Dynamic Systems:
Partial Differential Equations

Adapted From:

Numerical Methods in Biomedical Engineering
Stanley M. Dunn, Alkis Constantinides, Prabhas V. Moghe
Chapter 8

Kim Ferlin and John Fisher

Partial Differential Equations

- Transport processes are essential to the function of biological systems
  - Consequently are an important part of mathematical models that describe physiological and cellular processes
- Basis of transport phenomena is founded on the laws of conservation:
  - Mass
  - Momentum
  - Energy
- When applied to the flow of fluids, we develop equations of change
  - More than one independent variable
    - Describes velocity, temperature, concentration changes with respect to time and position
- These processes can be modeled by partial differential equations (PDEs)

Partial Differential Equations

Classification of Partial Differential Equations

- Classified according to order, linearity, and boundary conditions

$$a(*)\partial^2 u/\partial y^2 + 2b(*)\partial^2 u/\partial x \partial y + c(*)\partial^2 u/\partial x^2 + d(*) = 0$$

- If (*) = x, y, constants: the equation is linear
- If (*) = x, y, u, du/dx, du/dy: the equation is quasilinear
- If (*) = x, y, u, d^2u/dx^2, d^2u/dy^2, $d^2u/dxdy$: the equation is non linear

Partial Differential Equations

Initial and Boundary Conditions

- Initial and boundary conditions are necessary in order to obtain unique numerical solutions
- Consider one-dimensional unsteady-state diffusion:

$$\partial c/\partial t = D\partial^2 c/\partial x^2$$

Partial Differential Equations

Initial and Boundary Conditions

- Dirichlet conditions
  - The dependent variable is given at fixed values of the independent variables
- Neumann conditions
  - The derivative of the dependent variable is given as a constant or as a function of the independent variable
- Robbins conditions
  - The derivative of the dependent variable is given as a function of the dependent variable

Partial Differential Equations

Solutions to partial differential equations

- Finite differences

$$\left.\frac{\partial u}{\partial x}\right|_{i,j,k} = \frac{1}{2\Delta x}\left(u_{i+1,j,k} - u_{i-1,j,k}\right)$$

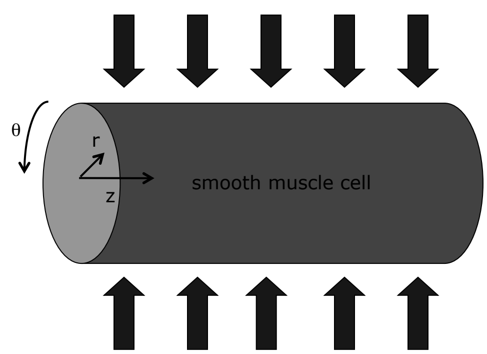
$$\left.\frac{\partial^2 u}{\partial x^2}\right|_{i,j,k} = \frac{1}{\Delta x^2}\left(u_{i+1,j,k} - 2u_{i,j,k} + u_{i-1,j,k}\right)$$

$$\left.\frac{\partial^2 u}{\partial x \partial y}\right|_{i,j,k} = \frac{1}{4\Delta x \Delta y}\left(u_{i+1,j+1,k} - u_{i-1,j+1,k} - u_{i+1,j-1,k} + u_{i-1,j-1,k}\right)$$

Lab Assignment 6

- Investigate the mechanisms behind calcium concentration maintenance in smooth muscle cells
- Develop a simple MATLAB model of calcium transport in a cylindrical model of the smooth muscle cell
  - Use both finite difference approximations and the MATLAB solver “pdepe”
- Compare the two methods of solving a partial differential equation

Modeling Calcium Transport



Ca^{2+} is diffusing into the cell from circumference, but not the ends

Modeling Calcium Transport

- Fick's Law of Diffusion

$$J = -D\nabla u$$

- Inserted into Law of Conservation

$$\frac{\partial u}{\partial t} = D\nabla^2 u + f$$

- Expanding the Laplacian into cylindrical coordinates

$$\frac{\partial c}{\partial t} = D\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial c}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 c}{\partial \theta^2} + \frac{\partial^2 c}{\partial z^2}\right] + f$$

- Simplifying for one dimension

$$\frac{\partial c}{\partial t} = D\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial c}{\partial r}\right) + f$$

  - c ≡calcium concentration,
  - r ≡ radial distance from center of cell,
  - f ≡ source or sink term,
  - D ≡ diffusion coefficient (constant)

Numerical Approximation to the Solution

- The model cell is divided up into concentric annuli or slices (j=50), each with .1 um thickness

- The equation for calcium diffusion without a source:

$$\frac{\partial c}{\partial t} = D\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial c}{\partial r}\right).$$

- Can then be written for each individual annulus as (eq 5)

$$\frac{\Delta Ca}{\Delta t} = \frac{D}{(\Delta r)^2}[(Ca_{j+1} - 2Ca_j + Ca_{j-1}) + \frac{(Ca_{j+1} - Ca_{j-1})}{2(j-1)}] \qquad \text{(Eq. 5)}$$

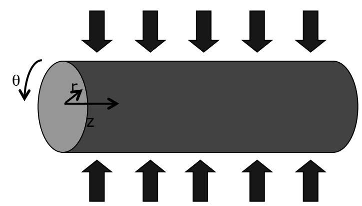
- For the case where j=1 the equation is written as (eq 6)

$$\frac{\Delta Ca}{\Delta t} = \frac{4D}{(\Delta r)^2}(Ca_2 - Ca_1) \qquad \text{(Eq. 6)}$$

Numerical Approximation to the Solution

- Solving these equations will give a value of dCa for a specific point in time
- The time dependent change of diffusion can then be calculated by adding the dCa values to the previous Ca values and continuing to solve the equation for slices in time

- To Summarize:
  - Assign a starting matrix of values for the initial concentration of calcium throughout the cell
  - Use equations 5 and 6 to compute the delta C values for a point in time
  - Add the delta C values to your initial values to give you the calcium concentration values for the next time point
  - Continue to use loops in Matlab to continue to compute these values for the entire simulation



Using MATLAB Solver "pdepe"

- This is an **example** illustrating the straightforward formulation, computation, and plotting of the solution of a single PDE
  - Use as a guide when writing your code
  - NOTE: equations will be different – this problem is not applicable to the assignment

$$\pi^2 \frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x}\right)$$

- This equation holds on an interval $0 \leq x \leq 1$ and for times $t > 0$
- The PDE satisfies the initial condition

$$u(x,0) = \sin \pi x$$

- and boundary conditions

$$u(0,t) = 0$$

$$\pi e^{-t} + \frac{du}{dx}(1,t) = 0$$

- It is convenient to use subfunctions to place all the functions required by pdepe in a single M-file.

Using MATLAB Solver “pdepe”

```
function pdex1
m = 0; % symmetry is zero
x = linspace(0,1,20);
t = linspace(0,2,5);

sol = pdepe(m,@pdex1pde,@pdex1ic,@pdex1bc,x,t);
% Extract the first solution component as u.
u = sol(:,:,1);
% A surface plot is often a good way to study a solution.
surf(x,t,u)
title('Numerical solution computed with 20 mesh points.')
xlabel('Distance x')
ylabel('Time t')
% A solution profile can also show alot
figure plot(x,u(end,:))
title('Solution at t = 2')
xlabel('Distance x')
ylabel('u(x,2)')
```


Using MATLAB Solver “pdepe”

```
% -------------------------------------------------------------
function [c,f,s] = pdex1pde(x,t,u,DuDx)  %This function is the pdepe function
c = pi^2;
f = DuDx;
s = 0;

% -------------------------------------------------------------
function u0 = pdex1ic(x) %This function evaluates initial conditions
u0 = sin(pi*x);

% -------------------------------------------------------------
function [pl,ql,pr,qr] = pdex1bc(xl,ul,xr,ur,t)
%This function evaluates boundary conditions
pl = ul; ql = 0;
pr = pi * exp(-t); qr = 1;
```