Implicit Scheme for the Heat Equation

Implicit scheme for the one-dimensional heat equation

Once again we consider the one-dimensional heat equation where we seek a $u(x,t)$ satisfying

$$\begin{aligned} u_t &= \nu u_{xx} + f(x,t) \quad (x,t) \in (0,1) \times (0,T] \\ u(x,0) &= u_0(x) \quad x \in [0,1] \\ u(0,t) &= 0 \quad t \in [0,T] \\ u(1,t) &= 0 \quad t \in [0,T] \end{aligned} \tag{1}$$

When we derived the explicit scheme we used a forward difference approximation for the time derivative u_t. Let's see what happens if we use a backward difference approximation.

We write the scheme at the point (x_i, t^n) so that the difference equation now becomes

$$\frac{U_i^n - U_i^{n-1}}{\Delta t} = \nu \frac{U_{i+1}^n - 2U_i^n + U_{i-1}^n}{(\Delta x)^2} + f(x_i, t^n) \quad \text{for } i = 1, \ldots, M-1$$

Now when we simplify this expression, we see that only U_i^{n-1} is known and we move it to the right hand side. The equation then becomes

$$-\lambda U_{i+1}^n + (1+2\lambda)U_i^n - \lambda U_{i-1}^n = U_i^{n-1} + \Delta t f(x_i, t^n)$$

where once again $\lambda = \nu \Delta t / (\Delta x)^2$. We write our scheme as

$$\begin{aligned} \text{Let } U_i^0 &= u_0(x_i) \quad i = 0, 1, \ldots, M \\ \text{For } n = 0, 1, 2, \ldots & \\ -\lambda U_{i+1}^n + (1+2\lambda)U_i^n - \lambda U_{i-1}^n &= U_i^{n-1} + \Delta t f(x_i, t^n) \quad \text{for } i = 1, \ldots, M-1 \\ U_0^{n+1} &= 0 \\ U_M^{n+1} &= 0 \end{aligned} \tag{2}$$

We can no longer solve for U_1^n and then U_2^n, etc. All of the values U_1^n, $U_2^n \dots U_{M-1}^n$ are coupled. We must solve for all of them at once. This requires us to solve a linear system at each timestep and so we call the method implicit.

Writing the difference equation as a linear system we arrive at the following tridiagonal system

$$\begin{pmatrix} 1+2\lambda & -\lambda & & & \\ -\lambda & 1+2\lambda & -\lambda & & \\ & \ddots & \ddots & \ddots & \\ & & -\lambda & 1+2\lambda & -\lambda \\ & & & -\lambda & 1+2\lambda \end{pmatrix} \begin{pmatrix} U_1^n \\ U_2^n \\ \vdots \\ U_{M-2}^n \\ U_{M-1}^n \end{pmatrix} = \begin{pmatrix} \Delta t f(x_1, t^n) + U_1^{n-1} \\ \Delta t f(x_2, t^n) + U_2^{n-1} \\ \vdots \\ \Delta t f(x_{M-2}, t^n) + U_{M-2}^{n-1} \\ \Delta t f(x_{M-1}, t^n) + U_{M-1}^{n-1} \end{pmatrix} \tag{3}$$

Remark: Note that if we had inhomogeneous Dirichlet boundary conditions then we would have to include additional terms in the right hand side.

Exercise Write down the right hand side for the difference equations when we solve the following IBVP using this implicit scheme.

$$\begin{aligned} u_t &= \nu u_{xx} + f(x,t) \quad (x,t) \in (0,1) \times (0,T] \\ u(x,0) &= u_0(x) \quad x \in [0,1] \\ u(0,t) &= 2 \quad t \in [0,T] \\ u(1,t) &= -1 \quad t \in [0,T] \end{aligned}$$

What are the properties of this matrix?

Clearly the matrix is tridiagonal and symmetric.

Exercise Show that the matrix is invertible.

How do we efficiently solve this system of equations?

First note that the coefficient matrix remains the same for all timesteps if we keep the timestep fixed. Consequently all we have to do is factor the matrix once and at each timestep perform a backsolve and a forward solve. In particular if we write the system as $Ax = f$ then A can be factored as

$$A = LL^T$$

and thus we can solve the system as

$$
\begin{aligned}
Ly &= f \\
L^T x &= y
\end{aligned}
$$

The second advantage is that the matrix is tridiagonal and so the system is “cheap” to solve. To factor a general symmetric $n \times n$ tridiagonal matrix A we write $A = LL^T$ where L is a lower triangular (actually bidiagonal) matrix and L^T is its transpose. Specifically we have

$$
\begin{pmatrix}
a_1 & b_1 & & & \\
b_1 & a_2 & b_2 & & \\
& \ddots & \ddots & \ddots & \\
& & b_{n-2} & a_{n-1} & b_{n-1} \\
& & & b_{n-1} & a_n
\end{pmatrix}
=
\begin{pmatrix}
\alpha_1 & & & & \\
\beta_1 & \alpha_2 & & & \\
& \ddots & \ddots & & \\
& & \beta_{n-2} & \alpha_{n-1} & \\
& & & \beta_{n-1} & \alpha_n
\end{pmatrix}
\begin{pmatrix}
\alpha_1 & \beta_1 & & & \\
& \alpha_2 & \beta_2 & & \\
& \ddots & \ddots & \ddots & \\
& & & \alpha_{n-1} & \beta_{n-1} \\
& & & & \alpha_n
\end{pmatrix}
\tag{4}
$$

where the α_i, β_i can be found from the formulas

$$
\alpha_1 = \sqrt{a_1}
$$

and for $i = 2, \ldots, n$

$$
\begin{aligned}
\beta_{i-1} &= \frac{b_{i-1}}{\alpha_{i-1}} \\
\alpha_i &= a_i - \beta_{i-1}^2
\end{aligned}
$$

The solution of the forward solve $Ly = f$ is given by the equations

$$
y_1 = \frac{f_1}{\alpha_1}, \qquad \text{for } i = 2, \ldots, n \quad y_i = \frac{f_i - \beta_{i-1} y_{i-1}}{\alpha_i}
$$

and the backward solve $L^T x = y$ is given by

$$
x_n = \frac{y_n}{\alpha_n} \qquad \text{for } i = n-1, \ldots, 1 \quad x_i = \frac{y_i - \beta_i x_{i+1}}{\alpha_i}
$$

Remark: For clarity we have used separate notation for the a_i and α_i, etc. but when the algorithm is implemented we simply overwrite each a_i with the α_i and similarly for the b_i, x_i. Consequently our total storage is four vectors.

Analysis of the scheme

We expect this implicit scheme to be order $(2,1)$ accurate, i.e., $\mathcal{O}(\Delta x^2 + \Delta t)$. Substitution of the exact solution into the differential equation will demonstrate the consistency of the scheme for the inhomogeneous heat equation and give the accuracy. We perform this computation here is to illustrate two differences from the consistency analysis of our explicit scheme. The first is to demonstrate consistency in the norm. Pointwise consistency is demonstrated identically to the case of the explicit scheme. The second is to illustrate how one handles the forcing function $f(x,t)$ in the analysis.

$$
\begin{aligned}
\text{remainder} &= -\lambda u(x_{i+1},t^n) + (1+2\lambda)u(x_i,t^n) - \lambda u(x_{i-1},t^n) - u(x_i,t^{n-1}) - \Delta t f(x_i,t^n) \\
&= -\lambda\Big[u(x_i,t^n) + \Delta x\ u_x(x_i,t^n) + \frac{(\Delta x)^2}{2!}u_{xx}(x_i,t^n) + \frac{(\Delta x)^3}{3!}u_{xxx}(x_i,t^n) + \frac{(\Delta x)^4}{4!}u_{xxxx}(\Theta_1,t^n)\Big] \\
&\quad +(1+2\lambda)u(x_i,t^n) \\
&\quad -\lambda\Big[u(x_i,t^n) - \Delta x\ u_x(x_i,t^n) + \frac{(\Delta x)^2}{2!}u_{xx}(x_i,t^n) - \frac{(\Delta x)^3}{3!}u_{xxx}(x_i,t^n) + \frac{(\Delta x)^4}{4!}u_{xxxx}(\Theta_2,t^n)\Big] \\
&\quad -\Big[u(x_i,t^n) - \Delta t u_t(x_i,t^n) + \frac{(\Delta t)^2}{2!}u_{tt}(x_i,\tau^n)\Big] - \Delta t f(x_i,t^n) \\
&= -\lambda(\Delta x)^2 u_{xx}(x_i,t^n) + \Delta t u_t(x_i,t^n) - \lambda\frac{(\Delta x)^4}{4!}u_{xxxx}(\Theta_1,t^n) - \lambda\frac{(\Delta x)^4}{4!}u_{xxxx}(\Theta_2 t^n) - \frac{(\Delta t)^2}{2!}u_{tt}(x_i,\tau^n) - \Delta t f(x_i,t^n)
\end{aligned}
$$

Now $u(x,t)$ satisfies the inhomogeneous DE $u_t = \nu u_{xx} + f$ so that

$$
\begin{aligned}
\text{remainder} &= -\frac{\nu\Delta t}{(\Delta x)^2}(\Delta x)^2 u_{xx}(x_i,t^n) + \Delta t(\nu u_{xx}(x_i,t^n) + f(x_i,t^n)) \\
&\quad -\frac{\nu\Delta t}{(\Delta x)^2}\frac{(\Delta x)^4}{4!}\Big[u_{xxxx}(\Theta_1,t^n) + u_{xxxx}(\Theta_2 t^n)\Big] - \frac{(\Delta t)^2}{2!}u_{tt}(x_i,\tau^n) - \Delta t f(x_i,t^n) \\
&= -\frac{\nu\Delta t(\Delta x)^2}{4!}\Big[u_{xxxx}(\Theta_1,t^n) + u_{xxxx}(\Theta_2 t^n)\Big] - \frac{(\Delta t)^2}{2!}u_{tt}(x_i,\tau^n)
\end{aligned}
$$

So

$$|\text{remainder}| = \Delta t\mathcal{O}\big((\Delta x)^2 + \Delta t\big)$$

and we have pointwise consistency.

In order to show consistency in the norm we need to go a step further. Our definition states that a 2-level scheme which we write as

$$\vec{U}^{n+1} = Q\vec{U}^n + \Delta t \vec{F}^n \tag{5}$$

is consistent provided u satisfies

$$\vec{u}^{n+1} = Q\vec{u}^n + \Delta t \vec{F}^n + \Delta t \vec{\tau}^n$$

where $\|\vec{\tau}^n\| \to 0$ as $\Delta x, \Delta t \to 0$.

Our first goal is to write our difference scheme in the general form (5). We have that

$$A\vec{U}^{n+1} = I\vec{U}^n + \Delta t \vec{F}^n$$

where I is the identity matrix and A is the tridiagonal matrix we derived for the scheme. Now to put this equation into the form of (5) we need to write

$$\vec{U}^{n+1} = A^{-1}\vec{U}^n + \Delta t A^{-1} \vec{F}^n$$

Comparing this to (5) we see that $Q = A^{-1}$ which is well defined since we know that A is invertible.

From our previous calculations for pointwise consistency we then know that

$$\vec{u}^{n+1} = A^{-1}\vec{u}^n + \Delta t A^{-1} \vec{F}^n + A^{-1}\Delta t \vec{r}^n$$

where $\vec{r}^n$ is the residual vector where we know each entry is $\Delta t \mathcal{O}((\Delta x)^2 + \Delta t)$.

Consequently, $\vec{\tau}^n = A^{-1}\vec{r}^n$ and so we must simply show that A^{-1} is bounded to get the desired result since

$$\|\vec{\tau}^n\| = \left\|A^{-1}\vec{r}^n\right\| \le \left\|A^{-1}\right\| \|\vec{r}^n\|$$

Now if we choose $\|\cdot\|_\infty$ then we know that $\|\vec{r}^n\|_\infty$ is bounded (provided of course that the derivatives u_{tt} and u_{xxxx} are bounded) and

$$1 + 3\lambda le\|A\|_\infty \le 1 + 4\lambda$$

since the infinity norm can be computed by taking the maximum row sum of a matrix. Since $\|A\|_\infty$ is bounded below we know that $\|A^{-1}\|_\infty$ is bounded and our result follows.

What is different in the implicit scheme from the explicit scheme we investigated is the stability.

Our implicit scheme is UNCONDITIONALLY STABLE

Remark: This means that we no longer have a restriction on our choice of Δt.

Remark: The stability of this scheme is not easy to demonstrate using the technique we employed with the explicit scheme. Consequently, we will delay the stability analysis until we learn Fourier stability analysis.

Remark: Since this is a 2-level scheme in time we can use the Lax Theorem to guarantee convergence (once we show the scheme is stable).

How would our scheme change if we were solving the 2-D heat equation?

Consider finding $u(x,y,t)$ satisfying

$$\begin{aligned} u_t &= \nu(u_{xx}+u_{yy}) + f(x,y,t) \quad (x,t) \in (0,1)\times(0,1)\times(0,T] \\ u(x,y,0) &= u_0(x,y) \\ u(0,y,t) &= 0 \qquad u(1,y,t)=0 \\ u(x,0,t) &= 0 \qquad u(x,1,t)=0 \end{aligned} \tag{6}$$

Let $U_{ij}^n \approx u(x_i,y_j,t^n)$. Our difference equation then becomes

$$\frac{U_{ij}^n - U_{ij}^{n-1}}{\Delta t} = \nu\left(\frac{U_{i+1,j}^n - 2U_{ij}^n + U_{i-1,j}^n}{(\Delta x)^2} + \frac{U_{i,j+1}^n - 2U_{ij}^n + U_{i,j-1}^n}{(\Delta y)^2}\right) + f(x_i,y_j,t^n)$$

for $i = 1,\ldots,M-1$ and $j = 1,\ldots,M-1$.

For simplicity of exposition, let's assume that $\Delta x = \Delta y$. Then simplifying the expression we arrive at

$$-\lambda U_{i+1,j}^n + (1+4\lambda)U_{ij}^n - \lambda U_{i-1,j}^n - \lambda U_{i,j+1}^n - \lambda U_{i,j-1}^n = U_{ij}^{n-1} + \Delta t f(x_i,t^n)$$

If we number the unknowns as U_{11}^n, U_{12}^n, $\cdots$, $U_{1,M-1}^n$, $U_{2,1}^n$, etc. then the linear system we must solve is a block tridiagonal matrix of the form

$$\begin{pmatrix} A & -\lambda I & & \\ -\lambda I & A & -\lambda I & \\ & \ddots & \ddots & \ddots \\ & & -\lambda I & A \end{pmatrix}$$

Each block is $(M-1)\times(M-1)$; here I represents the identity matrix and A is given by

$$\begin{pmatrix} 1+4\lambda & -\lambda & & & \\ -\lambda & 1+4\lambda & -\lambda & & \\ & \ddots & \ddots & \ddots & \\ & & & -\lambda & 1+4\lambda \end{pmatrix}$$

We have written the vector of unknowns and right hand side as

$$\begin{pmatrix} U^n_{11} \\ \vdots \\ U^n_{1,M-1} \\ ------- \\ U^n_{21} \\ \vdots \\ U^n_{2,M-1} \\ ------- \\ \vdots \\ ------- \\ U^n_{M-1,1} \\ \vdots \\ U^n_{M-1,M-1} \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} U^{n-1}_{11} + \Delta t f(x_1, y_1, t^n) \\ \vdots \\ U^{n-1}_{1,M-1} + \Delta t f(x_1, y_{M-1}, t^n) \\ ---------- \\ U^{n-1}_{21} + \Delta t f(x_2, y_1, t^n) \\ \vdots \\ U^{n-1}_{2,M-1} + \Delta t f(x_2, y_{M-1}, t^n) \\ ---------- \\ \vdots \\ ---------- \\ U^{n-1}_{M-1,1} + \Delta t f(x_{M-1}, 1, t^n) \\ \vdots \\ U^{n-1}_{M-1,M-1} + \Delta t f(x_{m-1}, y_{M-1}, t^n) \end{pmatrix}$$

Remark: Of course if our problem had inhomogeneous Dirichlet boundary data then we would have had some additional terms added onto the boundary.

Exercise Write out the right hand side for the implicit difference scheme approximating the IBVP

$$
\begin{aligned}
u_t &= \nu(u_{xx} + u_{yy}) \quad (x,t) \in (0,1) \times (0,1) \times (0,T] \\
u(x,y,0) &= u_0(x) \quad x \in [0,1] \\
u(0,y,t) &= 0 \quad t \in [0,T] \\
u(1,y,t) &= 0 \\
u(x,0,t &= 4 \\
u(x,1,t) &= -5
\end{aligned}
$$

Once again our coefficient matrix is symmetric and positive definite so we can perform an LL^T decomposition and solve as before.

A general $n \times n$ symmetric block tridiagonal matrix can be efficiently factored as

$$
\begin{pmatrix}
A_1 & B_1 & & \\
B_1 & A_2 & B_2 & \\
 & \ddots & \ddots & \ddots \\
 & & B_{n-1} & A_n
\end{pmatrix}
=
\begin{pmatrix}
C_1 & & & \\
D_1 & C_2 & & \\
 & \ddots & \ddots & \\
 & & D_{n-1} & C_n
\end{pmatrix}
\begin{pmatrix}
C_1^T & D_1^T & & \\
 & C_2^T & D_2^T & \\
 & & \ddots & \ddots \\
 & & & C_n^T
\end{pmatrix}
$$

where

$$
A_1 = C_1 C_1^T \qquad C_1 D_1^T = B_1, \quad \text{etc.}
$$

In our case A_1, etc., are tridiagonal matrices so we simply perform an LL^T decomposition to determine C_1. For determining the D_i we simply solve a lower triangular systems.