LECTURE 21

Finite Element Method and Laplace's Equation

In practice, the finite-element is the most commonly used method for developing numerical solutions to partial differential equations. The reason for this is two-fold. First and foremost is the ability of the finite-element method to adapt to irregularly shaped domains; e.g. allowing one to model the aerodynamics of a car. Secondly, instead of producing a table of values, the finite-element method actually produces a function approximating the solution.

Consider the Dirichlet problem for the Poisson equation (a.k.a. the inhomogeneous Laplace equation) on a planar domain D.

$$\nabla^2 u = f(\mathbf{x}) \qquad \forall\, \mathbf{x} \in D \tag{1a}$$

$$u(\mathbf{x})|_{\partial D} = 0 \tag{1b}$$

Here is the basic principle underlying the finite element method. Suppose $u(\mathbf{x})$ is the solution to the above Dirichlet problem. Then

$$\int_D f(\mathbf{x})\, v(\mathbf{x})\, dA = \int_D \left(\nabla^2 u\right) v(x)\, dA \tag{2}$$

for every function $v(x)$ on D. Using Green's theorem we can write the right hand side as

$$-\int_D \nabla u \cdot \nabla v \; dA := -\Phi(u, v)$$

and interprete the latter as a certain inner product between the functions u and v. The basic idea of the finite-element method is to construct a function u such that

$$\Phi(u, v) = \int_D f(\mathbf{x})\, v(\mathbf{x})\, dD$$

for **all** functions $\mathbf{v}(x)$.

A simple-minded way to think about this situation is the following. Suppose you had a vector $\mathbf{u}$ and you knew its inner product with each standard basis vector $\mathbf{e}_1, \ldots, \mathbf{e}_n$. Then you'd be able to figure out $\mathbf{u}$ as

$$\mathbf{u} = (\mathbf{u}, \mathbf{e}_1)\, \mathbf{e}_1 + \cdots + (\mathbf{u}, \mathbf{e}_n)\, \mathbf{e}_n$$

This would be true even if the inner products $(\mathbf{u}, \mathbf{e}_i)$ were *prescribed by some other computation.* That is what the formula

$$\int_D f(\mathbf{x})\, v(\mathbf{x})\, dD = \Phi(u, v)$$

is doing for us, it is prescribing the inner products of u with an arbitrary function v in terms of something we can compute from f.

Okay, that's the key idea. Here's how it's put into practice via the finite-element method.

The finite-element method begins with a *triangulation of the body* D; that is to say, a tiling of D by triagular subdivisions. The idea here is the view the domain D as a finite collection of non-overlapping triangles laid next to each other. Let the interior vertices of these triangles be labeled by $\mathbf{v}_1, \ldots, \mathbf{v}_n$.

Next, we pick a set of N trial functions, $h_1(x,y), \ldots, h_N(x,y)$. Each of these functions, say v_i, is chosen so that

- at the i^{th} vertex $\mathbf{v}_i$, $h_i(\mathbf{v}_i) = 1$ and every other vertex $h_i(\mathbf{v}_j) = 0$
- within each triangle h_i is a linear function, and is piece-wise linear throughout D.

Here is how you set up such a function. Suppose $\mathbf{v}_i$, $\mathbf{v}_j$ and $\mathbf{v}_k$ are the vertices of a triangle $\triangle_{ijk}$. Then on Δ_{ijk} we can define

$$h_i(x,y) = ax + by + c$$

with a, b, c chosen to satisfy

$$\begin{aligned} h_i(\mathbf{v}_i) &= av_{i,x} + bv_{i,y} + c = 1 \\ h_i(\mathbf{v}_j) &= av_{j,x} + bv_{j,y} + c = 0 \\ h_i(\mathbf{v}_k) &= av_{k,x} + bv_{k,y} + c = 0 \end{aligned}$$

From this system of three equations in three unknowns (a, b, and c), we can determine the restriction of h_i to $\triangle_{ijk}$, and similarly we can find the restriction of h_i to any other triangle with $\mathbf{v}_i$ as a vertex. And on all the other triangles in D we can simply set $h_i(x,y) = 0$. Note that once chose our triangulation, effectively all these trial functions $h_1, \ldots, h_n$ are explicitly constructible.

Next we think of approximating our solution $u(x,y)$ as a linear combination of these trial functions

$$u(x,y) \approx \sum_{i=1}^{n} u_i h_i(x,y) \tag{3}$$

and impose the conditions (coming from (2))

$$\int_D f(\mathbf{x})\, h_j(\mathbf{x})\, dA = -\int_D (\boldsymbol{\nabla} u) \cdot (\boldsymbol{\nabla} h_j)\, dA = -\sum_{j=1} u_i \int_D (\boldsymbol{\nabla} h_i) \cdot (\boldsymbol{\nabla} h_j)\, dA \qquad \text{for each } j = 1, \ldots, n \tag{4}$$

Let us now set

$$\begin{aligned} f_j &= \int_D f(\mathbf{x})\, h_j(\mathbf{x})\, dA \\ M_{ji} &= -\int_D (\boldsymbol{\nabla} h_i) \cdot (\boldsymbol{\nabla} h_j)\, dA \end{aligned}$$

The conditions (4) can be written as a matrix equation

$$\begin{bmatrix} M_{11} & \cdots & M_{1n} \\ \vdots & \ddots & \vdots \\ M_{n1} & \cdots & M_{nn} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_n \end{bmatrix}$$

the solution of which can be expressed as

$$\mathbf{u} = \mathbf{M}^{-1}\mathbf{f}$$

Once we find the solution vector $\mathbf{u}$, we have a piece-wise linear function $u(x,y)$ that approximating the actual solution of the original Dirichlet problem. But now look what happens when we evaluate (3) at a vertex $\mathbf{v}_i$

$$u(\mathbf{v}_i) = \sum_{j=1}^{n} u_j h_j(\mathbf{v}_i) = \sum_{j=1}^{n} u_j \delta_{ij} = u_j$$

Thus, **the solution vector u also furnishes with the approximate values of the solution at each the internal vertex points**.