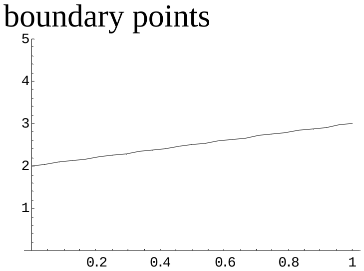
CS205 – Class 18

Covered in Class: 1, 2, 3, 4
Readings: Heath 11.1-11.2

Partial Differential Equations

1. There are three types of PDE's
   a. Elliptic, Hyperbolic, Parabolic
2. The **Laplace Equation** is $\nabla^2 p = \Delta p = \sum_{i=1}^{n} \frac{\partial^2 p}{\partial x_i^2} = f$ is the model elliptic equation
   a. For PDE's we often use subscript notation for derivatives. E.g. in 2D it is $p_{xx} + p_{yy} = f$, in 1D it is $p_{xx} = f$.
   b. Let's look at a simpler case where *f*=0.
   c. In 1D we have $p_{xx} = 0$. The solution analytically is a straight line, i.e., $p = ax + b$.
   d. To determine the exact formula for p we need certain constraints. Those usually come in the form of boundary conditions, i.e. constraints on the value of p or its derivatives for values of x belonging to the boundary of the domain in which we want to solve.
   e. We can also have boundary conditions which involve the derivatives of the function p, called ***Neumann*** boundary conditions. We note that $p'(0) = p'(1) = b$, thus if we supply Neumann conditions for all boundary points then the function p is only determined up to an additive constant. In order to determine the exact function p we must supply ***Dirichlet*** conditions for at least one of the boundary points

      i. Take this plot which represents the solution to the 1D Laplace equation.
         1. We can get it by specifying two Dirichlet conditions $p(0) = 2$ and $p(1) = 3$.
         2. We can also get it by specifying one Dirichlet p(0)=2 and one Neumann p'(0)=1 (or indeed anywhere in 1D i.e. $p_x(t) = 1$ for any t).
         3. Specifying both Neumann end points i.e. $p_x(0)$ and $p_x(1)$ is troublesome.

a. They need to be the same, because our solution is a line. Similar restrictions occur in multi-D. This is called the **compatibility condition**.

b. Additionally only can determine *p* up to constant which is not so bad as many applications only require the derivative of *p*.

f. In more than one dimensions, we need to supply one condition for each node on the boundary of our domain, i.e. a rectangle in 2-D or a rectangular box in 3-D

3. Solving Laplace's Equation Numerically

a. In the more general case of the elliptical PDE $\Delta p = f$ we can no longer analytically compute the solution, therefore we resort to numerical techniques to approximate it

b. In 1-D, we can use the central difference approximation for the second derivative to get $\frac{p_{i-1} - 2p_i + p_{i+1}}{\Delta x^2} = f_i$ for each node i in our grid.

c. Unlike an ODE the solution is determined by the values on the boundaries. As above, moving the end point Dirichlet condition will change the whole line. Laplace's equation is an example of a Boundary Value Differential Equation.

d. Rewriting this as $p_{i-1} - 2p_i + p_{i+1} = \Delta x^2 f_i$ we get the following system of coupled linear equations $1\begin{pmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{pmatrix}1 \cdot \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_{n-1} \\ p_n \end{pmatrix} = \begin{pmatrix} \Delta x^2 f_1 \\ \Delta x^2 f_2 \\ \vdots \\ \Delta x^2 f_{n-1} \\ \Delta x^2 f_n \end{pmatrix}$. We note that some coefficients fall "outside" the matrix, specifically those that would multiply the values of p_0 and p_{n+1}. If we supply Dirichlet boundary conditions for those, we could move those term to the right hand side to get $\begin{pmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{pmatrix} \cdot \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_{n-1} \\ p_n \end{pmatrix} = \begin{pmatrix} \Delta x^2 f_1 - p_0 \\ \Delta x^2 f_2 \\ \vdots \\ \Delta x^2 f_{n-1} \\ \Delta x^2 f_n - p_{n+1} \end{pmatrix}$. The matrix multiplying the unknown function values is a very sparse symmetric, negative definite matrix. Therefore, an iterative method such as preconditioned conjugate gradient would be very efficient in solving it (after a multiplication by -1 to make it positive definite).

e. If we alternatively provide Neumann boundary conditions for, say, the leftmost boundary of our domain we can use the following approximation

$$\frac{\frac{p_2 - p_1}{\Delta x} - \frac{p_1 - p_0}{\Delta x}}{\Delta x} = f_i$$ where $\frac{p_1 - p_0}{\Delta x} = p'_{1/2}$ and replace the first equation of our system with $p_2 - p_1 = \Delta x^2 f_i + \Delta x p'_{1/2}$. In that case the corresponding linear system would be

$$\begin{pmatrix} -1 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{pmatrix} \cdot \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_{n-1} \\ p_n \end{pmatrix} = \begin{pmatrix} \Delta x^2 f_1 + \Delta x\, p'_{1/2} \\ \Delta x^2 f_2 \\ \vdots \\ \Delta x^2 f_{n-1} \\ \Delta x^2 f_n - p_{n+1} \end{pmatrix}$$

. The resulting matrix is still symmetric and negative definite, so the same solution procedure applies.

f. If both boundary conditions were supplied as Neumann, the resulting system would be

$$\begin{pmatrix} -1 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} p_1 \\ p_2 \\ \vdots \\ p_{n-1} \\ p_n \end{pmatrix} = \begin{pmatrix} \Delta x^2 f_1 + \Delta x\, p'_{1/2} \\ \Delta x^2 f_2 \\ \vdots \\ \Delta x^2 f_{n-1} \\ \Delta x^2 f_n + \Delta x\, p'_{n+1/2} \end{pmatrix}$$

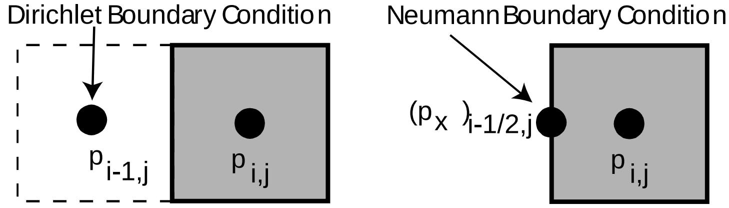
which is a singular system! The matrix A on the left hand side has a null space, which can be seen easily as $A \cdot \begin{pmatrix} 1 & 1 & \cdots & 1 \end{pmatrix}^T = 0$. If we have a given solution p^* of the PDE above, then for any vector z on the null space of A the function $p^* + \alpha z$ is also a solution that satisfies the boundary conditions. To see this consider $A(p^* + \alpha z) = Ap^* + \alpha Az = f + 0 = f$.

g. Luckily, the dimension of the null space of A is just 1 and there exists a version of conjugate gradient that can solve for p up to a constant vector $\alpha \begin{pmatrix} 1 & 1 & \cdots & 1 \end{pmatrix}^T$. That is we get a family of solutions that differ by a constant function. This will suffice if we only care about the derivatives of p and not about its value.

h. As a side note, it is important to consider the positions of the boundary conditions. Dirichlet conditions are located with p. Neumann conditions however are placed at the half grid points.

i. In two or more dimensions, we follow a similar discretization of the Laplacian operator. Specifically in 2-D we get $$\frac{p_{i-1,j} - 2p_{i,j} + p_{i+1,j}}{\Delta x^2} + \frac{p_{i,j-1} - 2p_{i,j} + p_{i,j+1}}{\Delta y^2} = f_{i,j}$$ or in the special case where $\Delta x = \Delta y$, $$\frac{p_{i,j-1} + p_{i-1,j} - 4p_{i,j} + p_{i+1,j} + p_{i,j+1}}{\Delta x^2} = f_{i,j}$$. The resulting linear system has only five nonzero coefficients in each line of the matrix A and can be solved with the same techniques as in the 1-D case.

j. Dirichlet boundary conditions are specified at the cell centers while Neumann conditions are specified at the cell edges. For example on the left of the cell p_{ij} we could specify dirichlet $p_{i-1,j} = \#$ or Neumann $(p_x)_{i-\frac{1}{2},j} = \#$



k. If all the borders are Neumann then the null space of all 1's i.e. $A(1,1,\ldots,1)^T = 0$.

l. Again, the sparsity is a key issue. If you have a 100 by 100 grid. It has 10^8 entries but only $5\text{x}10^4$ entries are non-zero (5 entries per unknown).

m. Use Preconditioned Conjugate Gradient with an incomplete Cholesky preconditioner to solve efficiently.