Numerical Methods in Engineering and Applied Science

Lecture 19. Linear parabolic equations.

Heat equation in 1D.

Consider the 1D equation with $\Omega = (0,1)$: find $u = u(x,t)$ for $(x,t) \in (0,1) \times (0,T)$ such that

$$\begin{aligned} & u_t - \kappa u_{xx} = 0 \quad \text{dans} \quad (0,1) \times (0,T) \\ & u(0,t) = g_0(t), \quad t \in (0,T) \\ & u(1,t) = g_1(t), \quad t \in (0,T) \\ & u(x,0) = \eta(x), \quad x \in (0,1) \end{aligned} \tag{1}$$

Suppose $\kappa = 1$.

For two given integers m and N, we discretize the intervals of space $[0,1]$ and time $[0,T]$ by introducing points

$$\begin{aligned} & x_i = ih, \quad = 0, ..., m+1 \\ & t_n = nk, \quad n = 0, ..., N \end{aligned} \tag{2}$$

where $h = 1/(m+1)$ is the step of spatial discretization and $k = T/N$ is the time step.

We denote by $U_i^n \approx u(x_i, t_n)$ the numerical approximation to (x_i, t_n). Since the heat equation is an evolution equation that is solved with an initial condition at $t = 0$, it is natural to determine U_i^{n+1} at all i using the values of U_i^n and perhaps previous time steps.

If we use central finite differences in space and explicit Euler method in time, we obtain an *explicit* scheme,

$$\frac{U_i^{n+1} - U_i^n}{k} = \frac{1}{h^2}(U_{i-1}^n - 2U_i^n + U_{i+1}^n). \tag{3}$$

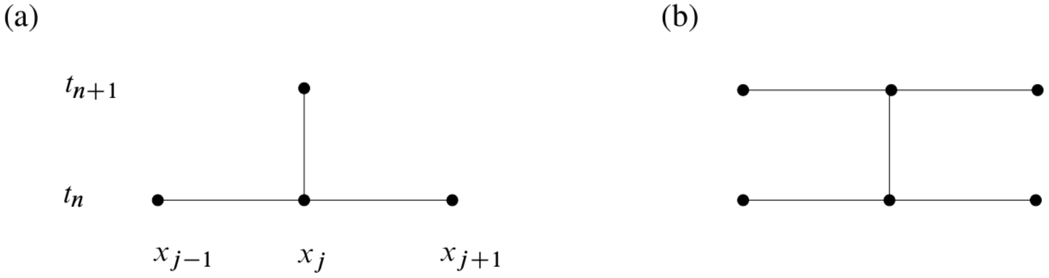
Consider the local truncation error of this scheme,

$$\begin{aligned} \tau(x,t) &= \frac{u(x,t+k) - u(x,t)}{k} - \frac{1}{h^2}(u(x-h,t) - 2u(x,t) + u(x+h,t)) \\ &= \left(u_t + \frac{1}{2}ku_{tt} + \frac{1}{6}k^2 u_{ttt} + ... \right) - \left(u_{xx} + \frac{1}{12}h^2 u_{xxxx} + ... \right). \end{aligned}$$

Since $u_t = u_{xx}$, the terms in $\mathcal{O}(1)$ cancel each other out, and $u_{tt} = u_{txx} = u_{xxxx}$. We obtain

$$\tau(x,t) = \left(\frac{1}{2}k - \frac{1}{12}h^2\right) u_{xxxx} + \mathcal{O}(k^2 + h^4).$$

The method is of order 2 in space and of order 1 in time.



An *implicit* method often used to solve the heat equation is the *Crank–Nicolson* method,

$$\frac{U_i^{n+1} - U_i^n}{k} = \frac{U_{i-1}^n - 2U_i^n + U_{i+1}^n}{2h^2} + \frac{U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i+1}^{n+1}}{2h^2}.$$

We can write this scheme in the form

$$-rU_{i-1}^{n+1} + (1+2r)U_i^{n+1} - rU_{i+1}^{n+1} = rU_{i-1}^n + (1-2r)U_i^n + rU_{i+1}^n,$$

where $r = k/2h^2$. In the matrix form, we have

$$\begin{bmatrix} (1+2r) & -r & & & & \\ -r & (1+2r) & -r & & & \\ & -r & (1+2r) & -r & & \\ & & \ddots & \ddots & \ddots & \\ & & & -r & (1+2r) & -r \\ & & & & -r & (1+2r) \end{bmatrix} \begin{bmatrix} U_1^{n+1} \\ U_2^{n+1} \\ U_3^{n+1} \\ \vdots \\ U_{m-1}^{n+1} \\ U_m^{n+1} \end{bmatrix}$$

$$= \begin{bmatrix} r(g_0(t_n) + g_0(t_{n+1})) + (1-2r)U_1^n + rU_2^n \\ rU_1^n + (1-2r)U_2^n + rU_3^n \\ rU_2^n + (1-2r)U_3^n + rU_4^n \\ \vdots \\ rU_{m-2}^n + (1-2r)U_{m-1}^n + rU_m^n \\ rU_{m-1}^n + (1-2r)U_m^n + r(g_1(t_n) + g_1(t_{n+1})) \end{bmatrix}.$$

Note the boundary conditions $u(0,t) = g_0(t)$ and $u(1,t) = g_1(t)$. Since the solution of a tridiagonal system requires $\mathcal{O}(m)$ operations, the number of operations per time step is of the same order as for an explicit method.

To compute the local truncation error of this scheme, we note that

$$\begin{aligned}
\frac{\partial u}{\partial t}(x, t+k/2) &= \frac{u(x,t+k)-u(x,t)}{k} + \mathcal{O}(k^2) \\
\frac{\partial^2 u}{\partial x^2}(x, t+k/2) &= \frac{1}{2}\left(\frac{\partial^2 u}{\partial x^2}(x,t) + \frac{\partial^2 u}{\partial x^2}(x,t+k)\right) + \mathcal{O}(k^2).
\end{aligned}$$

It follows that

$$\tau = (L_{h,k}u - Lu)(x, t+k/2) = \mathcal{O}(k^2 + h^2).$$

The Crank–Nicolson scheme is of order 2 in space and time.

To make the link between the stability of numerical methods for parabolic PDEs and the stability theory developed for ODEs, we use the *method of lines*. We only discretize with respect to x and we obtain a system of ODEs, one equation of which corresponds to one point of the discretization grid.

For example, for the heat equation we can obtain

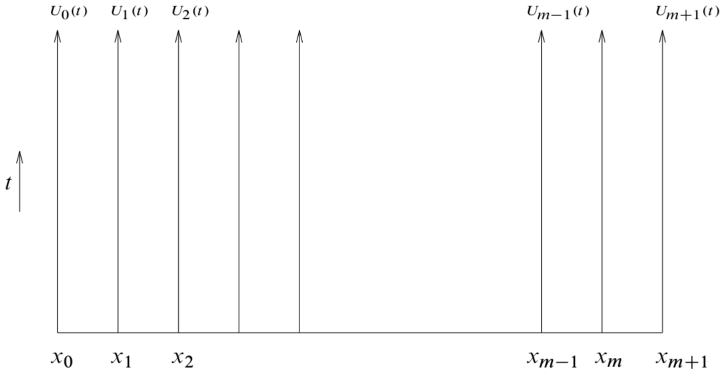
$$\frac{d}{dt}U_i(t) = \frac{1}{h^2}(U_{i-1}(t) - 2U_i(t) + U_{i+1}(t)) \quad \text{for} \quad i = 1, 2, ..., m,$$

We can rewrite this system in the form

$$\frac{d}{dt}U(t) = AU(t) + g(t),$$

$$A = \frac{1}{h^2}\begin{bmatrix} -2 & 1 & & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & & 1 & -2 \end{bmatrix}, \qquad g(t) = \frac{1}{h^2}\begin{bmatrix} g_0(t) \\ 0 \\ 0 \\ \vdots \\ 0 \\ g_1(t) \end{bmatrix}.$$

$U_0(t) \equiv g_0(t)$ and $U_{m+1}(t) \equiv g_1(t)$.Then, we use methods developed for ODEs to integrate with respect to time.



For example, if we use the explicit Euler method, we get

$$\frac{U^{n+1} - U^n}{k} = AU^n + g^n.$$

If we use the implicit trapezoidal rule, we obtain the Crank–Nicolson scheme,

$$\frac{U^{n+1} - U^n}{k} = \frac{(AU^{n+1} + g^{n+1}) + (AU^n + g^n)}{2}.$$

We can study the stability of these methods using the same approach as for the ODE systems. Note that A is diagonalizable. The method is stable if $k\lambda \in \mathcal{D}$ for every λ eigenvalue of A, where k is the time step and $\mathcal{D}$ is the stability domain of the temporal scheme.

The eigenvalues of the matrix A are

$$\lambda_p = \frac{2}{h^2}(\cos(p\pi h) - 1) \quad \text{for} \quad p = 1, 2, ..., m.$$

Note that the eigenvalues of A depend on the spatial discretization step h. They are all real negatives. The furthest value from zero is $\lambda_m \approx -4/h^2$. Then the stability condition is $-4k/h^2 \in \mathcal{D}$ (if the stability interval includes zero).

For example, for *Euler method* we have the condition $|1+k\lambda| \leq 1$ which gives $-2 \leq -4k/h^2 \leq 0$. We obtain

$$\frac{k}{h^2} \leq \frac{1}{2}.$$

Since the stability domain of the implicit trapezoidal rule is the entire real negative half-plane, and the eigenvalues of A are negative, *the Crank-Nicolson method* is unconditionally stable.

For precision reasons, we generally do calculations with $k = \mathcal{O}(h)$, since the method is second order in time and space.

The ODE system obtained for the heat equation is stiff: $\lambda_1 = -\pi^2$ and $\lambda_m \approx -4/h^2$, the ratio of eigenvalues $\lambda_m/\lambda_1 \approx 4/\pi^2 h^2 \to \infty$ is unbounded in the limit of $h \to 0$.

Convergence of finite difference methods for parabolic problems.

We will study the convergence of the numerical solutions obtained with the discretization steps (k_1, h_1), (k_2, h_2), ..., where $k_j \to 0$ and $h_j \to 0$. In general, k_j and h_j must satisfy a certain relation. For example, for the explicit Euler method in time and second-order finite differences in space, the approximation is convergent only if $k_j/h_j^2 \leq 1/2$, then we can assume $k = Ch^2$, $C < 1/2$, to study convergence in limit $k \to 0$.

We consider schemes written in the form

$$U^{n+1} = B(k)U^n + b^n(k) \tag{4}$$

where $B(k) \in \mathbb{R}^{m\times m}$ and $b^n(k) \in \mathbb{R}^m$ on the grid with $h = 1/(m+1)$. In general, B and b^n depend on h and k, but we specify $h = h(k)$ to study the limit in $k \to 0$.

For example, for the explicit Euler scheme we obtain

$$B(k) = I + kA,$$

where A is the tridiagonal finite difference matrix. For the Crank–Nicolson scheme we get

$$B(k) = \left(I - \frac{k}{2}A\right)^{-1}\left(I + \frac{k}{2}A\right). \tag{5}$$

The method is *stable in the sense of Lax–Richtmyer* if, for all T, there exists a constant C_T such that

$$||B(k)^n|| \leq C_T \tag{6}$$

for all $k \geq 0$ and n such that $kn \leq T$.

Lax Theorem Stability in the sense of Lax–Richtmyer is a necessary and sufficient condition to ensure the convergence of a consistent linear method (4).

The numerical solution satisfies

$$U^{n+1} = BU^n + b^n \tag{7}$$

For the exact solution $u(x,t)$, we can write

$$u^{n+1} = Bu^n + b^n + k\tau^n, \tag{8}$$

where

$$u^n = \begin{pmatrix} u(x_1,t_n) \\ u(x_2,t_n) \\ \vdots \\ u(x_m,t_n) \end{pmatrix}, \qquad \tau^n = \begin{pmatrix} \tau(x_1,t_n) \\ \tau(x_2,t_n) \\ \vdots \\ \tau(x_m,t_n) \end{pmatrix}.$$

We define the global error $E^n = U^n - u^n$. We subtract (8) from (7) and we get

$$E^{n+1} = BE^n - k\tau^n.$$

After N time steps, we have

$$E^N = B^N E^0 - k \sum_{n=1}^{N} B^{N-n} \tau^{n-1},$$

and we obtain

$$||E^N|| \le ||B^N|| ||E^0|| + k \sum_{n=1}^{N} ||B^{N-n}|| ||\tau^{n-1}||.$$

If the method is stable in the Lax–Richtmyer sense, then for $Nk \le T$

$$||E^N|| \le C_T ||E^0|| + T C_T \max_{1 \le n \le N} ||\tau^{n-1}|| \to 0 \quad \text{in the limit of} \quad k \to 0.$$

For example, for the explicit Euler method,

$$||B||_2 = \max_p(|\lambda_p(B)|) = \max_p(|\lambda_p(I + kA)|) = \max_p(|1 + k\lambda_p(A)|)$$
$$= \max_p(|1 + k\frac{2}{h^2}(\cos(p\pi h) - 1)|).$$

If the condition $k/h^2 \leq 1/2$ holds, we see that $|1 + k\frac{2}{h^2}(\cos(p\pi h) - 1)| \leq 1$ therefore $||B||_2 \leq 1$. We conclude that, under this condition, the method is stable in the sense of Lax–Richtmyer and therefore convergent.

For the Crank–Nicolson method, the eigenvalues of the matrix B are

$$\frac{1 + k\lambda_p(A)/2}{1 - k\lambda_p(A)/2} \quad \text{where} \quad \lambda_p(A) = \frac{2}{h^2}(\cos(p\pi h) - 1) < 0.$$

Therefore, $||B||_2 \leq 1$. This method converges for all $k > 0$.

The condition $||B|| \leq 1$ (*strong stability*) is sufficient for Lax–Richtmyer stability, but not necessary. If there is a constant α such that

$$||B|| \leq 1 + \alpha k$$

for sufficiently small k. In that case,

$$||B^n|| \leq (1 + \alpha k)^n \leq e^{\alpha T}$$

for $nk \leq T$. Recall that the matrix B depends on h and k and its size $m = \mathcal{O}(1/h)$ increases with $k, h \to 0$.

An important difference between the stability theory of numerical methods for PDEs and that for ODEs is the dependence of the eigenvalues $\lambda(B)$ on the time step k.
For ODEs, λ does not depend on k. For convergence, it is necessary to have $k\lambda$ inside the stability domain. This condition is verified for k sufficiently small, so it suffices to study the zero-stability (stability of the scheme with $k = 0$).
For PDEs, λ depends on k and h. For convergence, the values of $k\lambda$ must remain inside the stability domain.

Another difference is that, for the ODEs $u'(t) = f(u(t))$ we know how to prove convergence for any Lipschitz function $f(u)$. Lax's theorem for PDEs applies only to linear schemes which are obtained, generally, by discretizing linear

PDEs. For example, in the equation $u_t = u_{xx}$ we replace u_{xx} by $f(U) = AU$. This function is Lipschitz continuous but the Lipschitz constant $||A|| \sim 1/h^2$.

Von Neumann stability.

Stability analysis in the sense of Von Neumann consists in studying the effect of the scheme on an isolated Fourier mode. For the Fourier mode in question to be a solution of the continuous problem, we consider *the Cauchy problem* where the domain is infinite, $-\infty < x < \infty$. We can also consider the problem with *periodic boundary conditions*, for example, $0 \leq x \leq 1$ and $u(0,t) = u(1,t)$. It is equivalent to a Cauchy problem with a periodic initial condition.

We know how to solve the Cauchy problem for linear PDEs with constant coefficients by using the Fourier transform. The functions $e^{\iota\xi x}$, where $\xi = \text{const}$ is the wave number, are the eigenfunctions of the operator ∂_x,

$$\partial_x e^{\iota\xi x} = \iota\xi e^{\iota\xi x}.$$

Then they are also the eigenfunctions of any linear differential operator with constant coefficients.

We use the fact that the function $W_j = e^{\iota jh\xi}$ is a eigenfunction of any translation invariant finite difference operator. For example, if we approach $w'(x_j)$ by $D_0 W_j = \frac{1}{2h}(W_{j+1} - W_{j-1})$, we obtain

$$\begin{aligned} D_0 W_j &= \frac{1}{2h}\left(e^{\iota(j+1)h\xi} - e^{\iota(j-1)h\xi}\right) = \frac{1}{2h}\left(e^{\iota h\xi} - e^{-\iota h\xi}\right)e^{\iota jh\xi} \\ &= \frac{\iota}{h}\sin(h\xi)e^{\iota jh\xi} = \frac{\iota}{h}\sin(h\xi)W_j. \end{aligned}$$

W is an eigenfunction of D_0 and $\frac{\iota}{h}\sin(h\xi)$ is the eigenvalue.

Consider V_j defined at $x_j = jh$, where $j = 0, \pm 1, \pm 2, ...$ We assume that the norm $\left(h \sum_{j=-\infty}^{\infty} |V_j|^2\right)^{1/2}$ is finite. We can write V_j as a linear combination of the functions $e^{\iota jh\xi}$ for all $\xi \in [-\pi/h \leq \xi \leq \ ft/h]$. We write

$$V_j = \frac{1}{\sqrt{2\pi}} \int_{-\pi/h}^{\pi/h} \hat{V}(\xi) e^{\iota jh\xi} d\xi, \quad \text{where} \quad \hat{V}(\xi) = \frac{h}{\sqrt{2\pi}} \sum_{j=-\infty}^{\infty} V_j e^{-\iota jh\xi}.$$

We will use *Parseval equality*,

$$||\hat{V}||_2 = ||V||_2,$$

where

$$||V||_2 = \left(h \sum_{j=-\infty}^{\infty} |V_j|^2\right)^{1/2}, \quad ||\hat{V}||_2 = \left(\int_{-\pi/h}^{\pi/h} |\hat{V}(\xi)|^2 d\xi\right)^{1/2}.$$

To demonstrate the stability of a scheme, we must show that $||B||_2 \leq 1 + \alpha k$. There must be a constant α such that

$$||U^{n+1}||_2 \leq (1 + \alpha k)||U^n||_2$$

for all U^n. By Parseval equality, it is equivalent to

$$||\hat{U}^{n+1}||_2 \leq (1 + \alpha k)||\hat{U}^n||_2$$

For finite difference methods and one-step time schemes, we obtain the recurrence

$$\hat{U}^{n+1}(\xi) = g(\xi)\hat{U}^n(\xi),$$

where *the amplification factor* depends on the method. If we show that

$$|g(\xi) \leq 1 + \alpha k|,$$

where α is independent of ξ, we deduce that the method is stable, because if

$$|\hat{U}^{n+1}(\xi)| \leq (1 + \alpha k)|\hat{U}^n(\xi)| \quad \text{for all } \xi,$$

then

$$||U^{n+1}||_2 \leq (1+\alpha k)||U^n||_2.$$

Example. Consider the method

$$U_i^{n+1} = U_i^n + \frac{k}{h^2}(U_{i-1}^n - 2U_i^n + U_{i+1}^n).$$

For the Von Neumann analysis, we assume

$$U_j^n = e^{\iota jh\xi} \quad \text{et} \quad U_j^{n+1} = g(\xi)e^{\iota jh\xi}.$$

We substitute U_j^n and U_j^{n+1} in the diagram,

$$\begin{aligned} g(\xi)e^{\iota jh\xi} &= e^{\iota jh\xi} + \tfrac{k}{h^2}\left(e^{\iota(j-1)h\xi} - 2e^{\iota jh\xi} + e^{\iota(j+1)h\xi}\right) \\ &= \left(1 + \tfrac{k}{h^2}\left(e^{-\iota h\xi} - 2 + e^{\iota h\xi}\right)\right)e^{\iota jh\xi}, \end{aligned}$$

$$g(\xi) = 1 + 2\frac{k}{h^2}(\cos(\xi h) - 1).$$

We see that

$$1 - 4\frac{k}{h^2} \leq g(\xi) \leq 1$$

for all ξ. To guarantee $|g(\xi)| \leq 1$, it is necessary to verify the condition

$$4\frac{k}{h^2} \leq 2.$$

Example. The Crank–Nicolson method,

$$U_i^{n+1} = U_i^n + \frac{k}{2h^2}\left((U_{i-1}^n - 2U_i^n + U_{i+1}^n) + (U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i+1}^{n+1})\right),$$

gives the following equation for the amplification factor $g(\xi)$:

$$g(\xi) = 1 + \frac{k}{2h^2}\left(e^{-\iota\xi h} - 2 + e^{\iota\xi h}\right)(1 + g(\xi)).$$

We obtain

$$g(\xi) = \frac{1 + \frac{1}{2}z}{1 - \frac{1}{2}z} \quad \text{where} \quad z = \frac{k}{h^2}\left(e^{-\iota\xi h} - 2 + e^{\iota\xi h}\right) = \frac{2k}{h^2}(\cos(\xi h) - 1).$$

Since $z \leq 0$ for all ξ, we see that $|g| \leq 1$ and the method is stable for all k and h.

2D heat equation.
In 2D, in Cartesian coordinates, the heat equation takes the form

$$u_t = u_{xx} + u_{yy}.$$

We prescribe the initial condition $u(x, y, 0) = \eta(x, y)$ and the boundary conditions on the boundary of the domain Ω. We can discretize the Laplacian with the 5-point scheme.

$$\nabla_5^2 U_{ij} = \frac{1}{h^2}(U_{i-1,j} + U_{i+1,j} + U_{i,j-1} + U_{i,j+1} - 4U_{ij}).$$

If we choose the trapezoidal method for the temporal discretization, we obtain the Crank–Nicolson method in 2D,

$$U_{ij}^{n+1} = U_{ij}^n + \frac{k}{2}\left[\nabla_5^2 U_{ij}^n + \nabla_5^2 U_{ij}^{n+1}\right].$$

Since the method is implicit, we obtain a system of equations to solve at each time step. We can rewrite it in the form

$$\left(I - \frac{k}{2}\nabla_5^2\right) U_{ij}^{n+1} = \left(I + \frac{k}{2}\nabla_5^2\right) U_{ij}^n$$

The matrix of this system,

$$A = I - \frac{k}{2}\nabla_5^2,$$

has the same structure as $\nabla 5.2$. These eigenvalues are

$$\lambda_{p,q} = 1 - \frac{k}{h^2}\left[(\cos(p\pi h) - 1) + (\cos(q\pi h) - 1)\right],$$

where $p, q = 1, 2, 3, ..., m$. We use iterative methods to solve this system. The spectral radius of A is $\mathcal{O}(k/h^2)$ and the absolute value closest to zero is $1 + \mathcal{O}(k)$. Then the spectral conditioning number

$$\kappa(A) = \mathcal{O}(k/h^2)$$

while the spectral conditioning of ∇_5^2 is $\mathcal{O}(1/h^2)$. Recall that, for reasons of precision, we prescribe $k \sim h$. Better conditioning accelerates the convergence,

compared to the case of the Poisson equation.
Moreover, since $U_{ij}^{n+1} = U_{ij}^n + \mathcal{O}(k)$, we can use U_{ij}^n as the first approximation of $U_{ij}^{[0]}$ for iterations. We can also use the extrapolation,

$$U_{ij}^{[0]} = 2U_{ij}^n - U_{ij}^{n-1}.$$

Since the system is not very badly conditioned and we have a good first approximation, generally it suffices to do a few iterations at each time step.

Locally one-dimensional method (LOD)).

It is a splitting method. A time step is composed of two steps, each step corresponds to a spatial direction,

$$\begin{aligned}
U_{ij}^{*} &= U_{ij}^{n} + \frac{k}{2}(D_x^2 U_{ij}^{n} + D_x^2 U_{ij}^{*}),\\
U_{ij}^{n+1} &= U_{ij}^{*} + \frac{k}{2}(D_y^2 U_{ij}^{*} + D_y^2 U_{ij}^{n+1}).
\end{aligned}$$

In the matrix form,

$$\begin{aligned}
\left(I - \frac{k}{2}D_x^2\right) U^{*} &= \left(I + \frac{k}{2}D_x^2\right) U^{n},\\
\left(I - \frac{k}{2}D_y^2\right) U^{n+1} &= \left(I + \frac{k}{2}D_y^2\right) U^{*},
\end{aligned}$$

The first system contains the unknowns U_{ij}^{*}. For each fixed value of j, the system is decoupled from those for other values of j. We solve $m+2$ systems ($j = 0, 1, ..., m+1$). Each system has m equations and m unknowns U_{ij}^{*}, $i = 1, 2, ..., m$.

The second equation gives m systems $(i = 1, 2, ..., m)$, each of m equations. In sum, at each time step, we have to solve $2m + 2$ tridiagonal systems of size m. With the direct method, the number of operations is $\mathcal{O}(m^2)$, of the same order as for an explicit method.

To solve the systems at the second stage, it is necessary to know the values at the edges of the domain, U^*_{i0}, U^{n+1}_{i0}, $U^*_{i,m+1}$ and $U^{n+1}_{i,m+1}$. It is not trivial if the boundary conditions vary over time. The values at t_{n+1} come from the Dirichlet boundary condition. To compute U^*_{i0} and $U^*_{i,m+1}$, we solve the systems for $j = 0$ and $j = m + 1$ at the first stage.
To solve the systems at the first stage, we need to know U^n_{0j}, U^*_{0j}, $U^n_{m+1,j}$ and $U^*_{m+1,j}$. The values at t_n are known. To calculate U^*_{0j} and $U^*_{m+1,j}$, we can use

the formulas

$$U^*_{0j} = U^{n+1}_{0j} - \frac{k}{h^2}(U^{n+1}_{0,j-1} - 2U^{n+1}_{0j} + U^{n+1}_{0,j+1}),$$
$$U^*_{m+1,j} = U^{n+1}_{m+1,j} - \frac{k}{h^2}(U^{n+1}_{m+1,j-1} - 2U^{n+1}_{m+1,j} + U^{n+1}_{m+1,j+1}),$$

This corresponds to the integration of the anti-diffusion equation. Nevertheless, this method is stable since the integration is done in a single sub-step.
The LOD method is of order 2, unconditionally stable .

The ADI (alternating direction implicit) method.
At each step, the explicit terms contain the derivatives in the direction perpendicular to that of the implicit terms,

$$U^*_{ij} = U^n_{ij} + \frac{k}{2}(D^2_y U^n_{ij} + D^2_x U^*_{ij}),$$
$$U^{n+1}_{ij} = U^*_{ij} + \frac{k}{2}(D^2_x U^*_{ij} + D^2_y U^{n+1}_{ij}).$$

In the matrix form

$$\left(I - \frac{k}{2} D_x^2\right) U^* = \left(I + \frac{k}{2} D_y^2\right) U^n,$$
$$\left(I - \frac{k}{2} D_y^2\right) U^{n+1} = \left(I + \frac{k}{2} D_x^2\right) U^*,$$

At each step, the diffusion is in both directions. An advantage of this method is that, since U^* represents the solution to $t_{n+1/2}$ (order 1 approximation), one can easily evaluate the conditions to the limits. The method is of order 2, unconditionally stable.

Book: Randall J. LeVeque, “Finite Difference Methods for Ordinary and Partial Differential Equations. Steady-State and Time-Dependent Problems” (SIAM)

Example programs in Matlab :

`http://faculty.washington.edu/rjl/fdmbook/`

Also see

`http://www-math.mit.edu/cse/codes/`

```
http://www-math.mit.edu/cse/codes/mit18086_fd_heateqn.m

function mit18086_fd_heateqn
%MIT18086_FD_HEATEQN
%    Solves the heat equation u_t=u_xx by finite differences.
%    Example uses explicit Euler in time, homogeneous
%    Dirichlet b.c. on the left, and homogeneous Neumann b.c.
%    on the right.

% 02/2007 by Benjamin Seibold
%            http://www-math.mit.edu/~seibold/
% Feel free to modify for teaching and learning.

n = 13;         % number of space gridpoints without boundaries
dt = 1e-2;                                          % time step
tf = 2;                                            % final time
x = linspace(-1,1,n+2)';
xi = x(2:end-1);
h = x(2)-x(1);
disp(sprintf('CFL number: %0.2f',dt/h^2))
u0 = f(x);
u = u0(2:end-1);
I = eye(n);
D = 2*I;
```

```
D(end,end) = 1;      % Neumann boundary condition on right side
R = diag(ones(1,n-1),1);
A = (R-D+R')/h^2;
M = I+dt*A;  % change to M = inv(I-dt*A) for implicit time step
for tn = 1:ceil(tf/dt)
u = M*u;
clf
plot(x,u0,'b:',xi,u,'r.-')
title(sprintf('time t=%0.2f',tn*dt))
drawnow
end

function y = f(x)
% initial condition function
y = (1-x.^2).^2;
```