Problem Set 4

Ph.D. Course 2009:
An Introduction to DG-FEM for solving partial differential equations

If you have not already done so, please download all the Matlab codes from the book from

http://www.nudg.org/

and store and unpack them in a directory you can use with Matlab.

We consider the prototype model for nonlinear hyperbolic conservation laws, namely the inviscid Burgers equation in one spatial dimension

$$\frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial u^2}{\partial x} = g(x,t), \quad x \in [-10, 50] \tag{1}$$

In class and in the text there are numerous examples of shock solutions which can be found by using the Rankine-Hugoniot condition directly.

In the exercise we start by considering exact solutions of the form

$$u(x,t) = \frac{1}{\cosh^2(\varepsilon(x+5.0)-t)} + 1.$$

- Plot the function for values of ε equal to 0.1, 1, and 10 – what is the effect of changing ε?
- Derive the right-hand side, $g(x,t)$, for Burgers equation such that the above function is an exact solution.
- Derive and implement a nodal DG-FEM scheme for solving the inviscid Burgers equation - you can use the three files Burgers1Dxxx.m.
- The goal is to run your code until T=50. Try first and run the code for different values of $K = 10$ and $N = 6, 10, 16$ and $\varepsilon = 1.0$ – what do you observe – is the code behaving as you would expect, e.g., is there any substantial difference (other than accuracy) between the low resolution and the high resolution case ?
- Try and remove the dissipative terms in the LF flux (setting maxvel = 0 in **Burgers1DRHS.m**). What do you observe ? – take $K = 10$ and $N = 6$.
- What happens when you change ε ?

- Can you explain your observations ?

Now modify your code so that you can consider the following extensions

1. Exact integration for the nonlinear term
2. Filtering for stabilization
3. Limiting

- Now return to the original goal of running your code until T=50. Run the code for different values of K and N using the three different approaches above – what do you observe – is the code behaving as you would expect ? To make the case more clear you may want to remove the Lax-Friedrich dissipative term as above.
- Discuss the differences between the three approaches – advantages/disadvantages.
- Study carefully the impact of the parameters in the filter, e.g., its order, and how it impact the quality of the solution.
- Implement a TVD-RK scheme for the temporal integration – do you see any differences in the performance of the scheme ?

Enjoy!