**Temple 9200** **Computational Methods for Flow Problems** **Spring 2015**

# Problem Set 2

(Out Fri 01/30/2015, Due Wed 02/11/2015)

---

## Problem 2

---

Consider the 1d advection-reaction equation

$$\phi_t + u\phi_x = g(\phi) \tag{1}$$

on the domain $x \in [-1, 1]$, where the flow velocity field is $u(x) = \sin(2\pi x)$, the reaction term is $g(\phi) = -6(\phi - 1)\phi(\phi + 1)$, and the initial state is $\phi(x, 0) = \sin(\pi x)$.

(a) Derive the method of characteristics for equation (1), i.e., determine the functions $a$ and $b$ such that the characteristic equations are of the form $\dot{x}(t) = a(x(t), \phi(t))$ and $\dot{\phi}(t) = b(x(t), \phi(t))$.

(b) Now consider particles that move along the characteristic curves and that carry function values (evolving in time according to the reaction term). Explain how particles move in the $x$–$\phi$ plane (in particular, produce a quiver plot of the direction field $(a, b)$ in the $x$–$\phi$ plane).

(c) Write a code that plots the true solution of problem (1) given above at $t = 3$. Suggested procedure: use the method of characteristics as follows. Set up a regular grid at the final time $t = 3$. For each point on that grid, solve the characteristic equation for $x$ (derived above) *backwards* in time to the initial time $t = 0$. At that position, evaluate the initial conditions. With this value, solve the characteristic equation for $\phi$ forward in time to $t = 3$. This is the true solution value to plot. If you can solve the characteristic ODEs for $x$ and $\phi$ exactly, do so. If not, use a sufficiently accurate numerical ODE solver. Submit your code (by email) under the file name `yourfamilyname_problem2c.m`

(d) Download the Matlab file `temple9200_particle_1d_advection_reaction.m` from the course website `http://math.temple.edu/~seibold/teaching/2015_9200/` and run it. Explain what the code does, and why the final state (at $t = 3$) does not resemble the true solution very well. What is the reason for the inaccuracy.

(e) Fix the problem by adding particle management to the code: insertion of particles into holes, and merging of particles once they get too close. Choose suitable values for the maximum hole size and the minimum particle distance. Run your new code and demonstrate that it recovers the true solution at $t = 3$ very well. Submit (by email) your new Matlab file under the name `yourfamilyname_problem2e.m` .