Department of Mathematics, University of Utah

Orthogonal polynomials/Spectral methods for PDEs

MATH 5750/6880 – Section 002 – Fall 2018

Final project

Partial differential equations

Due December 11, 2018

Solve exactly 2 from the 4 problems below of your choice, and submit solutions for only those two problems.

1. Consider the partial differential equation

$$
\begin{aligned}
-\Delta u + u(x,y) &= f(x,y), & (x,y) &\in (-1,1)^2 \\
u(x,-1) = u(x,1) &= 0, & u(1,y) &= u(-1,y) = 0
\end{aligned}
$$

Use a tensorized Legendre-Galerkin method with polynomials up to degree N in each dimension to solve this partial differential equation. If you write all degrees of freedom in a vector, then the a solution scheme involves solving a large linear system. However, in this case it is easier if you arrange degrees of freedom in a matrix and write the conditions for the scheme in matrix form, in particular as the solution to a *Sylvester equation.*

- Code up a scheme that uses a Sylvester equation solver to compute solutions. (E.g., Python and Matlab have builtin Sylvester equation solvers.)
- Prove, without the use of Lax-Milgram or Céa's Lemma, that the Sylvester equation has a unique solution. (The requisite knowledge on Sylvester equations is available, e.g., from Wikipedia.)
- Choose f so that $u(x,y) = (1-x^2)(1-y^2)\sin(3x+4y)$. Show a convergence plot of the Legendre-Galerkin method as a function of N. What kind of convergence do you see?

2. Consider the partial differential equation

$$
u_t = \boldsymbol{c} \cdot \nabla u(x,y),
$$

with periodic boundary conditions on $(x,y) \in [0,2\pi)^2$. Let the wavespeed be

$$
\boldsymbol{c}(x,y) = (\exp(\sin x), \exp(-\cos x)).
$$

- Code up a Fourier-Galerkin and a Fourier-collocation method for this PDE.
- Discuss the computational complexity of each of your solvers.
- Investigate the accuracy of your solver, both in terms of timestep size and number of polynomial terms N. (E.g., by using an extremely refined comptuational solution as the "exact" solution.)

3. Consider the viscous Burgers' partial differential equation,

$$
\begin{aligned}
u_t + \left(\frac{u^2}{2}\right)_x &= \nu u_{xx}, \quad x \in (-1,1) \\
u(\pm 1, t) &= 0. \\
u(x,0) &= \sin(\pi x).
\end{aligned}
$$

Implement both a Legendre-Galerkin and Legendre-collocation solver for this equation.

- For *viscous* Burgers', $\nu > 0$, show and discuss results for $t > 0$ when ν is small, and when ν is large.
- For *inviscid* Burgers', $\nu = 0$, what differences do you observe between the collocation and Galerkin methods? Do the solutions appear to be accurate for large t?
- For the inviscid Burgers' equation, introduce a filter into your scheme. (Filters for polynomial methods are applied just as filters for Fourier Series methods, with the polynomial degree replacing the frequency parameter.) You can apply a filter after every time step, or after every $P > 1$ timesteps. Experiment with different filters and values of P, and report results for what appears to be a "good" setting for these. How do the filtered results for this PDE compare to the unfiltered results for the viscous Burgers' equation?

4. Consider the second-order wave equation,

$$
\begin{aligned}
u_{tt} &= u_{xx}, & x &\in (-1,1) \\
u(x,0) &= u_0(x), & u_t(x,0) &= v_0(x)
\end{aligned}
$$

We cannot directly solve this equation as written using techniques introduced in this class so far. Introduce an auxilliary variable $v(x,t)$ defined implicitly by the PDE

$$
u_t = v_x, \qquad v(x,0) = \int_{-1}^{x} v_0(s)\,\mathrm{d}s
$$

Show that this definition allows one to write the second-order wave equation to a *system* of two linear first-order, coupled, wave equations of the form $\boldsymbol{w}_t = \boldsymbol{A}\boldsymbol{w}_x$ for a vector $\boldsymbol{w} = (u,v)^T$.

- Derive the appropriate boundary conditions for this system: transform your system via a transformation defined from the diagonalization of the Jacobian $\boldsymbol{A}$ to reveal an uncoupled system, and use this to derive an appropriate set of boundary conditions for the second-order wave equation.
- Impose homogeneous Dirichlet boundary conditions and code up both Legendre-Galerkin and Legendre-collocation schemes to compute solutions to u. Investigate the accuracy of your solver, both in terms of timestep size and number of polynomial terms N. (E.g., by using an extremely refined comptuational solution as the "exact" solution.)