13 Implicit methods for the Heat equation

13.1 Derivation of the Crank–Nicolson scheme

We continue studying numerical methods for the IBVP (12.1)–(12.3). In Sec. 12.3, we have seen that the Heat equation (12.1) could be represented as a coupled system of ODEs (12.15). Moreover, in one of the problems in Homework #12, you were asked whether that system was stiff (and the answer was 'yes, it is stiff'). As we remember from Lecture 5, the way to deal with stiff systems is by using *implicit* methods, which may be constructed so as to be unconditionally stable *irrespective of the step size in the evolution variable*. In Lecture 4, we stated that the highest order that an unconditionally stable method can have is 2 (that is, in our current notations, the error can tend to zero no faster than $O(\kappa^2)$). From Lecture 4 we also recall the particular example of an implicit, unconditionally stable method of order 2: this is the modified implicit Euler method (3.43). In our current notations, it is:

$$\vec{\mathbf{U}}^{n+1} = \vec{\mathbf{U}}^n + \frac{\kappa}{2}\left(\vec{\mathbf{f}}(t_n, \vec{\mathbf{U}}^n) + \vec{\mathbf{f}}(t_{n+1}, \vec{\mathbf{U}}^{n+1})\right) . \tag{13.1}$$

In the case of system (12.15), the form of function $\vec{\mathbf{f}}$ is:

$$f_m(\vec{\mathbf{U}}^n) = \frac{U^n_{m+1} - 2U^n_m + U^n_{m-1}}{h^2} . \tag{13.2}$$

Since the operator on the r.h.s. of the above equation will appear very frequently in the remainder of this course, we introduce a special notation for it:

$$\frac{U_{m+1} - 2U_m + U_{m-1}}{h^2} \equiv \frac{1}{h^2}\delta_x^2 U_m . \tag{13.3}$$

Similarly, we denote

$$\frac{U^{n+1} - U^n}{\kappa} \equiv \frac{1}{\kappa}\delta_t U^n . \tag{13.4}$$

Then Eq. (13.1) with f given by (13.2) takes on the form:

$$\frac{U^{n+1}_m - U^n_m}{\kappa} = \frac{1}{2}\left[\frac{U^n_{m+1} - 2U^n_m + U^n_{m-1}}{h^2} + \frac{U^{n+1}_{m+1} - 2U^{n+1}_m + U^{n+1}_{m-1}}{h^2}\right] , \tag{13.5}$$

or, in the above shorthand notations,

$$\frac{1}{\kappa}\delta_t U^n_m = \frac{1}{2h^2}\left[\delta_x^2 U^n_m + \delta_x^2 U^{n+1}_m\right] . \tag{13.6}$$

The finite-difference equation (13.5) can be rewritten as

$$U^{n+1}_m - \frac{r}{2}\left(U^{n+1}_{m+1} - 2U^{n+1}_m + U^{n+1}_{m-1}\right) = U^n_m + \frac{r}{2}\left(U^n_{m+1} - 2U^n_m + U^n_{m-1}\right) ; \tag{13.7}$$

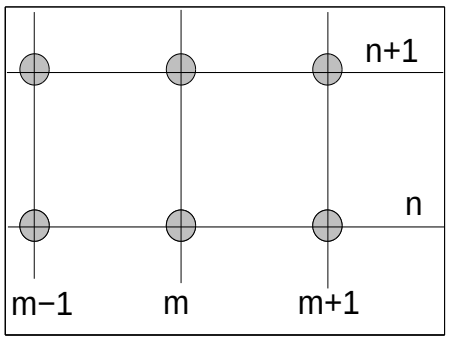
and correspondingly, Eq. (13.6), as

$$\left(1 - \frac{r}{2}\delta_x^2\right) U^{n+1}_m = \left(1 + \frac{r}{2}\delta_x^2\right) U^n_m, \qquad m = 1, \ldots, M-1 , \tag{13.8}$$

where

$$r = \frac{\kappa}{h^2} . \tag{12.13}$$

Scheme (13.7) (or, equivalently, (13.8)) is called the **Crank–Nicolson (CN) method**. Its stencil is shown on the right. Both from the stencil and from the defining equations one can see that U_m^{n+1} *cannot* be determined in isolation. Rather, one has to determine $\vec{\mathbf{U}}$ on the *entire* $(n+1)$*th time level*. Using our standard notation for the solution vector,



$$\vec{\mathbf{U}}^n = \left[U_1^n,\, U_2^n,\, \cdots,\, U_{M-1}^n\right]^T,$$

we rewrite Eq. (13.8) in the vector form:

$$\left(I - \frac{r}{2}A\right)\vec{\mathbf{U}}^{n+1} \;=\; \left(I + \frac{r}{2}A\right)\vec{\mathbf{U}}^n + \vec{\mathbf{b}}\,, \tag{13.9}$$

where I is the unit matrix and

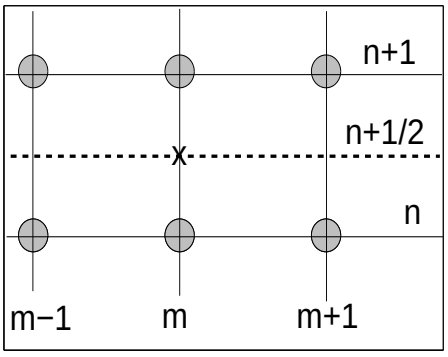
$$A = \begin{pmatrix} -2 & 1 & 0 & \cdot & \cdot & 0 \\ 1 & -2 & 1 & 0 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & 0 & 1 & -2 & 1 \\ 0 & \cdot & \cdot & 0 & 1 & -2 \end{pmatrix} \quad \text{and} \quad \vec{\mathbf{b}} = \frac{r}{2}\begin{pmatrix} U_0^n + U_0^{n+1} \\ 0 \\ \cdot \\ 0 \\ U_M^n + U_M^{n+1} \end{pmatrix} \equiv \frac{r}{2}\begin{pmatrix} g_0(t_n) + g_0(t_{n+1}) \\ 0 \\ \cdot \\ 0 \\ g_1(t_n) + g_1(t_{n+1}) \end{pmatrix}. \tag{13.10}$$

Thus, to find $\vec{\mathbf{U}}^{n+1}$, we need to solve a tridiagonal linear system, which we can do by the Thomas algorithm of Lecture 8, using only $O(M)$ operations.

Above, we have derived the CN scheme using the analogy with the modified implicit Euler method for ODEs. This analogy allows us to expect that the two key features of the latter method: the unconditional stability and the second-order accuracy, are inherited by the CN method. Below we show that this is indeed the case.

13.2 Truncation error of the Crank–Nicolson method

The easiest (and most instructive — because we will also use it in Lecture 14 for multiple generalizations) way to derive the truncation error of the CN method is to use the following observation. Note that the stencil for this method is symmetric relative to the "virtual node" $(mh, (n+\frac{1}{2})\kappa)$, marked by a cross in the figure on the right. This motivates one to expand quantities U_m^n etc. about that virtual node. Let us denote the value of the solution at that point by $\bar{U}$:



$$\bar{U} \;\equiv\; u\left(mh, (n+\frac{1}{2})\kappa\right).$$

Then, using the Taylor expansion of a function of two variables (see Lecture 0), we obtain:

for $\varepsilon = -1$, 0, or 1:

$$\begin{aligned} U^{n+1}_{m+\varepsilon} \;=\; & \bar{U}_m + \left(\frac{\kappa}{2}\bar{U}_{m,t} + \varepsilon h \bar{U}_{m,x}\right) \\ & + \frac{1}{2!}\left(\left(\frac{\kappa}{2}\right)^2 \bar{U}_{m,tt} + 2\frac{\kappa}{2}\varepsilon h \bar{U}_{m,xt} + (\varepsilon h)^2 \bar{U}_{m,xx}\right) \\ & + \frac{1}{3!}\left(\left(\frac{\kappa}{2}\right)^3 \bar{U}_{m,ttt} + 3\left(\frac{\kappa}{2}\right)^2 \varepsilon h \bar{U}_{m,xtt} + 3\frac{\kappa}{2}(\varepsilon h)^2 \bar{U}_{m,xxt} + (\varepsilon h)^3 \bar{U}_{m,xxx}\right) \\ & + O(\kappa^4 + \kappa^3 h + \kappa^2 h^2 + \kappa h^3 + h^4)\,, \end{aligned} \tag{13.11}$$

where $\bar{U}_{m,t} = \frac{\partial}{\partial t} U_m|_{t=(n+\frac{1}{2})\kappa}$, etc. Similarly,

$$\begin{aligned} U^{n}_{m+\varepsilon} \;=\; & \bar{U}_m + \left(-\frac{\kappa}{2}\bar{U}_{m,t} + \varepsilon h \bar{U}_{m,x}\right) \\ & + \frac{1}{2!}\left(\left(-\frac{\kappa}{2}\right)^2 \bar{U}_{m,tt} + 2\left(-\frac{\kappa}{2}\right)\varepsilon h \bar{U}_{m,xt} + (\varepsilon h)^2 \bar{U}_{m,xx}\right) \\ & + \frac{1}{3!}\left(\left(-\frac{\kappa}{2}\right)^3 \bar{U}_{m,ttt} + 3\left(-\frac{\kappa}{2}\right)^2 \varepsilon h \bar{U}_{m,xtt} + 3\left(-\frac{\kappa}{2}\right)(\varepsilon h)^2 \bar{U}_{m,xxt} + (\varepsilon h)^3 \bar{U}_{m,xxx}\right) \\ & + O(\kappa^4 + \kappa^3 h + \kappa^2 h^2 + \kappa h^3 + h^4)\,. \end{aligned} \tag{13.12}$$

In a homework problem you will be asked to provide details of the following derivation. Namely, substituting expressions (13.11) and (13.12) with $\varepsilon = 0$ into the l.h.s. of (13.6), one obtains:

$$\frac{1}{\kappa}\delta_t U^n_m \;=\; \bar{U}_{m,t} + O(\kappa^2)\,. \tag{13.13}$$

Next, substituting expressions (13.11) and (13.12) into the r.h.s. of (13.5), one obtains:

$$\frac{1}{2h^2}\left(\delta_x^2 U^n_m + \delta_x^2 U^{n+1}_m\right) \;=\; \bar{U}_{m,xx} + O(h^2 + \kappa^2)\,. \tag{13.14}$$

Finally, combining the last two equations yields

$$\frac{1}{\kappa}\delta_t U^n_m - \frac{1}{2h^2}\left(\delta_x^2 U^n_m + \delta_x^2 U^{n+1}_m\right) \;=\; \bar{U}_{m,t} - \bar{U}_{m,xx} + O(\kappa^2 + h^2)\,, \tag{13.15}$$

which means that the CN scheme (13.6) is second-order accurate in time.

Remark Note that the notation $O(\kappa^2 + h^2)$ for the truncation error of the CN method does *not* necessarily imply that the sizes of κ and h may be taken to be about the same in practice. One of the homework problems explores this issue in detail.

Let us now consider the following obvious generalization of the modified implicit Euler scheme:

$$\vec{\mathbf{U}}^{n+1} = \vec{\mathbf{U}}^n + \kappa\left((1-\theta)\vec{\mathbf{f}}(t_n, \vec{\mathbf{U}}^n) + \theta\vec{\mathbf{f}}(t_{n+1}, \vec{\mathbf{U}}^{n+1})\right)\,, \tag{13.16}$$

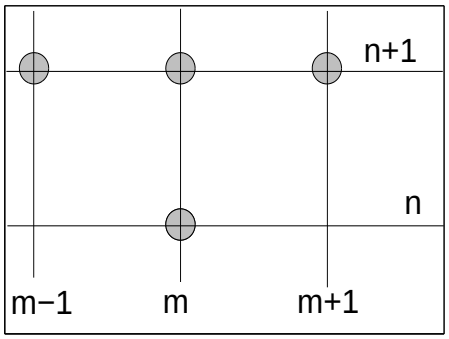
where the constant $\theta \in [0, 1]$. The corresponding method for the Heat equation is, instead of (13.9):

$$(I - r\theta A)\,\vec{\mathbf{U}}^{n+1} \;=\; (I + r(1-\theta)A)\,\vec{\mathbf{U}}^n + \vec{\mathbf{b}}_\theta\,. \tag{13.17}$$

In a QSA, you will be asked to write out the explicit form of $\vec{\mathbf{b}}_\theta$.

Obviously, when:

$\theta = \frac{1}{2}$, $\Rightarrow$ (13.17) is the CN method;
$\theta = 0$, $\Rightarrow$ (13.17) is the simple explicit method (12.12);
$\theta = 1$, $\Rightarrow$ (13.17) is an analogue of the simple implicit Euler method for the Heat equation; its stencil is shown on the right.



We will refer to methods (13.17) with all possible values of θ as the θ-family of methods.

Following the derivation of the Douglas method in Sec. 12.4 and of Eq. (13.15) above, one can show that the discretization error[40] of the θ-family of methods is

$$\text{discretization error of (13.17)} \;=\; \left(\left(\frac{1}{2}-\theta\right)\kappa - \frac{h^2}{12}\right) u_{xxxx} + O(\kappa^2 + h^4)\,. \tag{13.18}$$

Then it follows that in addition to the special value $\theta = \frac{1}{2}$, which gives the second-order accurate in time CN method, there is another special values of θ:

$$\theta = \frac{1}{2} - \frac{1}{12r}\,. \tag{13.19}$$

When θ is given by the above formula, the first term on the r.h.s. of (13.18) vanishes, and the discretization error becomes $O(\kappa^2 + h^4)$. The corresponding scheme is called *Crandall's method* or the *optimal method*. For values of θ other than (13.19) and $\theta = 1/2$ (when one has the CN scheme), the discretization error of (13.17) is only $O(\kappa + h^2)$.

13.3 Stability of the θ-family of methods

Here we use Method 1 of Lecture 12 to study stability of scheme (13.17). In a homework problem, you will be asked to obtain the same results using the von Neumann stability analysis (Method 2 of Lecture 12).

Following the lines of Method 1 (which is based on the material presented in Appendix of Lecture 12 and which you should review), we rewrite Eq. (13.17) with $\vec{\mathbf{b}}_\theta$ being set to zero as

$$\vec{\mathbf{U}}^{n+1} = (I - r\theta A)^{-1}\,(I + r(1-\theta)A)\,\vec{\mathbf{U}}^{n}. \tag{13.20}$$

Now recall from Linear Algebra that matrices A, $aA + bI$, and $(cA + dI)^{-1}$ have the same eigenvectors, with the corresponding eigenvalues being λ, $a\lambda + b$, and $(c\lambda + d)^{-1}$. (You will be asked to confirm these statements, along with Eq. (13.21) below, in a homework problem.)

Then, since the eigenvectors of $(I + r(1-\theta)A)$ and $(I - r\theta A)^{-1}$ are the same, the eigenvalues of the matrix appearing on the r.h.s. of (13.20) are easily found to be

$$\frac{1 + r(1-\theta)\lambda_j}{1 - r\theta\lambda_j}, \tag{13.21}$$

[40] Recall that this is one of the three types of error that was defined in Lecture 1 (please review that material if you do not remember the difference between the types). The discretization error is the one caused by the replacement of derivatives by finite differences in the equation. It has the same order (in κ and h) as the global error, but, unlike the latter, can be explicitly found.

where λ_j are the eigenvalues of A. According to the Lemma of Lecture 12,

$$\lambda_j \;=\; -2 + 2\cos\frac{\pi j}{M} \;=\; -4\sin^2\left(\frac{\pi j}{2M}\right), \qquad j = 1, \ldots, M-1. \tag{13.22}$$

As pointed out above, for stability of the scheme, it is necessary that

$$\left|\frac{1 + r(1-\theta)\lambda_j}{1 - r\theta\lambda_j}\right| \le 1, \qquad j = 1, \ldots, M-1. \tag{13.23}$$

Using (13.22) and denoting

$$\phi_j \equiv \frac{\pi j}{2M},$$

one rewrites (13.23) as

$$\left|1 - 4r(1-\theta)\sin^2\phi_j\right| \le \left|1 + 4r\theta\sin^2\phi_j\right|. \tag{13.24}$$

Since $\theta \ge 0$ by assumption and $r > 0$ by definition, then the expression under the absolute value sign on the r.h.s. of (13.24) is positive, and therefore the above inequality can be written as

$$-\left(1 + 4r\theta\sin^2\phi_j\right) \le 1 - 4r(1-\theta)\sin^2\phi_j \le 1 + 4r\theta\sin^2\phi_j. \tag{13.25}$$

The right part of this double inequality is automatically satisfied for all ϕ_j. The left part is satisfied when

$$4r(1-2\theta)\sin^2\phi_j \le 2\,. \tag{13.26}$$

The strongest restriction on r (and hence on the step size in time) occurs when $\sin^2\phi_j$ takes on its largest value, i.e. 1.[41]

In that case, (13.26) yields

$$(1-2\theta)r \le \frac{1}{2}\,. \tag{13.27}$$

This inequality should be considered separately in two cases:

$\underline{\frac{1}{2} \le \theta \le 1}$ $\quad\Rightarrow\quad$ r is arbitrary. (13.28)

That is, scheme (13.17) is unconditionally stable for any r.

$\underline{0 \le \theta < \frac{1}{2}}$ $\quad\Rightarrow\quad$ Scheme (13.17) is stable provided that

$$r \le \frac{1}{2(1-2\theta)}\,. \tag{13.29}$$

The above results show that the CN method, as well as the purely implicit method (13.17) with $\theta = 1$, are unconditionally stable. That is, **no** *relation between κ and h must hold in order for these schemes to converge* to the exact solution of the Heat equation. This is the **main advantage** of the CN method over the explicit methods of Lecture 12.

Let us also note that Crandall's method "(13.17) + (13.19)" belongs to the conditionally stable case, (13.29). However, since for Crandall's method,

$$r = \frac{1}{6(1-2\theta)} < \frac{1}{2(1-2\theta)}, \tag{13.30}$$

then, according to (13.29), this method is stable.

[41] Strictly speaking, $\sin^2\phi_j$ is always less than 1 since even the largest $\phi_j = \phi_{M-1}$ is slightly less than $\pi/2$; however, for large M, $\sin^2\phi_{M-1}$ is very close to 1.

13.4 Ways to improve on the accuracy of the Crank–Nicolson method

To improve on the accuracy of the CN method in time, one may use higher-order multi-step methods or implicit Runge–Kutta methods. Recall, however, that no method of order higher than 2 is absolutely stable, and therefore any scheme that one may expect to obtain along these lines will be (at best) only conditionally stable. Note also that the stencil for a multi-step generalization of the CN scheme will contain nodes on more than two time levels.

To improve the accuracy of the CN method in space, one can use the analogy with Numerov's method. Namely, we rewrite the Heat equation $u_{xx} = u_t$ as

$$\frac{\delta_x^2 U_m^n + \delta_x^2 U_m^{n+1}}{2h^2} = \frac{1}{12}\left(\frac{\delta_t U_{m+1}^n}{\kappa} + 10\frac{\delta_t U_m^n}{\kappa} + \frac{\delta_t U_{m-1}^n}{\kappa}\right). \tag{13.31}$$

However, one can verify that the resulting scheme is nothing but Crandall's method!

Finally, we also mention a method attributed to DuFort and Frankel:

$$\frac{U_m^{n+1} - U_m^{n-1}}{2\kappa} = \frac{1}{h^2}\left(U_{m+1}^n - \left[U_m^{n+1} + U_m^{n-1}\right] + U_{m-1}^n\right). \tag{13.32}$$

This method has the discretization error of

$$O\left(\kappa^2 + h^2 + \left(\frac{\kappa}{h}\right)^2\right),$$

which means that it is consistent with the Heat equation only when $(\kappa/h) \to 0$. When $(\kappa/h) \to$ const $\neq 0$, the DuFort–Frankel method approximates a *different*, hyperbolic, PDE. Although the DuFort–Frankel method can be shown to be unconditionally stable, it is not used for solution of parabolic PDEs because of the aforementioned need to have $\kappa \ll h$ in order to provide a scheme consistent with the PDE.

13.5 Questions for self-assessment

1. Why does one want to use an implicit method to solve the Heat equation?
2. Make sure you can obtain (13.7), and from it, (13.8) and (13.9).
3. How many (order of magnitude) operations is required to advance the solution of the Heat equation from one time level to the next using the CN method? Is the CN a time-efficient method?
4. Make sure you can obtain (13.11) and (13.12).
5. What is the order of discretization error of the CN method?
6. Make sure you can obtain (13.17). In particular, write out the explicit form of $\vec{\mathbf{b}}_\theta$.
7. Why do you think Crandall's method is called the "optimal" method?
8. Verify the first equality in (13.22).
9. What is the significance of the inequality (13.27)?

10. What is the main advantage of the CN method over the simple explicit method of Lecture 12?

11. Do you think Crandall's method has the same advantage?
    Also, explain the origin of each part of formula (13.30).

12. Is it possible to derive an unconditionally stable method with accuracy $O(\kappa^3)$ (or better) for the Heat equation? If 'yes', then how? If 'no', why?

13. Draw the stencil for the DuFort–Frankel method.

14. Do you think the DuFort–Frankel method is time-efficient?