Krzysztof Szyszkiewicz-Warzecha

The Heat/Diffusion equation

The problem is to find the solution to the following partial differentia equation

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2}, \qquad 0 < x < 1, \quad t > 0, \tag{1}$$

subject to the Dirichlet boundary conditions

$$u(0,t) = 0,\ u(1,t) = 0, \quad t > 0, \tag{2}$$

and the initial conditions

$$u(x,0) = u_0(x), \quad 0 < x < 1, \tag{3}$$

where function $u_0 : [0,1] \to \mathbb{R}$ and coefficient $D > 0$ are given. In a compact form we usually write this problem as

$$\begin{cases} \dfrac{\partial u}{\partial t} = D\dfrac{\partial^2 u}{\partial x^2}, & 0 < x < 1, \quad t > 0, \\ u(0,t) = 0,\ u(1,t) = 0, & t > 0, \\ u(x,0) = u_0(x), & 0 < x < 1. \end{cases} \tag{4}$$

The choice of the unit interval $[0,\ 1]$ for the domain of x is not a real restriction as we can always transform every closed interval $[a,\ b]$ into the unit interval by the change of variables

$$x = a + (b-a)\overline{x}, \quad \overline{x} \in [0,\ 1].$$

Now if we have the equations

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2},$$

for $x \in [a,\ b]$, we can consider the function $\overline{u}(\overline{x},t) := u(a+(b-a)\overline{x},t)$ that is defined on the unit interval for space variable and calculating derivatives

$$\frac{\partial \overline{u}}{\partial t} = \frac{\partial u}{\partial t}, \quad \frac{\partial^2 \overline{u}}{\partial \overline{x}^2} = (b-a)^2\frac{\partial^2 u}{\partial x^2},$$

the equation is transformed into

$$\frac{\partial \overline{u}}{\partial t} = \frac{D}{(b-a)^2}\frac{\partial^2 \overline{u}}{\partial \overline{x}^2}, \quad 0 < \overline{x} < 1, \tag{5}$$

thus we end up with the standard form (4) but with a new diffusion coefficient, $\overline{D} := D/(b-a)^2$.

Sometimes it is also convenient to rescale the time variable. This means introducing the variable $t = t_c \overline{t}$, where $t_c > 0$ is the rescaling factor. Now we have the following substitution

$$\overline{u}(\overline{x},\overline{t}) := u(t_c\overline{t}, a+(b-a)\overline{x}),$$
$$\frac{\partial \overline{u}}{\partial \overline{t}} = t_c \frac{\partial u}{\partial t}, \quad \frac{\partial^2 \overline{u}}{\partial \overline{x}^2} = (b-a)^2 \frac{\partial u}{\partial t}, \tag{6}$$

what gives

$$\frac{\partial \overline{u}}{\partial \overline{t}} = \overline{D}\frac{\partial^2 \overline{u}}{\partial \overline{x}^2}, \qquad 0 < \overline{x} < 1, \ \overline{t} > 0, \tag{7}$$

where $\overline{D} = \dfrac{t_c D}{(b-a)^2}$. Now we see that if we select the time rescaling factor $t_c = \frac{(b-d)^2}{D}$, then the diffusion coefficient is one, and the diffusion equations is just $\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$.

Finite Difference Schemes for the Diffusion equation – an explicit scheme

We start we the most straightforward discretization type for diffusion equation (1) which is an explicit scheme. The way to obtain such scheme is similar to the way used for the two-point boundary value problem for ODE on the interval but this time we have to also discretize with respect to time. Let us introduce the grid $\{(x_j, t_m)\}$ on the domain $\overline{\Omega} = [0, 1] \times [0, \infty)$. Here we take uniform grid

$$\begin{aligned} &0 = x_0 < x_1 < \ldots < x_{n+1} = 1, \\ &0 = t_0 < t_1 < t_2 < \ldots \\ &x_{j+1} - x_j = h, \quad t_{m+1} - t_m = k. \end{aligned} \tag{8}$$

The equation (1) at the grid point (x_j, t_m)

$$\frac{\partial u}{\partial t}(x_j, t_m) = D\frac{\partial^2 u}{\partial x^2}(x_j, t_m), \tag{9}$$

and we use the central difference for the second derivative

$$\frac{\partial^2 u}{\partial x^2}(x_j, t_m) \approx \frac{u(x_{j+1}, t_m) - 2u(x_j, t_m) + u(x_{j-1}, t_m)}{h^2} \tag{10}$$

and forward difference for time derivative

$$\frac{\partial u}{\partial t}(x_j, t_m) \approx \frac{u(x_j, t_{m+1}) - u(x_j, t_m)}{k} \tag{11}$$

If we denote by u_j^m the approximation to $u(x_j, t_m)$, than the above differences motivate the following scheme

$$\frac{u_j^{m+1}-u_j^m}{k}=D\frac{u_{j-1}^m-2u_j^m+u_{j+1}^m}{h^2} \quad \text{for} \quad j=1,2,\dots,n,\ m=0,1,2\dots$$

By using the boundary conditions (2), we also have

$$u_0^m=0 \text{ and } \quad u_{n+1}^m=0$$

for all m $m\in\mathbb{N}$ and initial condition (3) produces

$$u_j^0=u_0(x_j) \quad \text{for} \quad j=1,\dots,n.$$

Let $r:=D\frac{k}{h^2}$, then the whole scheme can be written as

$$\begin{cases} u_j^{m+1}=ru_{j-1}^m+(1-2r)u_j^m+ru_{j+1}^m, \\ u_j^0=u_0(x_j), \quad u_0^m=u_{n+1}^m=0, \quad j=1,\dots,n,\ m=0,1,2,\dots \end{cases} \tag{12}$$

When the scheme is written in this form, we see why it is called an *explicit* scheme. It is because the values on the time level $t=t_{m+1}$ are computed using only the values from the previous time level $t=t_m$.(This is in contrast to *implicit* schemes where we have to solve some addition system of equations to pass from one time level to the next. Nevertheless such schemes do posses some properties that make them attractive and useful).

The expressions in (12) may be easily implemented in the form of the C/C++ procedure as follows. Here we assume that we calculate M time steps.

```
u[0]=0; u[n+1]=0;
for (j=1; j <= n; j++) u[j] = u0(x[j]);
for (m=0; m < M; m++) {
    for (j=1; j<=n; j++) u1[j]=r*u[j-1]+(1-2*r)u[j]+r*u[j+1];
     //output or store results from table u1
    //update table u and proceed to the next time step
    for (j=1; j<=n; j++) u[j]=u1[j];
}
```

Stability analysis for explicit scheme

This type of analysis is aimed at the explanation of the behavior of iterations for long times, i.e. what happens to the sequence $u^m=(u_1^m,\dots,u_n^m)$ as $m\to\infty$. Let us write the scheme (12) in a matrix notation

$$u^{m+1}=Au^m, \tag{13}$$

where the matrix $A\in\mathbb{R}^{n\times n}$ is given by

$$A = \begin{bmatrix} 1-2r & r & 0 & 0 & \dots & 0 \\ r & 1-2r & r & 0 & \dots & 0 \\ 0 & r & 1-2r & r & & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & & & r & 1-2r & r \\ 0 & \dots & \dots & 0 & r & 1-2r \end{bmatrix}. \qquad (14)$$

Obviously we have $u^{m+1} = Au^m = AAu^{m-1} = A^2u^{m-1} = \ldots = A^m u^0$, so the iterations (13) may be written as

$$u^{m+1} = A^m u^0, \quad m = 0,1,2,\ldots \qquad (15)$$

For long times the solution of the problem (4) converges to zero no matter what were the initial conditions: $\lim_{t\to\infty} u(x,t) = 0$. Thus our basic requirements for the reliability of the explicit scheme (12) would be to follow that pattern, that is for every $u^0 \in \mathbb{R}^n$ it should produce a sequence $\{u^m\}_{m=0}^{\infty}$ such that $\lim_{m\to\infty} u^m = 0$. The criterion for such property of the matrix A is established in a standard course of linear algebra and it says that the spectral radius $\rho(A)$ is less than 1. The spectral radius is defined as the greatest absolute value of all eigenvalues of the matrix

$$\rho(A) = \max\{|\lambda|: \ \lambda \text{ is an eigenvalue of } A\}. \qquad (16)$$

It can be shown that the eigenvalues of the matrix (14) are following

$$\lambda_j = 1 - 2r(1 - \cos\tfrac{j\pi}{n+1}), \quad j = 1,\ldots,n.$$

Thus the stability criterion $\rho(A) < 1$ for the (15) iteration now means $|\lambda_j| < 1$ for all $j = 1,\ldots,n$ so

$$-1 < 1 - 2r(1 - \cos\tfrac{j\pi}{n+1}) < 1,$$

which gives $r(1 - \cos\tfrac{j\pi}{n+1}) < 1$. As $\cos\tfrac{j\pi}{n+1}$ may be close to -1, we get the basic restriction on r

$$r \le \frac{1}{2}. \qquad (17)$$

Because $r = D\frac{k}{h^2}$, this restriction relates the time step k to spatial step h in the following manner

$$k \le 0.5Dh^2. \qquad (18)$$

It is a quite severe restriction. For example if $D = 1$ and $h = 0.01$, then $k \le 5\cdot 10^{-5}$. It means that to simulate the process over the time duration $T = 10$ we would have to go through $10/(5\cdot 10^{-5})$ or $200\,000$ times steps!

Finite Difference Schemes for the Diffusion equation – an implicit scheme

We have presented one particular numerical method for solving the diffusion equation (4). This explicit method is expressed by the formulas (12) and obviously it is very easy to implement on a computer. However, the explicit method suffers one major drawback, that is it requires very small time step due to the stability condition (18).

Using the notation from the explicit scheme, we apply now the following approximation for the discretization of (4)

$$\frac{\partial^2 u}{\partial x^2}(x_j, t_{m+1}) \approx \frac{u(x_{j+1}, t_{m+1}) - 2u(x_j, t_{m+1}) + u(x_{j-1}, t_{m+1})}{h^2} \tag{19}$$

and

$$\frac{\partial u}{\partial t}(x_j, t_m) \approx \frac{u(x_j, t_{m+1}) - u(x_j, t_m)}{k}. \tag{20}$$

We see that the only difference between these approximations and those employed in the explicit scheme – (10), (11) – is the time level for the second derivative approximation in expression (19), which is now t_{m+1} rather than t_n. This leads to the following scheme

$$\frac{u_j^{m+1} - u_j^m}{k} = D\frac{u_{j-1}^{m+1} - 2u_j^{m+1} + u_{j+1}^{m+1}}{h^2} \quad \text{for} \quad j = 1, 2, \ldots, n, \; m = 0, 1, 2\ldots$$

The boundary conditions (2) imply that

$$u_0^m = 0 \text{ and } \quad u_{n+1}^m = 0$$

for all m $m \in \mathbb{N}$ and initial condition (3) produces

$$u_j^0 = u_0(x_j) \quad \text{for} \quad j = 1, \ldots, n.$$

Using the same notation as in the explicit scheme, $r = D\frac{k}{h^2}$, the above equation may be written as

$$\begin{cases} -ru_{j-1}^{m+1} + (1+2r)u_j^{m+1} - ru_{j+1}^{m+1} = u_j^m, \\ u_j^0 = u_0(x_j), \quad u_0^m = u_{n+1}^m = 0, \quad j = 1, \ldots, n, \; m = 0, 1, 2, \ldots \end{cases} \tag{21}$$

In order to pass from time level t_m to t_{m+1} we have to compute $u^{m+1} = (u_1^{m+1}, \ldots, u_n^{m+1})$ from known vector $u^m = (u_1^m, \ldots, u_n^m)$ and it requires solving the system of linear equation (21). Fortunately, in this case it is a tridiagonal system. In a matrix from it can be written as

$$Au^{m+1} = u^m, \quad m = 0, 1, 2, \ldots \tag{22}$$

where the matrix $A \in \mathbb{R}^{n \times n}$ has the following form

$$A = \begin{bmatrix} 1+2r & -r & 0 & 0 & \dots & 0 \\ -r & 1+2r & -r & 0 & \dots & 0 \\ 0 & -r & 1+2r & -r & & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & & & -r & 1+2r & -r \\ 0 & \dots & \dots & 0 & -r & 1+2r \end{bmatrix}. \quad (23)$$

Stability analysis for implicit scheme

The iterations (22) can be formally written as $u^{m+1} = A^{-1}u^m = (A^{-1})^2 u^{m-1} = \dots (A^{-1})^m u^0$, so the stability of the implicit scheme is now governed by the condition $\rho(A^{-1}) < 1$. The eigenvalues of the matrix (23) are

$$\lambda_j = 1 + 2r(1 - \cos\tfrac{j\pi}{n+1}), \quad j = 1, \dots, n, \quad (24)$$

which are all greater than one, $\lambda_j > 1$. The eigenvalues of A^{-1} are $\{1/\lambda_j\}_{j=1}^n$, thus they all are less than one and greater than zero, $0 < \frac{1}{\lambda_j} < 1$. Hence the stability condition $\rho(A^{-1}) < 1$ is met for every value of $r = D\frac{k}{h^2}$. Such numerical method is referred to as *unconditionally stable*. Analogously the method which is stable only for some parameters is referred to as *conditionally stable*.

Finally, it should be noted here that although we are able to compute approximations using arbitrarily long time steps, the issue of accuracy has not been discussed.