial coordinates, and $Re > 0$ is the Reynolds Number of the flow. The purpose of this project is to develop a Galerkin approximation scheme of Fourier type to solve the equations in the periodic domain $\Omega = [0, 2\pi]$. The method is party based on [3]. This can be considered a toy-model for simulating turbulent behavior in more complicated geometries, and in higher dimensions.

Construction of the Approximation Scheme

Using Greens formula [1] we get that the weak formulation of the problem is given by

$$\int_\Omega (w_t + \psi_y w_x - \psi_x w_y)\phi d(x,y) = \int_\Omega \Delta w \phi d(x,y) = -\int_\Omega \nabla w \cdot \nabla \phi d(x,y),$$
$$\int_\Omega w\phi d(x,y) = -\int_\Omega \Delta\psi\phi d(x,y) = \int_\Omega \nabla\psi \cdot \nabla\phi d(x,y),$$

where ϕ is an arbitrary test function. To construct our approximation scheme we introduce a uniform grid $(x_{jk}, y_{jk}) = (2\pi j/N_x, 2\pi k/N_y)$, $j = 0, \ldots, N_x$, $k = 0, \ldots, N_y$ on Ω. We can then represent w (and similarly ψ) using the standard $2D$ Fourier interpolant (see [2]):

$$w(t,x,y) \approx \sum_{m=-N_x/2}^{N_x/2-1} \sum_{n=-N_y/2}^{N_y/2-1} \widetilde{w}_{m,n}(t)\phi_{m,n}(x,y),$$

where $\phi_{m,n} = e^{i(mx+ny)}$ is the tensor product of the usual Fourier basis, and $\widetilde{w}_{m,n}$ are the discrete Fourier coefficients. Recall that the key idea for the Galerkin method is to substitute these interpolants into the weak formulation to derive some system of equations involving the coefficients. Indeed, doing so yields the equations

$$\frac{d}{dt}\widetilde{w}_{m,n} + (\widetilde{\psi_y w_x})_{m,n} - (\widetilde{\psi_x w_y})_{m,n} = (m^2+n^2)\widetilde{w}_{m,n},$$
$$\widetilde{w}_{m,n} = -(m^2+n^2)\widetilde{\psi}_{m,n},$$

for all $m = -N_x/2, \ldots, N_x/2-1$, $n = -N_y/2, \ldots, N_y/2-1$. Combining these equations allows us to get an expression for the temporal derivative of each modal coefficient.

Pseudospectral method in 2D

The nonlinear terms in the above equations require special treatment to implement efficiently. For instance, inserting the approximate interpolants of ψ_y and w_x respectively into the expression for the (m,n)'th continuous coefficient

$$(\widehat{\psi_y w_x})_{m,n} = \frac{1}{4\pi^2}\int_\Omega \psi_y w_x \overline{\phi}_{m,n} d(x,y)$$
$$\approx \sum_{j=-N_x/2}^{N_x/2-1} \sum_{k=-N_y/2}^{N_y/2-1} \sum_{p=-N_x/2}^{N_x/2-1} \sum_{q=-N_y/2}^{N_y/2-1} \widetilde{\psi}_{j,k} ik \widetilde{w}_{p,q} ip \delta_{j+p,n}\delta_{k+q,n},$$

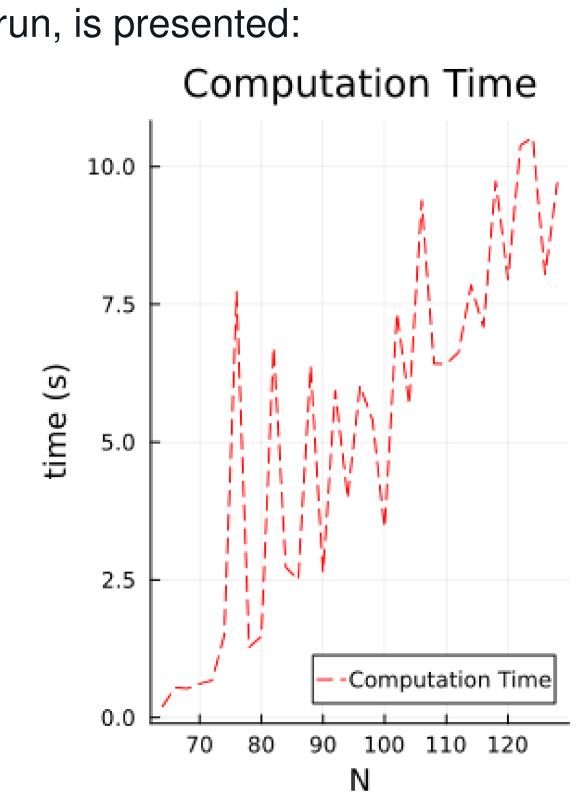
result in a convolution term, which can be evaluated in $\mathcal{O}(n^4)$-time. The pseudospectral method in 2D remedies this through the convolution theorem. In practice we zero-pad our matrices (to prevent aliasing errors, in accordance with Orzsag's $3/2$-rule) and then use the Fast Fourier Transform to compute the terms

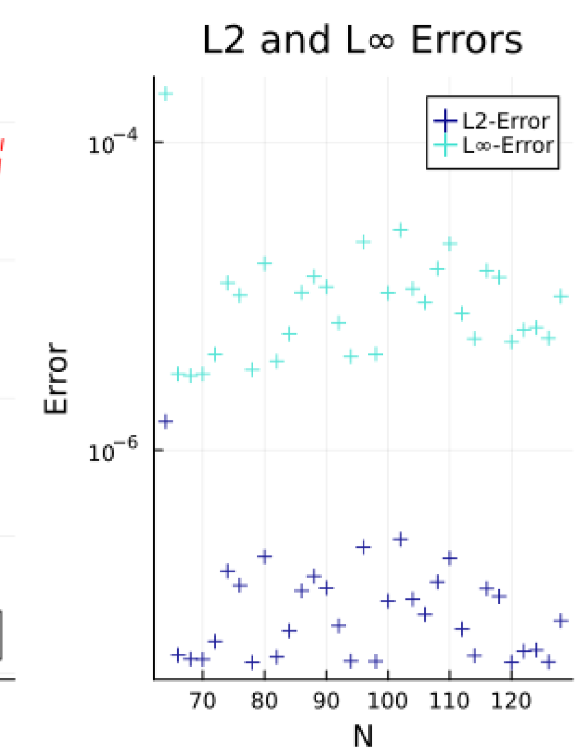
Verification and Performance

For temporal integration we use an adaptive time-stepping method [6], which is readily implemented in [4]. This allows us to circumvent possible difficulties with determining an upper bound for Δt. To verify correctness of the overall derivation and implementation of the method, we use the famous Taylor-Green vortex, defined as

$$w(t,x,y) = 2k\cos(kx)\cos(ky)\exp\left(-2k^2 t/Re\right),$$

where $k = 4$ is the number of vortices in the spacial directions, and we fix $Re = 1$ which corresponds to a high dissipativity of the flow. For simplicy we use $N_x = N_y$ in all computations. A plot of the the resulting errors (both in terms of L^2 and L^∞) for increasing stencil lengths, as well as a plot of the computation time for each run, is presented:

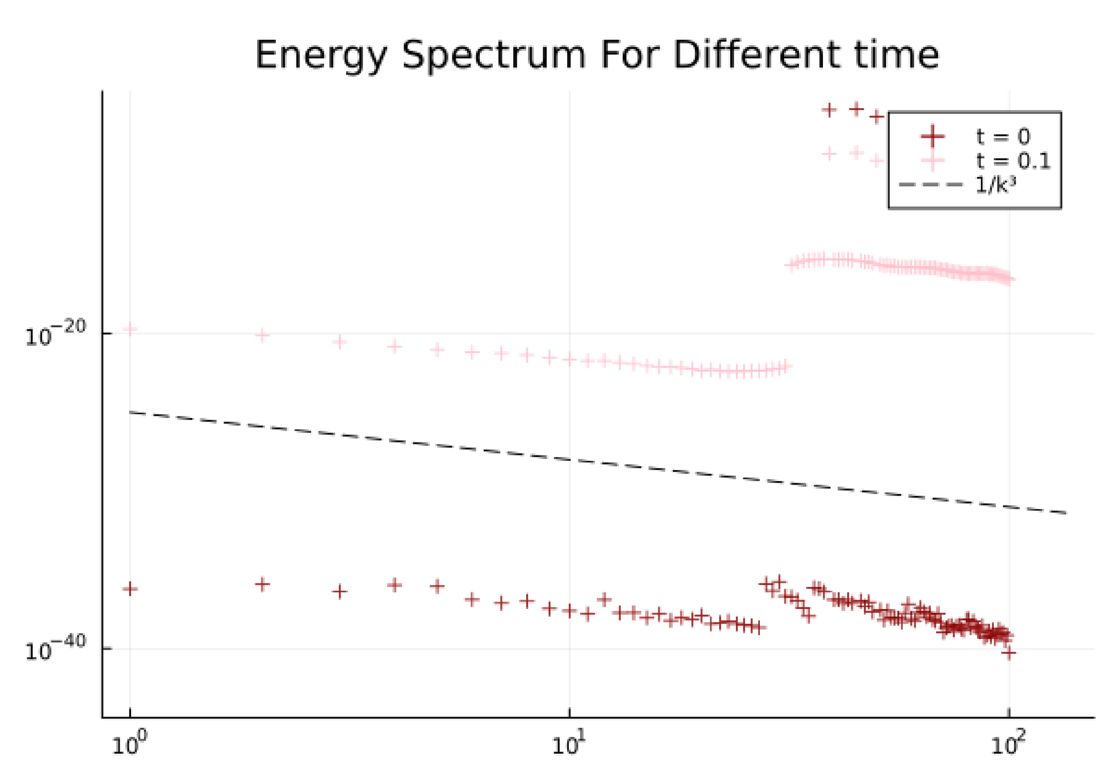




We observe very high accuracy in both norms. Unfortunately the errors do not seem to decrease significantly with the increased resolution. For another measure of the accuracy of our model, consider the following definition for a kinetic energy spectrum function:

$$\hat{E}(k,t) = \sum_{k<m^2+n^2<k+1} |\widetilde{\psi}_{m,n}(t)|^2,$$

where the $\widetilde{\psi}_{m,n}$ are the stream coefficients of the flow at time t. There are other ways to represent this quantity, however is it well known [5] that the theoretical analogue should asymptotically decay with $\mathcal{O}(k^{-3})$ for a fluid model. Indeed we observe in a double log plot:



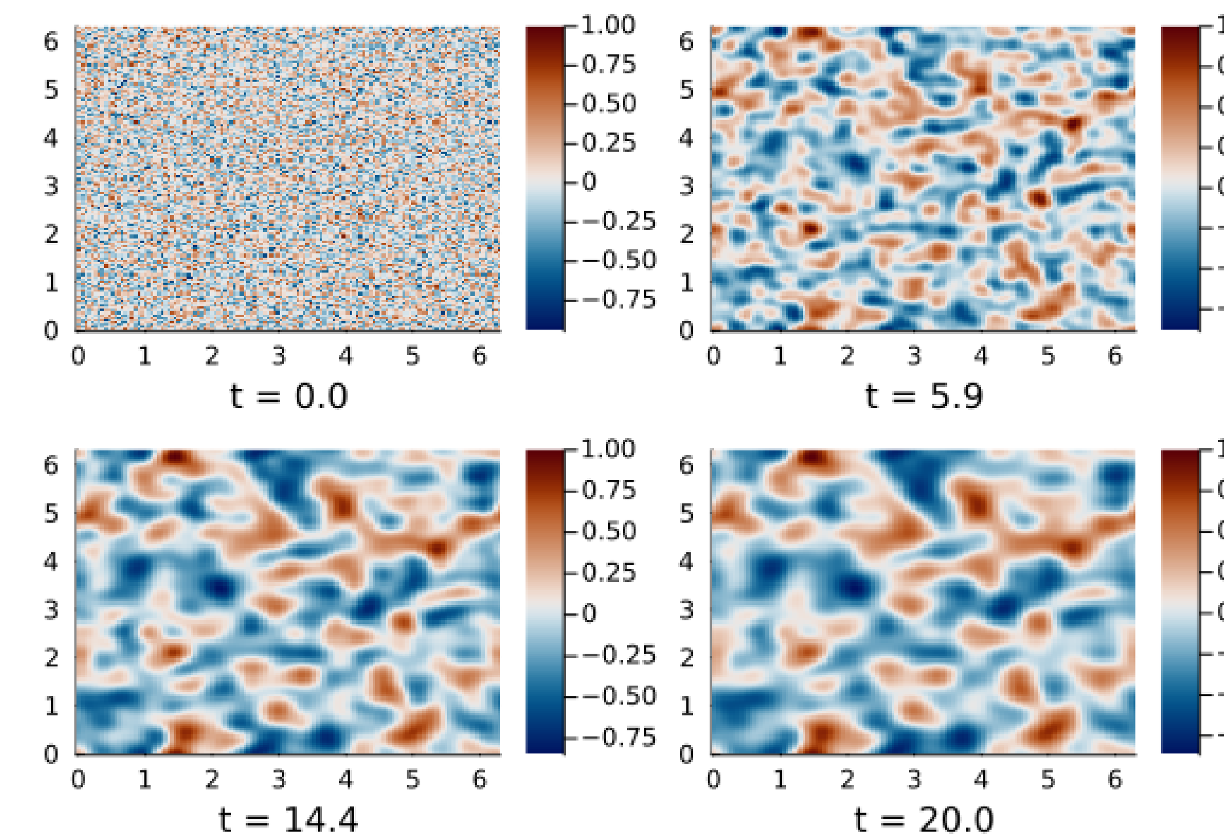
Perhaps a different formulation of the energy spectrum will yield a more accurate result, but it is still quite good, and in accordance with the above mentioned established theory.

Vortices From Random Initial Data

Now that the method and its implementation has been verified we can run numerical experiments with other types of initial data. Any 2π-periodic function will work as an initial field. However, perhaps more interestingly, we can also generate a random initial field, in the sense that $\psi_{m,n} \sim \mathcal{N}(0, \sigma^2)$. For normalization purposes we consider the following definition of the total energy of the flow at time t.

$$E(t) = \sum_{m=-N_x/2}^{N_x/2-1} \sum_{n=-N_y/2}^{N_y/2-1} |\widetilde{\psi}_{m,n}(t)|^2.$$

Note that $E = \hat{E}(0) + \ldots + \hat{E}(N)$ for some sufficiently large N. The initial stream coefficients are normalized in the sense that $E(0) = 1/2$. This construction can simulate the emergence of vortices from the random initial data if we fix a higher Reynolds number, $Re = 1000$, which implies low dissipation of energy. We plot the resulting normalized interpolants from running the simulation with $N_x = N_y = 128$ at a few different time-steps:



We clearly observe the formation of growing vortices for increasing time. Since the plots are normalized, the high dissipativity of this flow is not noticed.

Final Remarks

- The overall method is very accurate but quickly becomes expensive to run, partly due to the curse of dimensionality.
- For simplicity, the scheme was only implemented for stencils with $N_x = N_y$. It could be of potential interest to generalize the current implementation. The model is also not very optimized, so the computation times could be significantly reduced.

References

[1] L. Evans. *Partial Differential Equations.* American Mathematical Society, 2010, pp. 714–715.

[2] David A. Kopriva. *Implementing Spectral Methods for Partial Differential Equations.* Springer, 2009.

[3] Marin Lauber. *Pseudo-spectral method for homogeneous isotropic decaying turbulence.* 2021. URL: https://marinlauber.github.io/2D-Turbulence-Python/.

[4] Christopher Rackauckas and Qing Nie. "Differentialequations.jl–a performant and feature-rich ecosystem for solving differential equations in julia". In: *Journal of Open Research Software* 5.1 (2017).

[5] Omer San and Anne E. Staples. "High-order methods for decaying two-dimensional homogeneous isotropic turbulence". In: *Computers &amp Fluids* 63 (June 2012), pp. 105–127. DOI: 10.1016/j.compfluid.2012.04.006.

[6] Ch. Tsitouras. "Runge–Kutta pairs of order 5(4) satisfying only the first column simplifying assumption". In: *Computers Mathematics with Applications* 62 (2011), pp. 770–775.