410 Tutorial 11: Partial Differential Equations

November 2017

Last week we looked at the basics of spectral methods for interpolation and computing accurate derivatives. This week, we will apply the results of the past few tutorials to solving partial differential equations of parabolic/hyperbolic type. In particular, we will focus on the viscus Burgers' equation defined on a periodic grid.

$$\frac{\partial u(x,t)}{\partial t} = -u(x,t)\frac{\partial u(x,t)}{\partial x} + a\frac{\partial^2 u(x,t)}{\partial x^2} \tag{1}$$

$$u(0,x) = 1 + f(x) \tag{2}$$

1 Finite Difference Methods

By now, you have all of the necessary components to create functional finite difference solvers from scratch. The general procedure is extremely simple; chose a set of finite difference stencils for evaluating spatial derivatives and pair it with some temporal integration scheme. Common choices for non-stiff equations include using second order centered finite difference stencils paired with RK2 or Crank Nicholson integration or fourth order finite difference stencils paired with RK4. For the purpose of this tutorial, we will make use of second order finite difference stencils paired with and RK2 integrator.

1. Write a function which evaluates the right hand side of (1). Remember that you have to treat the boundaries with care as they are periodic.
2. Write a RK2 integrator for this system.
3. With a starting distribution given by $1+\exp(-x^2)$, investigate the stability of the method. You should find that you require $\Delta t < a\Delta x$ with a some number of order 1 for this integration method for stability. This is due to the fact that Burgers' is of hyperbolic type and all modes propagate with finite speed. If $\Delta t > \Delta x$, then the domain of integration doesn't capture the "light-cone" of the problem.
4. With a small, you should get shock formation for generic initial data and your method will inevitably become unstable. This can be fixed with

more advanced techniques from fluid physics which are used to “capture” shocks using flux preserving schemes and adaptive mesh refinement in the presence of shocks.

5. We can further stabilize the method by adding in Kreiss Oliger dissipation to damp out the formation of high frequency modes which are not well resolved on our grid. Modify (1) to

$$\frac{\partial u(x,t)}{\partial t} = -u(x,t)\frac{\partial u(x,t)}{\partial x} + a\frac{\partial^2 u(x,t)}{\partial x^2} + \frac{\epsilon \Delta x^4}{16\Delta t}\frac{\partial^4 u}{\partial x^4} \tag{3}$$

   What does this additional term do from a mechanical perspective? How large may ϵ be? Does adding this term change the differential equation itself or just the numeric implementation?

2 Pseudo Spectral Methods

Spectral methods work by decomposing our function into a series of basis functions and evolving their coefficients as though they were ordinary differential equations. For example, the wave equation, $\frac{\partial^2\psi}{\partial t^2} = \frac{\partial^2\psi}{\partial x^2}$, decomposes under a Fourier transform to become,

$$\frac{\partial^2 \Psi(t,k)}{\partial t^2} = -\left(2\pi k\right)^2 \Psi(t,k) \tag{4}$$

which is completely decoupled and may be integrated independently for each k. Unfortunately, ordinary multiplications become convolutions when Fourier transformed, so non-linearities make the pure spectral method unwieldy for all but the simplest equations. Fortunately, we can efficiently evaluate the non-linear multiplications in real space. Doing so yields what is known as a pseudo spectral method. As a simple example, if we add a non-linear term to our wave equation, $\frac{\partial^2\psi}{\partial t^2} = \frac{\partial^2\psi}{\partial x^2} + \sigma\psi^2$, we may write our evolution equation as:

$$\frac{\partial^2 \Psi(t,k)}{\partial t^2} = -\left(2\pi k\right)^2 \Psi(t,k) + \sigma \mathrm{FT}\left(\mathrm{FT}^{-1}\left(\Psi(t,k)\right)\mathrm{FT}^{-1}\left(\Psi(t,k)\right)\right) \tag{5}$$

where the Fourier transforms my be computed using FFTs for speed.

1. Fourier transform the viscous Burgers’ equation and write down an evolution equation in terms of its Fourier components. Which pieces need to be evaluated in real space?

2. Write an RK2 integrator for this system. Upon testing, you will find that the method is massively unstable. This is due to the fact that as evolution proceeds, higher and higher frequency modes become promoted until they are no longer “well resolved” on our grid. We can

fix this by the process of anti-aliasing; forcably zeroing the high frequency components. A good rule of thumb is that you should zero the highest third of the frequency components after each partial timestep. `Psi_n(int32(nx/3)+1:int32(2*nx/3)) = 0;`

3. Compare the results of the pseudo-spectral and finite difference methods. If you have time, perform convergence tests and/or use independent residual evaluation to ensure you are solving the correct equations.