Lecture Notes on Numerical Methods for Partial Differential Equations

2007-2017

Jinn-Liang Liu
Institute of Computational and Modeling Science
National Tsing Hua University, Taiwan

Lecture 1

1D Poisson's Equation and Finite Difference Method (FDM)

2017/4/18

1D Poisson's Problem: Given a function $f(x)$ and two constants g_D and g_N, find the solution $u(x)$ satisfying

$$\begin{aligned} -u'' &= f(x) \quad , \quad \forall\, x \in (0,1) = \Omega : \text{Open set} && (1.1a) \\ u(0) &= g_D \;\; \text{Dirichlet Boundary Condition} && (1.1b) \\ u'(1) &= g_N \;\; \text{Neumann BC} && (1.1c) \end{aligned}$$

Here Ω is the domain of the problem and $\partial\Omega$ is the boundary of Ω, i.e.,$\partial\Omega = \{0,1\}$, $\overline{\Omega} = \Omega \cup \partial\Omega$.

This is a simplified **mathematical model**, an ordinary differential equation with Dirichlet and Neumann boundary conditions (or a boundary value problem). To study a given complex mathematical model, we usually simplify the problem and then construct an *exact* (*true*, *analytical*) solution to the simplified problem. With the solution, we can study the main properties of the problem under investigation. In general, the exact solution of a realistic model is impossible to find by analytical method (by hand). We then resort to a computer to find an *approximate* solution for us. For this, we need a variety of numerical methods that can be implemented (written in a computer programming language such as C++) on the computer. Before applying the numerical methods to the real problem, we must firstly verify the methods with a simplified problem for which the exact solution is already known so that we can check whether our numerical methods are effective and efficient. This course is meant to teach you standard numerical methods for

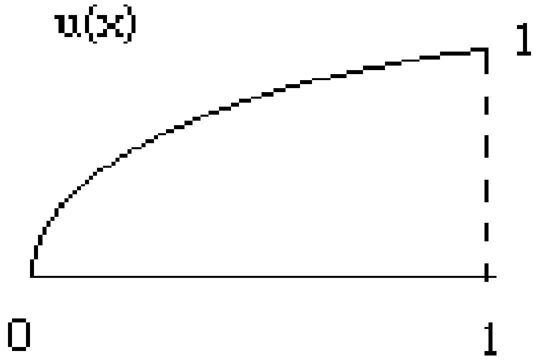

Figure 1: The exact solution graph of Question 1.1(c).

ordinary or partial differential equations (ODEs) or (PDEs) (Part 1) and for *linear algebra* (Part 2).

Question 1.1. Look very closely where the variable x is defined (interior or boundary). (a) What is the unknown in (1.1a)? (b) If $f(x) = 2$, can you find a solution (call an exact solution) of (1.1a)? More solutions? Infinitely many solutions (general solution)? (c) If $g_D = 0$ and $g_N = 0$, how many solutions you get? Can you draw a picture for your solution(s)? (d) If we change the conditions $u(0) = 0$ and $u'(1) = 0$ to $u'(0) = 0$ and $u'(1) = 0$, how many solutions you get? (e) Now if you are given $f(x) = \sin x + \cos x^2 + \ln x^4 + e^{\sin x}$, can you use your hand to find an exact solution for (1.1a)?

Now another important question is: *How do we find an approximate solution of (1.1) for any arbitrary $f(x)$, g_D, and g_N?* This is the main purpose of this course to teach you how to find an approximation solution of an ODE or PDE problem. Here is the simplest method for Problem (1.1) in Part 1.

Part 1: Numerical Method for PDEs

Finite Difference Method (FDM):

Step 1. Domain Discretization (Mesh Generation)

Uniform Mesh (Partition): We partition (discretize) the domain $\overline{\Omega} = [0, 1]$ into $N-1$ subintervals (meshes or elements) with uniform mesh size $\Delta x = h = \frac{1}{N-1}$ and N mesh (grid) points (nodes) x_i, $i = 1, \cdots N$. Hence, $x_i = 0 + (i-1)\Delta x$, $x_{i+1} = 0 + i\Delta x$, $x_{i+\frac{1}{2}} = 0 + (i - \frac{1}{2})\Delta x$ etc.

Step 2. Central Difference Approximation

The following is the definition of a derivative that you learn from Calculus.

$$
\begin{aligned}
u'(x_i) &= \lim_{\Delta x \to 0} \frac{u(x_i + \Delta x) - u(x_i)}{\Delta x} \quad \text{(Forward)} && (1.2) \\
&= \lim_{\Delta x \to 0} \frac{u(x_i) - u(x_i - \Delta x)}{\Delta x} \quad \text{(Backward)} && (1.3) \\
&= \lim_{\Delta x \to 0} \frac{u(x_{i+\frac{1}{2}}) - u(x_{i-\frac{1}{2}})}{\Delta x} \quad \text{(Central)} && (1.4) \\
&\approx \frac{u_{i+\frac{1}{2}} - u_{i-\frac{1}{2}}}{\Delta x} \quad \text{(Central Difference)} && (1.5) \\
&\approx \frac{U_{i+\frac{1}{2}} - U_{i-\frac{1}{2}}}{\Delta x} && (1.6)
\end{aligned}
$$

Note the difference between u_i (exact) and U_i (approximation), i.e., $u_i = u(x_i) \approx U\ _i$ where $U\ _i$ are unknown scalars that we are looking for.

$$
\begin{aligned}
u''(x_i) &= \lim_{\Delta x \to 0} \frac{u'_{i+\frac{1}{2}} - u'_{i-\frac{1}{2}}}{\Delta x} && (1.7) \\
&= \lim_{\Delta x \to 0} \frac{\lim_{\Delta x \to 0} \frac{u_{i+1} - u_i}{\Delta x} - \lim_{\Delta x \to 0} \frac{u_i - u_{i-1}}{\Delta x}}{\Delta x} && (1.8) \\
&\approx \frac{u_{i-1} - 2u_i + u_{i+1}}{(\Delta x)^2} && (1.9)
\end{aligned}
$$

Substituting this expression into (1.1a), we have

$$
-\frac{U_{i-1} - 2U_i + U_{i+1}}{(\Delta x)^2} = f_i = f(x_i), \ \forall i = 2, ..., N-1. \qquad (1.10)
$$

The Dirichlet BC (1.1b) $u(0) = u(x_1) = g_D$ implies that

$$
U_1 = g_D \qquad (1.11)
$$

whereas the Neumann BC (1.1c) can be approximated by

$$\begin{aligned} g_N &= u'(1) = u'(x_N) && (1.12)\\ &= \lim_{\Delta x \to 0} \frac{u(x_N) - u(x_{N-1})}{\Delta x} && (1.13)\\ &\approx \frac{U_N - U_{N-1}}{\Delta x} \Rightarrow && (1.14)\\ U_N - U_{N-1} &= \Delta x \cdot g_N = h \cdot g_N && (1.15) \end{aligned}$$

Combining (1.11), (1.10), and (1.12), we obtain the system of linear algebraic equations:

$$A_{NxN}\overrightarrow{U} = \overrightarrow{b} \quad (1.16)$$

$$A_{NxN} = \begin{bmatrix} 1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ -1 & 2 & -1 & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & -1 & 2 & -1 & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & -1 & 2 & -1 & 0 & \cdots \\ \vdots & \cdots & \vdots & \ddots & \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 0 & -1 & 2 & -1 \\ 0 & \cdots & \cdots & \cdots & 0 & 0 & -1 & 1 \end{bmatrix} \quad (1.17)$$

$$\overrightarrow{U} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ \vdots \\ U_i \\ \vdots \\ \vdots \\ U_N \end{bmatrix} \qquad \overrightarrow{b} = \begin{bmatrix} g_D \\ h^2 f_2 \\ \vdots \\ h^2 f_i \\ \vdots \\ h^2 f_{N-1} \\ h g_N \end{bmatrix} \quad (1.18)$$

where A is called an N by N coefficient matrix, $\overrightarrow{x}$ is an N by 1 unknown vector, and $\overrightarrow{b}$ is an N by 1 known vector.

In linear algebra, we usually use the notation $A\overrightarrow{x} = \overrightarrow{b}$ for $A\overrightarrow{U} = \overrightarrow{b}$. Do not confuse $\overrightarrow{x}$ with the grid points x_i.

Step 3. Solving the Linear System $A\overrightarrow{x} = \overrightarrow{b}$ ($A\overrightarrow{U} = \overrightarrow{b}$)

Question 1.2. Write in detail from Domain Discretization to the linear system $A\overrightarrow{U} = \overrightarrow{b}$ with $N = 5$ for the example

$f(x) = 2$, $g_D = 0$ and $g_N = 0$.

Homework 1.1. Can you define a (polynomial of degree m) function, say $U(x)$, such that the graph of $U(x)$ passes through the points (x_i, U_i)? Use the same degree of polynomial to define another function, say $u^I(x)$, such that its graph passes through the points (x_i, u_i). Draw a picture to tell the difference between $u(x)$, $u^I(x)$, and $U(x)$. We say that $u^I(x)$ is a *piecewise linear interpolation* of $u(x)$ at $x_1, x_2, \cdots, x_N$ if $m = 1$ and $N = 5$.

Part 2: Numerical Method for Linear Algebra $A\overrightarrow{x} = \overrightarrow{b}$

Given A and $\overrightarrow{b}$, there are two ways to solve $A\overrightarrow{x} = \overrightarrow{b}$ for the unknown vector $\overrightarrow{x}$.

(1) *Direct* Methods: Gaussian Elimination etc. These methods are appropriate for small systems with $(N < 10000)$.

(2) *Iterative* Methods: Jacobi, Gauss-Seidel, SOR, Conjugate-Gradient, etc. for very large systems.

We will spend much of our class hours on Part 2 for this course.