Problem Set VII: Solving the 1D Heat Equation

Vlad Jouravlev

October 28, 2017

Abstract

In this paper, the 1D heat equation is solved by separation of variables and boundary conditions. A finite-difference method is then used to construct a numerical solution to the equation.

1 Theory

The 1D heat equation models the time evolution of heat-distribution in a rod of length L. The equation can be written as

$$\frac{\partial u(x,t)}{\partial t} = \alpha \frac{\partial^2 u(x,t)}{\partial x}$$

This equation describes a heat-conducting rod, with in this case insulated ends. Constraints on the derivative at the boundary are known as Neumann boundary conditions (constraints on the function value at the boundary are known as Dirichlet boundary conditions. The boundary conditions for two insulated ends are:

$$\begin{aligned} u(0,t) &= 0 \\ t &> 0 \\ u(l,t) &= 0 \\ t &> 0 \\ u(x,0) &= f(x) \end{aligned}$$

To solve this equation, separation of variables is used:

$$\begin{aligned} X(x)T'(t) &= \alpha X''(x)T(t) \\ \frac{X''(x)}{X(x)} &= \alpha \frac{T'(t)}{T(t)} = \lambda \end{aligned}$$

Solving these two equations then gives:

$$\begin{aligned} X(x) &= Ae^{\sqrt{\lambda}x} + Be^{-\sqrt{\lambda}x} \\ T(t) &= Ce^{\alpha\lambda t} \end{aligned}$$

Applying the boundary conditions and doing some work on these solutions gives the following final solution to the 1D heat equation:

$$\begin{aligned} u(x,t) &= \sum_{k=1}^{\infty} \beta_k \sin(k\pi x/l) e^{-k^2\alpha\pi^2 t/l} \\ \beta_k &= \frac{2}{l}\int_0^l f(x)\sin(k\pi x/l)dx \end{aligned}$$

This is the analytical solution to the 1D heat equation.

To numerically estimate the solution to this equation, the finite-difference method is used to cast the equation into:

$$\frac{u_{i,n+1} - u_{i,n}}{\Delta t} - \alpha \frac{u_{i+1,n} - 2u_{i,n} + u_{i-1,n}}{(\Delta x)^2}$$

From this form a matrix can be constructed with upper and lower diagonals of ones and a central diagonal of -2. The equation can then be solved in the eigenvalue fashion.

2 Bibliography

Landau, Rubin H., et al. Computational Physics: Problem Solving with Python. Wiley-VCH, 2015.