# MAT 228A Notes

Sam Fleischer

September 27, 2016

## 1 Solving the Poisson Equation using Fourier Series

In 1-D, suppose $u_{xx} = f$ on $(0, 1)$ with $u(0) = u(1) = 0$. Note that if $f = -(n\pi)^2 \sin(n\pi x)$ then the solution is $u = \sin(n\pi x)$ by observation. Reframe the problem as $Lu = f$ with $L = \dfrac{\partial^2}{\partial x^2}$. Then with $u_n \coloneqq \sin(n\pi x)$, we have

$$Lu_n = -(n\pi)^2 u_n \tag{1}$$

and thus $u_n$ is an eigenfunction of $L$ with eigenvalue $-(n\pi)^2$. We can show they are orthogonal in $L^2(0, 1)$ by

$$\langle \sin(n\pi x), \sin(m\pi x)\rangle_{L^2(0,1)} = \int_0^1 \sin(n\pi x)\sin(m\pi x)\mathrm{d}x = \begin{cases} \frac{1}{2} & \text{if } n \neq m \\ 0 & \text{else} \end{cases}. \tag{2}$$

It also turns out these eigenfunctions form a complete set in $L^2$, that is $\{u_n\}_{n=1}^{\infty}$ is a basis. Thus, for a given $f \in L^2(0, 1)$, there are coefficients $a_n$ such that

$$f(x) = \sum_{n=1}^{\infty} a_n u_n(x) \tag{3}$$

where the convergence is in $\|\cdot\|_{L^2(0,1)}$. The solution $u$ of $Lu = f$ can also be written as a linear combination of $u_n$,

$$u(x) = \sum_{n=1}^{\infty} \beta_n u_n(x) \tag{4}$$

We can use orthogonality of $u_n$ to explicitly compute $a_n$, and we obtain

$$a_n = 2\langle f(x), u_n(x)\rangle_{L^2(0,1)} \tag{5}$$

Finally, we have

$$L\Big[\sum \beta_n u_n(x)\Big] = \sum a_n u_n(x), \tag{6}$$

and we exploit orthogonality again, taking an inner product of both sides with $u_m$, and we obtain

$$\beta_n = -\frac{a_n}{(n\pi)^2} \qquad \text{for } n = 1, 2, \ldots. \tag{7}$$

In 2-D, it is the same basic idea. We have $\nabla^2 u = u_{xx} + u_{yy} = f$ on $(0, 1) \times (0, 1)$ with $u(x, y) = 0$ on the boundary. The eigenfunctions are given by

$$u_{n,m}(x) = \sin(n\pi x)\sin(m\pi y) \qquad \text{for } n, m = 1, 2, \ldots \tag{8}$$

with eigenvalues $\lambda_{n,m} = -(n^2 + m^2)\pi^2$. The remaining calculations of the general solution to $\nabla^2 u = f$ for any $f \in L^2((0, 1)^2)$ are similar to the solution in 1-D.

# 2 Methods of Solving PDEs Numerically

## 2.1 Finite Differences

Given a PDE, a domain $\Omega$, and boundary conditions, we take the following steps:

1) Discretize the Domain $\Omega$, that is, represent $\Omega$ by a set of points. For example, draw a grid and the points are at the intersections and at the boundary.

2) Represent functions by values at those points.

3) Use discrete values to approximate derivatives using algebraic formulas

The result, assuming the PDE is linear, is an algebraic equation of the form

$$A(\underset{\sim}{u}) = \underset{\sim}{b}. \tag{9}$$

## 2.2 Finite Elements

Reformulate the problem as a variational problem. Rather than solving $\nabla^2 u = f$, define the functional $F$ by

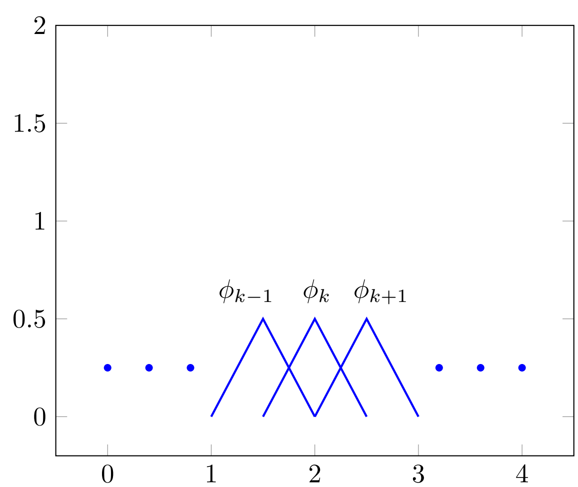
$$F(u) := \int_\Omega \frac{1}{2} \boldsymbol{\nabla} u \cdot \boldsymbol{\nabla} u + uf \mathrm{d}x. \tag{10}$$

The minimizer of $F$ also solves $\nabla^2 u = f$ (using the Euler-Lagrange equation). So we want to find $u \in S$ to minimize $F$ where $S$ is the space of "admissible" functions.

Now let's discretize the function space, i.e. choose a subset of the basis elements of $S$ to represent $S$. Define this subset $S_h$, where $\dim S_h = N < \infty$. Then for any $u_h \in S_h$,

$$u_h(x) = \sum_{k=1}^{N} a_k \phi_k(x) \tag{11}$$

A good basis $\{\phi_k\}_{k=1}^N$ for $S_h$ are tent functions with small overlap, for example,




It turns out $S_h$ is the space of "connect-the-dot" functions (piecewise linear functions). Then we can approximate a function $u \in L^2$ by its closest representation using the basis $\{\phi_k\}_{k=1}^N$, i.e.

$$u(x) \approx u_h(x) := \sum_{k=1}^{N} a_k \phi_k(x). \tag{12}$$

The projection theorem tells us there are unique $\{a_k\}$ such that $\|u - u_h\|_{L^2}$ is minimized.

Anyway, this looks a lot like finite differences, but is philosophically different: the function space is discretized, but not the domain.

Finally, we can solve a minimization problem on $S_h$. What does $F(u_h)$ look like?

$$F(u_h) = \frac{1}{2} \sum_{i,j=1}^{N} A_{i,j} a_i a_j + \sum_{i=1}^{N} b_i a_i \tag{13}$$

where

$$A_{i,j} = \int_{\Omega} \boldsymbol{\nabla}\phi_i \cdot \boldsymbol{\nabla}\phi_j \mathrm{d}x \qquad \text{and} \qquad b_i = \int_{\Omega} f(x)\phi_i(x)\mathrm{d}x \tag{14}$$

and $a_i$ are unknown. To minimize this expression with respect to $\{a_1, \ldots, a_N\}$, we take partial derivatives with respect to each $a_i$, i.e. $\boldsymbol{\nabla}[a_1, \ldots, a_N]$, and set it equal to zero. However,

$$\boldsymbol{\nabla}F = 0 \implies A\underline{a} = \underline{b}, \tag{15}$$

which is a linear system. It turns out that for very simple problems, this is identical to the linear system we acheive using finite differences.

Note that we chose a locally-supported since this gives rise to a sparse matrix $A$, which is easier to solve than arbitrary systems.

### 2.3 Spectral Methods

We can use a representation of $u$, i.e.

$$u_h(x) = \sum_{k=1}^{N} a_k\phi_k(x) \tag{16}$$

where $\phi_k$ are known functions (not limiting ourselves to locally-supported $\phi_k$), for example, take $\phi_k = \sin k\pi x$. From here,

1) Solve the variational problem (but $A$ is not sparse - compuationally expensive)
2) For a set of points $\{x_1, \ldots, x_N\}$ in the domain $\Omega$, solve $\nabla^2 u x_i = f(x_i)$. That is, actually solve the PDE on the grid points (spectral collocation, which is a generalization of the finite differences method).

Thus, again, we have

$$\sum_{i,j=1}^{N} a_j\phi_j(x_i) \qquad \text{for i,j = 1, 2, \ldots N.} \tag{17}$$

And once again we get a linear system:

$$A_{ij} = \phi_j^{''}(x_i), \tag{18}$$

The main problems of this method are

- In general, the matrix $A$ is dense, not sparse