AMath 574
Homework 2
Due by 11:00pm on February 2, 2017

For submission instructions, see:

`http://faculty.washington.edu/rjl/classes/am574w2017/homework2.html`

Problem #6.1 in the book

Problem #6.2 in the book

Problem #6.5 in the book

Problem #6.6 in the book

Problem #6.7 in the book

Problem #7.1 in the book

Problem #7.2 in the book

Additional Problem:

Show that the flux-limiter method (6.40),(6.41) can be written as a *wave limiter* method as:

$$Q_i^{n+1} = Q_i^n - \frac{\Delta t}{\Delta x}(\bar{u}^+ \mathcal{W}_{i-1/2} + \bar{u}^- \mathcal{W}_{i+1/2}) - \frac{\Delta t}{\Delta x}(\widetilde{F}_{i+1/2} - \widetilde{F}_{i-1/2}),$$

where $\mathcal{W}_{i-1/2} = Q_i^n - Q_{i-1}^n$ and the "correction flux" is

$$\widetilde{F}_{i-1/2} = \frac{1}{2}|\bar{u}|\left(1 - \frac{\Delta t}{\Delta x}|\bar{u}|\right)\widetilde{\mathcal{W}}_{i-1/2},$$

with the limited waves $\widetilde{\mathcal{W}}$ defined by

$$\widetilde{\mathcal{W}}_{i-1/2} = \phi(\theta_{i-1/2})\mathcal{W}_{i-1/2}.$$

The ratio $\theta_{i-1/2}$ is defined in (6.35) and the function ϕ might be one of limiters from (6.39).

Programming problem

Modify the IPython notebook `$AM574/notebooks/Advection_Examples2.ipynb` to create a new notebook `Advection_Limiters.ipynb` that illustrates your solutions to the following:

a. Implement the wave limiter methods for advection, as described in the previous problem. Note that it's impossible to use half-integer indices, so you might want to declare arrays such as `Ftilde` in which `Ftilde[i]` holds the correction flux $\widetilde{F}_{i-1/2}$. (This is the convention used in Clawpack — the index i often refers to information at the left edge of the cell $x_{i-1/2}$.)

   Copy the `upwind` function definition to a new cell in the notebook and modify it to create a new function `wave_limiter` that has one additional argument `phi` in the calling sequence, so that a limiter function $\phi(\theta)$ can be passed in. The function ϕ might be one of those listed in (6.39a,b) in the book.

   For example:

```
def phi_minmod(theta):
    return(max(0., min(theta,1)))
```

   would define the minmax limiter, and then

```
figs = wave_limiter(x,tfinal,nsteps,u,qtrue,nplot,phi_minmod)
```

   should solve the problem using the minmod wave-limiter method and return a list of figures to animate.

b. Test your function by writing a notebook that produces animations up to time $t = 10$ that correspond to Figures 6.2 and 6.3 from the book. This requires trying out several different limiter functions with 2 sets of initial conditions.

Submit your notebook `Advection_Limiters.ipynb`.

Note: Before submitting your notebook, clear all the output (From the "Cell" menu select "All Output" and then "Clear"), and then save the notebook. Otherwise png figures are stored in the notebook file and it may be very large.

Installing Clawpack. You don't need to turn in anything for this, but you should make sure you have Clawpack installed and working for future assignments.

I posted two videos at `http://faculty.washington.edu/rjl/classes/am574w2017/lectures.html` with tips on installing and getting started with running the code and changing the inputs.

You might want to also clone the `apps` repository, see `http://www.clawpack.org/apps.html`. The directory `apps/fvmbook/chap6` has the Clawpack code that creates Figures 6.2 and 6.3 (newly updated for Version 5.4.0). You might want to use the plots you can create in this directory for comparison.