Parabolic PDEs

32.4

Introduction

Second-order partial differential equations (PDEs) may be classified as parabolic, hyperbolic or elliptic. Parabolic and hyperbolic PDEs often model time dependent processes involving initial data.

In this Section we consider numerical solutions of parabolic problems.

Prerequisites

Before starting this Section you should ...

- review difference methods for first and second derivatives (HELM 31.3)

Learning Outcomes

On completion you should be able to ...

- implement simple methods to obtain approximate solutions of the heat diffusion equation

1. Definitions

We begin by giving some definitions.
Suppose that $u = u(x,t)$ satisfies the second order partial differential equation

$$Au_{xx} + Bu_{xt} + Cu_{tt} + Du_x + Eu_t + Fu = G$$

in which $A, \ldots, G$ are given functions. This equation is said to be

parabolic if $B^2 - 4AC = 0$

hyperbolic if $B^2 - 4AC > 0$

elliptic if $B^2 - 4AC < 0$

These may look like rather abstract definitions at this stage, but we will see that equations of different types give rise to mathematical models of different physical situations. In this Section we will consider equations only of the parabolic type. The hyperbolic type is dealt with later in this Workbook and the elliptic type is discussed in HELM 33.

2. Motivation

Consider an example of the type seen in the earlier material concerning separable solutions of the heat conduction equation. Suppose that $u = u(x,t)$ is the temperature of a metal bar a distance x from one end and at time t. For the sake of argument let us suppose that the metal bar has length equal to ℓ and that the ends are held at constant temperatures u_L at the left and u_R at the right.

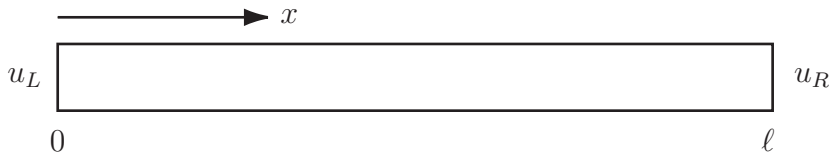

Figure 2

We also suppose that the temperature distribution at the initial time is known to be $f(x)$, with $f(0) = u_L$ and $f(\ell) = u_R$ so that the initial and boundary conditions do not give rise to a conflict at the ends of the bar at the initial time.

This physical situation may be modelled by

$$\left.\begin{array}{rcll} u_t & = & \alpha u_{xx} & (0 < x < \ell, \quad t > 0) \\ u(0,t) & = & u_L & (t > 0) \\ u(\ell,t) & = & u_R & (t > 0) \\ u(x,0) & = & f(x) & (0 < x < \ell) \end{array}\right\}$$

in which $\alpha > 0$ is a constant called the **thermal diffusivity** or simply the **diffusivity** of the metal. If the bar is made of aluminium then $\alpha = 0.86$ cm^2 s^{-1}, and if made of copper then $\alpha = 1.14$ cm^2 s^{-1}.

Using separation of variables and Fourier series (neither of which are required for the remainder of this Section) it can be shown that the solution to the above problem (in the case where $u_L = u_R = 0$) is

$$u(x,t) = \sum_{m=1}^{\infty} B_m e^{-m^2\alpha\pi^2 t/\ell^2} \sin(m\pi x/\ell), \qquad \text{where} \qquad B_m = \frac{2}{\ell}\int_0^{\ell} f(s)\sin(m\pi s/\ell)\, ds.$$

Now, let us be realistic. Any evaluation of u for particular choices of x and t must involve approximating the infinite series that defines u (that is, just taking the first few terms - and care is required if we are to be sure that we have taken enough). Also, in each of the terms we retain in the sum, we need to find B_m by integration. It is not surprising that computation of this procedure is a common approach. So if we (eventually) resort to computation in order to find u, why not *start* with a computational approach?

(This is not to say that there is no value in the analytic solution involving the B_m. The solution above is of great value, but we simply observe here that there are times when a computational approach is all we may end up needing.)

So, the aim of this Section is to derive methods for obtaining numerical solutions to parabolic problems of the type above. In fact, it is sufficient for our present purposes to restrict attention to that particular problem.

3. Approximating partial derivatives

Earlier, in HELM 31.3, we saw methods for approximating first and second derivatives of a function of one variable. We review some of that material here. If $y = y(x)$ then the forward and central difference approximations to the first derivative are:

$$\frac{dy}{dx} \approx \frac{y(x+\delta x) - y(x)}{\delta x}, \qquad \frac{dy}{dx} \approx \frac{y(x+\delta x) - y(x-\delta x)}{2\delta x}$$

and the central difference approximation to the second derivative is:

$$\frac{\mathrm{d}^2 y}{dx^2} \approx \frac{y(x+\delta x) - 2y(x) + y(x-\delta x)}{(\delta x)^2}$$

in which δx is a small x-increment. The quantity δx is what we previously referred to as h, but it is now convenient to use a notation which is more closely related to the independent variable (in this case x). (Examples implementing the difference approximations for derivatives can be found in HELM 31.)

We now return to the subject of this Section, that of *partial* derivatives. The PDE $u_t = \alpha u_{xx}$ involves the first derivative $\dfrac{\partial u}{\partial t}$ and the second derivative $\dfrac{\partial^2 u}{\partial x^2}$. We now adapt the ideas used for functions of one variable to the present case involving $u = u(x,t)$.

Let δt be a small increment of t, then the partial derivative $\dfrac{\partial u}{\partial t}$ may be approximated by:

$$\frac{\partial u}{\partial t} \approx \frac{u(x,t+\delta t) - u(x,t)}{\delta t}$$

Let δx be a small increment of x, then the partial derivative $\dfrac{\partial^2 u}{\partial x^2}$ may be approximated by:

$$\frac{\partial^2 u}{\partial x} \approx \frac{u(x+\delta x,t) - 2u(x,t) + u(x-\delta x,t)}{(\delta x)^2}$$

The two difference approximations above are the ones we will use later in this Section. Example 14 below refers to these and others.

Example 14

Consider the function u defined by

$$u(x,t) = \sin(x^2 + 2t)$$

Using increments of $\delta x = 0.004$ and $\delta t = 0.04$, and working to 8 decimal places, approximate

(a) $u_x(2,3)$ with a one-sided forward difference

(b) $u_{xx}(2,3)$ with a central difference

(c) $u_t(2,3)$ with a one-sided forward difference

(d) $u_t(2,3)$ with a central difference.

Enter your approximate derivatives to 3 decimal places.

Solution

The evaluations of u we will need are $u(x,t) = -0.54402111$, $u(x+\delta x,t) = -0.55738933$, $u(x-\delta x,t) = -0.53054047$, $u(x,t+\delta t) = -0.60933532$, $u(x,t-\delta t) = -0.47522703$. It follows that

(a) $u_x(2,3) \approx \dfrac{-0.55738933 + 0.54402111}{0.004} = -3.342$

(b) $u_{xx}(2,3) \approx \dfrac{-0.55738933 + 2 \times 0.54402111 - 0.53054047}{0.004^2} = 7.026$

(c) $u_t(2,3) \approx \dfrac{-0.60933532 + 0.54402111}{0.04} = -1.633$

(d) $u_t(2,3) \approx \dfrac{-0.60933532 + 0.47522703}{2 \times 0.04} = -1.676$

to 3 decimal places. (Workings shown to 8 decimal places.)

4. An explicit numerical method for the heat equation

The approximations used above for approximating partial derivatives can now be applied in order to derive a numerical method for solving the heat conduction problem

$$\begin{aligned} u_t &= \alpha u_{xx} && (0 < x < \ell, \quad t > 0) \\ u(0,t) &= 0 && (t > 0) \\ u(\ell,t) &= 0 && (t > 0) \\ u(x,0) &= f(x) && (0 < x < \ell). \end{aligned}$$

In order to specify the numerical method we choose values for δt and δx and use these in approximations of the two derivatives in the partial differential equation. It is convenient to divide the

interval $0 < x < \ell$ into equally spaced subintervals so, in effect, we choose a whole number J so that $\delta x = \dfrac{\ell}{J}$.

Key Point 15

In order to specify the numerical procedure for solving the heat conduction equation

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

we need to choose

δt — the time step

δx — the space step

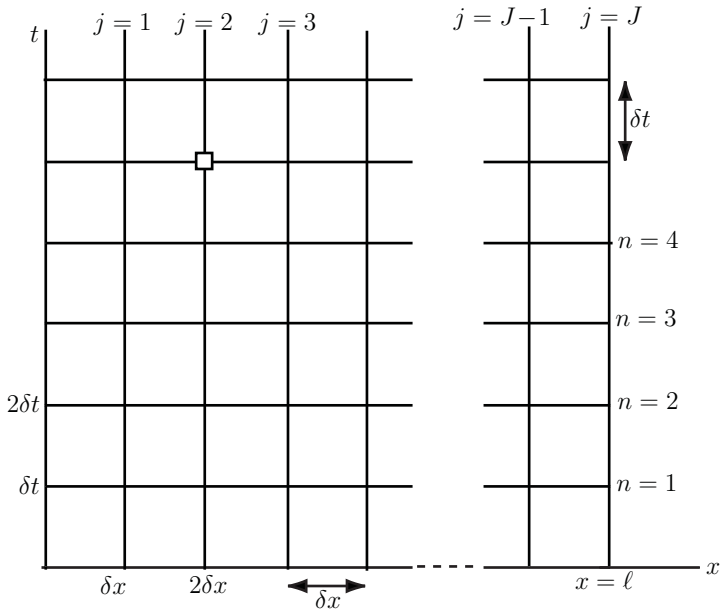

Figure 3

The diagram above shows the independent variables x and t at which we seek the function u. The numerical solution we shall find is a sequence of numbers which approximate u at a sequence of (x, t) points.

Key Point 16

The numerical approximations to $u(x,t)$ that we will find will be approximations to u at (x,t) values where the horizontal and vertical lines cross in the above diagram (Figure 3).

The notation we use is that

$$\underset{\substack{\uparrow \\ \text{numerical} \\ \text{approximation}}}{u_j^n} \quad \approx \quad \underbrace{u(j\ \delta x\ ,\ n\ \delta t)}_{\substack{\uparrow \\ \text{exact (i.e., unknown) solution} \\ \text{evaluated at } x = j \times \delta x,\ t = n \times \delta t}}$$

The idea is that the subscript j counts how many "steps" to the right we have taken from the origin and the superscript n counts how many time-steps (up, on the diagram) we have taken. To say this another way

u_j^n — the superscript counts up the t values; the subscript counts across the x values

For example, consider the point on Figure 3 which is highlighted with a small square. This point is two steps to the right of the origin (so that $j = 2$) and five steps up (so that $n = 5$). The exact solution evaluated at this point is $u(2\delta x, 5\delta t)$ and our numerical approximation to that value is u_2^5. Combining this new notation with the familiar idea for approximating derivatives we obtain the following approximation to the PDE

$$\frac{u_j^{n+1} - u_j^n}{\delta t} = \alpha\ \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{(\delta x)^2}$$

Key Point 17

The exact solution $u = u(x,t)$ satisfies the partial differential equation

$$u_t = \alpha u_{xx}$$

The approximate (numerical) solution satisfies the difference equation

$$\frac{u_j^{n+1} - u_j^n}{\delta t} = \alpha\ \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{(\delta x)^2}$$

The difference between the unknown exact solution and the numerical solution will be governed by how well the one-sided and central differences approximate the partial derivatives in the PDE.

To simplify (the appearance of) the numerical method we define a new quantity $r = \dfrac{\alpha\delta t}{(\delta x)^2}$ so that our numerical procedure can be written

$$u_j^{n+1} = u_j^n + r(u_{j-1}^n - 2u_j^n + u_{j+1}^n) = ru_{j-1}^n + (1-2r)u_j^n + ru_{j+1}^n$$

This equation defines a numerical "stencil" which allows us to find one of the values at the $n+1$ time level in terms of values at the previous level, n. In Figure 4 we envisage terms on the right-hand side of the above equation leading towards a result equal to the left-hand side, and the arrows therefore point towards the point at which u_j^{n+1} approximates u.

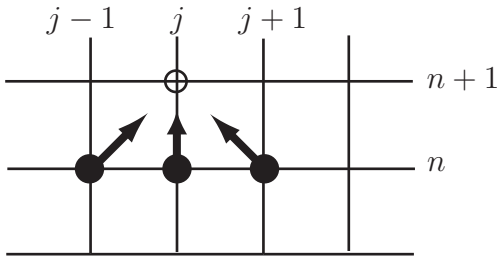

Figure 4

At the stage of the process depicted above, the solid circles represent points in the (x,t) plane where we have already found our numerical approximation. The unfilled circle is the point for which the new approximation u_j^{n+1} is being found.

Implementation

The initial condition gives u at $t=0$, and this information can be used to find

$$u_0^0, \quad u_1^0, \quad u_2^0, \quad \ldots, \quad u_J^0$$

that is, the numerical solution at all the selected x values and at $t=0$. In general

$$u_j^0 = f(j \times \delta x) = f_j$$

where f_j is a shorthand notation for $f(j \times \delta x)$.
Then we use the boundary conditions and numerical method

$$u_j^{n+1} = u_j^n + r(u_{j-1}^n - 2u_j^n + u_{j+1}^n)$$

(with $n=0$) to work out u_j^1 for $j = 0, 1, 2, \ldots, J$. (This completes the first time-step.)

The time-stepping procedure is then used repeatedly to find u_j^{n+1} in terms of the u_j^n, which are known either from the last time-step or (at the beginning) from the initial condition.

The time-stepping procedure is summarised in the following Key Point.

Key Point 18

Here the step-by-step process used to implement the numerical procedure is presented.

1. The initial condition implies that

$$u_j^0 = f_j \qquad (j = 0, 1, 2, \ldots, J)$$

(the boundary conditions could be used to find u_0^0 and u_J^0, but our supposition is that this is consistent with taking f_0 and f_J).

2. The first time-step

Here we find u_j^1 for $j = 0, 1, \ldots, J$.

(a) The boundary condition at $x = 0$ is $u(0, t) = u_L$. It follows that $u_0^1 = u_L$.

(b) The boundary condition at $x = \ell$ is $u(\ell, t) = u_R$. It follows that $u_J^1 = u_R$.

(c) Now we work from left to right finding u_j^1 at the interior points. This is achieved by repeatedly applying the general numerical scheme:

$$\begin{aligned} u_1^1 &= u_1^0 + r(u_0^0 - 2u_1^0 + u_2^0) \\ u_2^1 &= u_2^0 + r(u_1^0 - 2u_2^0 + u_3^0) \\ &\vdots \\ u_{J-1}^1 &= u_{J-1}^0 + r(u_{J-2}^0 - 2u_{J-1}^0 + u_J^0) \end{aligned}$$

This completes the first time-step. We have taken the initial data and used our approximation to the PDE to obtain an approximate solution at time $t = \delta t$.

3. The second time-step

Here we find u_j^2 for $j = 0, 1, \ldots, J$.

(a) The boundary condition at $x = 0$ is $u(0, t) = u_L$. It follows that $u_0^2 = u_L$.

(b) The boundary condition at $x = \ell$ is $u(\ell, t) = u_R$. It follows that $u_J^2 = u_R$.

(c) Now we work from left to right finding u_j^2 at the interior points. This is achieved by repeatedly applying the general numerical scheme:

$$\begin{aligned} u_1^2 &= u_1^1 + r(u_0^1 - 2u_1^1 + u_2^1) \\ u_2^2 &= u_2^1 + r(u_1^1 - 2u_2^1 + u_3^1) \\ &\vdots \\ u_{J-1}^2 &= u_{J-1}^1 + r(u_{J-2}^1 - 2u_{J-1}^1 + u_J^1) \end{aligned}$$

This completes the second time-step. We now have an approximation to u at time $t = 2\delta t$.

4. And so on ....

The following is a concrete example of the time-stepping procedure.

Example 15

The temperature $u(x,t)$ of a metal bar of length $\ell = 2$ at a distance x from one end and at time t is modelled by the partial differential equation

$$u_t = \alpha u_{xx} \qquad (0 < x < \ell, \quad t > 0)$$

It is given that the metal has diffusivity $\alpha = 4$, that the two ends of the bar are kept at temperature $u = 0$ and that the initial temperature distribution is

$$u(x,0) = f(x) = x(\ell - x)$$

Use the explicit difference scheme with $\delta x = 0.5$ and $\delta t = 0.01$ to approximate $u(x,t)$ at $t = \delta t$ and $t = 2\delta t$.

Solution

In this case $r = \alpha\delta t/(\delta x)^2 = 0.16$ so that the numerical method can be written

$$u_j^{n+1} = u_j^n + 0.16(u_{j-1}^n - 2u_j^n + u_{j+1}^n) = 0.68u_j^n + 0.16(u_{j-1}^n + u_{j+1}^n)$$

We now find u_j^0

$$\begin{array}{lcll} u_0^0 & = & 0 & \text{from the left-hand boundary condition} \\ u_1^0 & = & f(\delta x) = 0.75 & \text{from the initial condition} \\ u_2^0 & = & f(2\delta x) = 1 & \text{from the initial condition} \\ u_3^0 & = & f(3\delta x) = 0.75 & \text{from the initial condition} \\ u_4^0 & = & 0 & \text{from the boundary condition at the right hand end} \end{array}$$

The first time-step will find u_j^1, but first we note that $u_0^1 = u_4^1 = 0$ from the two boundary conditions. Now

$$\begin{array}{lcl} u_1^1 & = & 0.68u_1^0 + 0.16(u_0^0 + u_2^0) = 0.68 \times 0.75 + 0.16(0 + 1) = 0.670 \\ u_2^1 & = & 0.68u_2^0 + 0.16(u_1^0 + u_3^0) = 0.68 \times 1 + 0.16(0.75 + 0.75) = 0.920 \\ u_3^1 & = & 0.68u_3^0 + 0.16(u_2^0 + u_4^0) = 0.68 \times 0.75 + 0.16(1 + 0) = 0.670 \end{array}$$

The second time-step will find u_j^2, but first we note that $u_0^2 = u_4^2 = 0$ from the two boundary conditions. Now

$$\begin{array}{lcl} u_1^2 & = & 0.68u_1^1 + 0.16(u_0^1 + u_2^1) = 0.68 \times 0.67 + 0.16(0 + 0.92) = 0.603 \\ u_2^2 & = & 0.68u_2^1 + 0.16(u_1^1 + u_3^1) = 0.68 \times 0.92 + 0.16(0.67 + 0.67) = 0.84 \\ u_3^2 & = & 0.68u_3^1 + 0.16(u_2^1 + u_4^1) = 0.68 \times 0.67 + 0.16(0.92 + 0) = 0.603 \end{array}$$

(Quantities have been rounded to three decimal places here.)

Figure 5 plots the numerical solutions found in the example above. The initial condition is shown as circles. Results of the first time-step appear as squares and the second time-step is shown as stars. The line joining the values we found are not part of the numerical solution and are included only as

an aid to clarity.

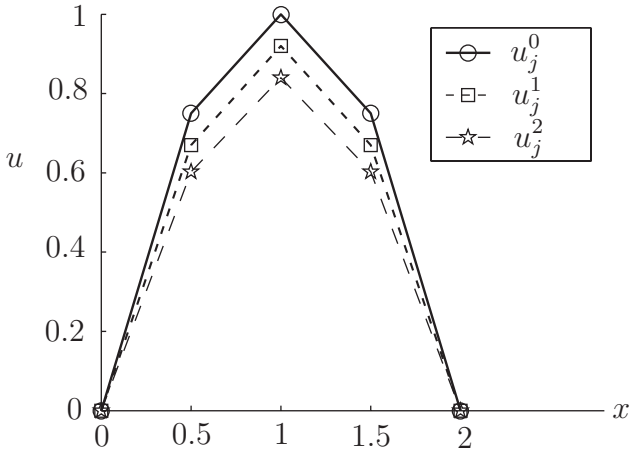

Figure 5

Notice how the numerical results are behaving as they should. The temperature decreases slightly at each time-step.

The temperature $u(x,t)$ of a metal bar of length $\ell = 2$ at a distance x from one end and at time t is modelled by the partial differential equation

$$u_t = \alpha u_{xx} \qquad (0 < x < \ell, \quad t > 0)$$

It is given that the metal has diffusivity $\alpha = 2.25$, that the two ends of the bar are kept at temperature $u = 0$ and that the initial temperature distribution is

$$u(x,0) = f(x) = \sin(\pi x/\ell)$$

Use the explicit difference scheme with $\delta x = 0.5$ and $\delta t = 0.05$ to approximate $u(x,t)$ at $t = \delta t$ and $t = 2\delta t$.

Your solution

Initial condition and first time-step:

Answer

In this case $r = \alpha\delta t/(\delta x)^2 = 0.45$ so that the numerical scheme can be written

$$u_j^{n+1} = u_j^n + 0.45(u_{j-1}^n - 2u_j^n + u_{j+1}^n) = 0.1u_j^n + 0.45(u_{j-1}^n + u_{j+1}^n)$$

The first stage is to use the given data to find u_j^0

$$\begin{array}{rcll} u_0^0 & = & 0 & \text{from the boundary condition} \\ u_1^0 & = & f(\delta x) = f(0.5) = 0.707 & \text{from the initial condition} \\ u_2^0 & = & f(2\delta x) = f(1) = 1 & \text{from the initial condition} \\ u_3^0 & = & f(3\delta x) = f(1.5) = 0.707 & \text{from the initial condition} \\ u_4^0 & = & 0 & \text{from the boundary condition} \end{array}$$

The first time-step will find u_j^1. First we note that the boundary condition implies that $u_0^1 = u_4^1 = 0$.

$$\begin{array}{rcl} u_1^1 & = & 0.1u_1^0 + 0.45(u_0^0 + u_2^0) = 0.1 \times 0.71 + 0.45(0 + 1) = 0.521 \\ u_2^1 & = & 0.1u_2^0 + 0.45(u_1^0 + u_3^0) = 0.1 \times 1 + 0.45(0.71 + 0.71) = 0.736 \\ u_3^1 & = & 0.1u_3^0 + 0.45(u_2^0 + u_4^0) = 0.1 \times 0.71 + 0.45(1 + 0) = 0.521 \end{array}$$

Your solution

Second time-step:

Answer

The second time-step will find u_j^2. First we note that the boundary condition implies that $u_0^2 = u_4^2 = 0$. Now

$$\begin{array}{rcl} u_1^2 & = & 0.1u_1^1 + 0.45(u_0^1 + u_2^1) = 0.1 \times 0.52 + 0.45(0 + 0.74) = 0.383 \\ u_2^2 & = & 0.1u_2^1 + 0.45(u_1^1 + u_3^1) = 0.1 \times 0.74 + 0.45(0.52 + 0.52) = 0.542 \\ u_3^2 & = & 0.1u_3^1 + 0.45(u_2^1 + u_4^1) = 0.1 \times 0.52 + 0.45(0.74 + 0) = 0.383 \end{array}$$

5. Stability of the simple explicit scheme

The purpose of the time-stepping scheme is to approximate $u(x,t)$ at later and later times t. It is clear that the larger we take the time step δt, the fewer steps will be necessary to reach a particular time t. One constraint on the size of δt is that we know from our earlier look at difference methods that derivative approximations are most accurate when *small* increments are used. However, as we will see in the next couple of pages, a far more telling constraint on the size of δt arises on consideration of **stability**. We begin with an Example.

Example 16

The temperature $u(x,t)$ of a metal bar of length $\ell = 1$ at a distance x from one end and at time t is modelled by the partial differential equation

$$u_t = \alpha u_{xx} \qquad (0 < x < \ell, \quad t > 0)$$

It is given that the metal has diffusivity $\alpha = 1$, that the two ends of the bar are kept at temperature $u = 0$ and that the initial temperature distribution is

$$u(x,0) = f(x) = x(\ell - x)$$

Use the explicit difference scheme with $\delta x = 0.25$ and $\delta t = 0.075$ to approximate $u(x,t)$ at $t = \delta t$ and $t = 2\delta t$.

Solution

In this case $r = \alpha\delta t/(\delta x)^2 = 1.2$ so that the numerical scheme can be written

$$u_j^{n+1} = u_j^n + 1.2(u_{j-1}^n - 2u_j^n + u_{j+1}^n) = -1.4u_j^n + 1.2(u_{j-1}^n + u_{j+1}^n)$$

The first stage is to use the given data to find u_j^0

u_0^0	$=$	0	from the boundary condition
u_1^0	$=$	$f(\delta x) = f(0.25) = 0.188$	from the initial condition
u_2^0	$=$	$f(2\delta x) = f(0.5) = 0.25$	from the initial condition
u_3^0	$=$	$f(3\delta x) = f(0.75) = 0.188$	from the initial condition
u_4^0	$=$	0	from the boundary condition

The first time-step will find u_j^1. First we note that the boundary condition implies that $u_0^1 = u_4^1 = 0$.

$$\begin{aligned} u_1^1 &= -1.4u_1^0 + 1.2(u_0^0 + u_2^0) = -1.4 \times 0.19 + 1.2(0 + 0.25) = 0.038 \\ u_2^1 &= -1.4u_2^0 + 1.2(u_1^0 + u_3^0) = -1.4 \times 0.25 + 1.2(0.188 + 0.188) = 0.1 \\ u_3^1 &= -1.4u_3^0 + 1.2(u_2^0 + u_4^0) = -1.4 \times 0.19 + 1.2(0.25 + 0) = 0.038 \end{aligned}$$

The second time-step will find u_j^2. First we note that the boundary condition implies that $u_0^2 = u_4^2 = 0$. Now

$$\begin{aligned} u_1^2 &= -1.4u_1^1 + 1.2(u_0^1 + u_2^1) = -1.4 \times 0.04 + 1.2(0 + 0.1) = 0.067 \\ u_2^2 &= -1.4u_2^1 + 1.2(u_1^1 + u_3^1) = -1.4 \times 0.1 + 1.2(0.038 + 0.038) = -0.05 \\ u_3^2 &= -1.4u_3^1 + 1.2(u_2^1 + u_4^1) = -1.4 \times 0.04 + 1.2(0.1 + 0) = 0.067 \end{aligned}$$

Figure 6 shows the results found in Example 16.

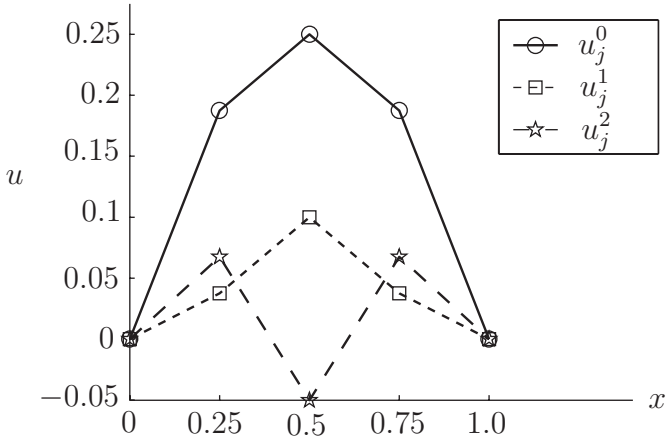

Figure 6

Something has gone wrong here. And it only gets worse in subsequent time-steps. After 9 time-steps the numerical solution approximating $u(x,t)$ at $t = 9\delta t$ is

$$\begin{aligned} u(0.25, 9\delta t) &\approx u_1^9 = -140.5531 \\ u(0.50, 9\delta t) &\approx u_2^9 = 198.7722 \\ u(0.75, 9\delta t) &\approx u_3^9 = -140.5531 \end{aligned}$$

(to 4 decimal places). This is an example of **instability**. A part of the numerical solution wants to keep growing and growing in a way that is not a part of the engineering application being modelled. There are many different definitions of (in)stability, and they often depend on the specific application in mind. For the heat conduction problem under discussion here, the following definition is sufficient.

Key Point 19

The explicit difference scheme $u_j^{n+1} = ru_{j-1}^n + (1-2r)u_j^n + ru_{j+1}^n \qquad \left(r = \dfrac{\alpha\delta t}{(\delta x)^2}\right)$

$$\begin{aligned} u_0^n &= 0 && (n > 0) \\ u_J^n &= 0 && (n > 0) \\ u_j^0 &= f(j\ \delta x) && (j = 1, 2, \ldots, J-1) \end{aligned}$$

where $J\delta x = \ell$, approximating the heat conduction problem

$$\left.\begin{aligned} u_t &= \alpha u_{xx} && (0 < x < \ell, \quad t > 0) \\ u(0,t) &= 0 && (t > 0) \\ u(\ell,t) &= 0 && (t > 0) \\ u(x,0) &= f(x) && (0 < x < \ell). \end{aligned}\right\}$$

is said to be **stable** if the approximations u_j^n do not grow in magnitude with n.

(Of course, there are applications where the principal quantity of interest *does* grow with time, and in these cases other definitions of stability are appropriate.)

The main stability result for the explicit scheme is proved in many textbooks on the subject, but for this Workbook it is sufficient to simply state it.

Key Point 20

The explicit scheme is stable if and only if

$$r \leq \tfrac{1}{2}$$

Writing this another way we see that the restriction on the time-step is that

$$\delta t \leq \frac{\delta x^2}{2\alpha}$$

Why is the stability constraint a problem?

In the above account it has been stated that the stability constraint is a severe restriction on the time-step δt. Here we discuss why this is the case.

For sake of argument let us take an example where $\alpha = 1$ and choose $\delta x = \dfrac{1}{10}$. The stability requirement insists that we must choose

$$\delta t \leq \frac{1}{2}\delta x^2 = \frac{1}{200},$$

which is much smaller than δx. If we require an even smoother approximation in the x direction we could halve δx taking it to be equal to $\dfrac{1}{20}$. It is now necessary that

$$\delta t \leq \frac{1}{2}\delta x^2 = \frac{1}{800}.$$

Decreasing δx by a factor of 2 causes δt to decrease by a factor of 4. The problem is that the upper bound on δt involves the *square* of δx, which is likely to be very small.

The following method overcomes the requirement of tiny time-steps.

6. The Crank-Nicolson method

In the notation established for the explicit method, the so-called Crank-Nicolson scheme can be written

$$u_j^{n+1} = u_j^n + \tfrac{1}{2}r\left(\underbrace{u_{j-1}^n - 2u_j^n + u_{j+1}^n}_{\dagger} \quad + \quad \underbrace{u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1}}_{\ddagger}\right)$$

which might, at first glance, look off-puttingly complicated. To aid clarity, certain groups of terms have been gathered together in the above:

† these are the terms that appeared on the right hand side of the explicit method and are involved with approximating u_{xx} at time $t = n\ \delta t$

‡ these are very similar to the † terms, but all the superscripts are $n+1$ instead of n, that is the terms ‡ approximate u_{xx} at time $t = (n+1)\ \delta t$

(the factor of $\frac{1}{2}$ outside the large bracket shows that we take the *average* of † and ‡)

Figure 7 shows another way of thinking of this numerical method. As in the earlier diagram of this type, arrows point away from positions relating to terms on the right-hand side of the numerical scheme.

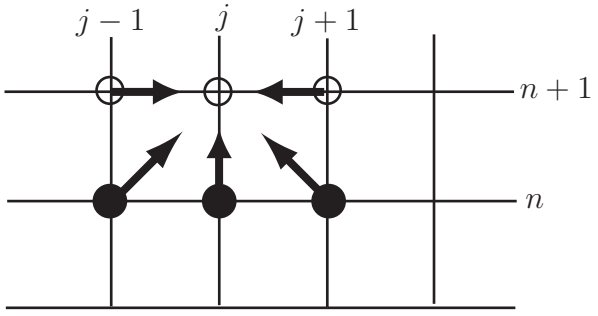

Figure 7

The new terms in the Crank-Nicolson method, as compared with the explicit method, give rise to two new unfilled circles on the diagram and the horizontal arrows.

The implementation of this method is similar to that used for the explicit method, but there is a key difference. The Crank-Nicolson scheme is **implicit**, for consider its use in the first time-step when finding u_j^1,

$$u_j^1 = \underbrace{u_j^0}_{\checkmark} \ + \ \tfrac{1}{2}r\left(\underbrace{u_{j-1}^0}_{\checkmark} - 2\underbrace{u_j^0}_{\checkmark} + \underbrace{u_{j+1}^0}_{\checkmark} \quad + \quad \underbrace{u_{j-1}^1 - 2u_j^1 + u_{j+1}^1}_{?}\right)$$

The terms labelled ✓ are known from the initial condition. But there are other unknown terms on the right-hand side. We cannot simply "read off" the values at the new time-step as we did using the explicit scheme. Instead we have to store all of the equations given by the stencil at a particular time-step and then solve them as a system of simultaneous equations. The following Example illustrates this point.

Example 17

The temperature $u(x,t)$ of a metal bar of length $\ell = 1.2$ at a distance x from one end and at time t is modelled by the partial differential equation

$$u_t = \alpha u_{xx} \qquad (0 < x < \ell, \quad t > 0).$$

It is given that the metal has diffusivity $\alpha = 1$, that the two ends of the bar are kept at temperature $u = 0$ and that the initial temperature distribution is

$$u(x,0) = f(x) = x\sqrt{(\ell - x)^3}$$

Use the Crank-Nicolson difference scheme with $\delta x = 0.4$ and $\delta t = 0.1$ to approximate $u(x,t)$ at $t = \delta t$ and $t = 2\delta t$.

Solution

In this case $r = \alpha\delta t/(\delta x)^2 = 0.62500$ so that the numerical scheme can be written

$$u_j^{n+1} = u_j^n + \frac{0.62500}{2}(u_{j-1}^n - 2u_j^n + u_{j+1}^n + u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1})$$

Moving the unknowns to the left of the equation we obtain

$$-0.31250u_{j-1}^{n+1} + 1.62500u_j^{n+1} - 0.31250u_{j+1}^{n+1} = 0.37500u_j^n + 0.31250(u_{j-1}^n + u_{j+1}^n)$$

The first stage is to use the given data to find u_j^0

u_0^0	$=$	0	from the boundary condition
u_1^0	$=$	$f(\delta x) = f(0.4) = 0.28622$	from the initial condition
u_2^0	$=$	$f(2\delta x) = f(0.8) = 0.20239$	from the initial condition
u_3^0	$=$	0	from the boundary condition

The first time-step will find u_j^1. First we note that the boundary condition implies that $u_0^1 = u_3^1 = 0$. Two uses of the stencil give

$$\begin{aligned} -0.31250u_0^1 + 1.62500u_1^1 - 0.31250u_2^1 = 0.37500u_1^0 + 0.31250(u_0^0 + u_2^0) &= 0.17058 \\ -0.31250u_1^1 + 1.62500u_2^1 - 0.31250u_3^1 = 0.37500u_2^0 + 0.31250(u_1^0 + u_3^0) &= 0.16534 \end{aligned}$$

The implicit nature of this method means that we have to do some extra work to complete the time-step. We must now solve the simultaneous equations

$$\begin{pmatrix} 1.62500 & -0.31250 \\ -0.31250 & 1.62500 \end{pmatrix} \begin{pmatrix} u_1^1 \\ u_2^1 \end{pmatrix} = \begin{pmatrix} 0.17058 \\ 0.16534 \end{pmatrix}$$

In this case there are only two unknowns and it is a simple matter to solve the pair of equations to give $u_1^1 = 0.12932$ and $u_2^1 = 0.12662$.

Solution (contd.)

The second time-step will find u_j^2. First we note that the boundary condition implies that $u_0^2 = u_3^2 = 0$. Two uses of the stencil give

$$\begin{aligned} -0.31250u_0^2 + 1.62500u_1^2 - 0.31250u_2^2 = 0.37500u_1^1 + 0.31250(u_0^1 + u_2^1) &= 0.08806 \\ -0.31250u_1^2 + 1.62500u_2^2 - 0.31250u_3^2 = 0.37500u_2^1 + 0.31250(u_1^1 + u_3^1) &= 0.08789 \end{aligned}$$

The implicit nature of this method means that we have to do some extra work to complete the time-step. We must now solve the simultaneous equations

$$\begin{pmatrix} 1.62500 & -0.31250 \\ -0.31250 & 1.62500 \end{pmatrix} \begin{pmatrix} u_1^2 \\ u_2^2 \end{pmatrix} = \begin{pmatrix} 0.08806 \\ 0.08789 \end{pmatrix}$$

In this case there are only two unknowns and it is a simple matter to solve the pair of equations to give $u_1^2 = 0.06707$ and $u_2^2 = 0.06699$.

Figure 8 depicts the numerical solutions found in Example 17 above. (Again, the dotted lines are intended to aid clarity, they are not part of the numerical solution.)

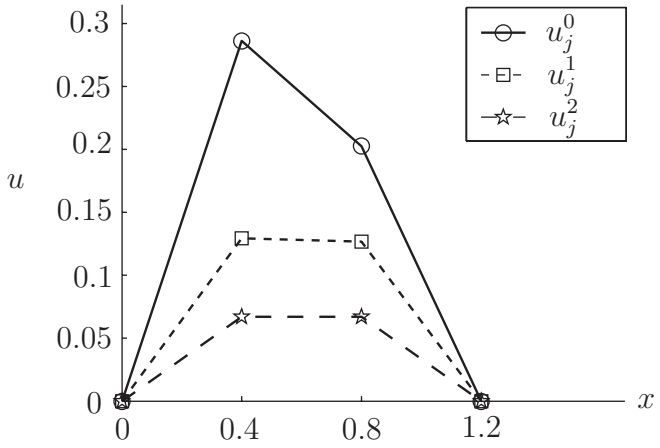

Figure 8

The temperature $u(x,t)$ of a metal bar of length $\ell = 0.9$ at a distance x from one end and at time t is modelled by the partial differential equation

$$u_t = \alpha u_{xx} \qquad (0 < x < \ell, \quad t > 0).$$

It is given that the metal has diffusivity $\alpha = 0.25$, that the two ends of the bar are kept at temperature $u = 0$ and that the initial temperature distribution is

$$u(x,0) = f(x) = \sin(\pi x/\ell)$$

Use the Crank-Nicolson difference scheme with $\delta x = 0.3$ and $\delta t = 0.2$ to approximate $u(x,t)$ at $t = \delta t$ and $t = 2\delta t$.

Your solution

Initial condition and first time-step:

Answer

In this case $r = \alpha\delta t/(\delta x)^2 = 0.55556$ so that the numerical scheme can be written

$$u_j^{n+1} = u_j^n + \frac{0.55556}{2}(u_{j-1}^n - 2u_j^n + u_{j+1}^n + u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1})$$

Moving the unknowns to the left of the equation we obtain

$$-0.27778u_{j-1}^{n+1} + 1.55556u_j^{n+1} - 0.27778u_{j+1}^{n+1} = 0.44444u_j^n + 0.27778(u_{j-1}^n + u_{j+1}^n)$$

The first stage is to use the given data to find u_j^0

u_0^0	$=$	0	from the boundary condition
u_1^0	$=$	$f(\delta x) = f(0.3) = 0.86603$	from the initial condition
u_2^0	$=$	$f(2\delta x) = f(0.6) = 0.86603$	from the initial condition
u_3^0	$=$	0	from the boundary condition

The first time-step will find u_j^1. First we note that the boundary condition implies that $u_0^1 = u_3^1 = 0$.

Two uses of the stencil give

$$\begin{aligned} -0.27778u_0^1 + 1.55556u_1^1 - 0.27778u_2^1 = 0.44444u_1^0 + 0.27778(u_0^0 + u_2^0) &= 0.62546 \\ -0.27778u_1^1 + 1.55556u_2^1 - 0.27778u_3^1 = 0.44444u_2^0 + 0.27778(u_1^0 + u_3^0) &= 0.62546 \end{aligned}$$

The implicit nature of this method means that we have to do some extra work to complete the time-step. We must now solve the simultaneous equations

$$\begin{pmatrix} 1.55556 & -0.27778 \\ -0.27778 & 1.55556 \end{pmatrix} \begin{pmatrix} u_1^1 \\ u_2^1 \end{pmatrix} = \begin{pmatrix} 0.62546 \\ 0.62546 \end{pmatrix}$$

In this case there are only two unknowns and it is a simple matter to solve the pair of equations to give $u_1^1 = 0.48949$ and $u_2^1 = 0.48949$.

Your solution

Second time-step:

Answer
The second time-step will find u_j^2. First we note that the boundary condition implies that $u_0^2 = u_3^2 = 0$. Two uses of the stencil give

$$\begin{aligned} -0.27778u_0^2 + 1.55556u_1^2 - 0.27778u_2^2 = 0.44444u_1^1 + 0.27778(u_0^1 + u_2^1) &= 0.35352 \\ -0.27778u_1^2 + 1.55556u_2^2 - 0.27778u_3^2 = 0.44444u_2^1 + 0.27778(u_1^1 + u_3^1) &= 0.35352 \end{aligned}$$

The implicit nature of this method means that we have to do some extra work to complete the time-step. We must now solve the simultaneous equations

$$\begin{pmatrix} 1.55556 & -0.27778 \\ -0.27778 & 1.55556 \end{pmatrix} \begin{pmatrix} u_1^2 \\ u_2^2 \end{pmatrix} = \begin{pmatrix} 0.35352 \\ 0.35352 \end{pmatrix}$$

In this case there are only two unknowns and it is a simple matter to solve the pair of equations to give $u_1^2 = 0.27667$ and $u_2^2 = 0.27667$.

In general

Having now seen some instances with a relatively large δx, we now look at the general case where the space step may be much smaller. In this case there will be a larger system of equations to solve at each time-step than was the case above.

In general, the procedure of moving the unknowns to the left hand side of the equation leads to

$$-\frac{r}{2}u_{j-1}^{n+1} + (1+r)u_j^{n+1} - \frac{r}{2}u_{j+1}^{n+1} = \frac{r}{2}u_{j-1}^n + (1-r)u_j^n + \frac{r}{2}u_{j+1}^n$$

which we apply all the way along the x-axis. That is, we put $j = 1, 2, 3, \ldots, J-1$ in the above expression and hence derive a system of equations for all the u with superscript $n+1$.

$$\begin{pmatrix} 1+r & -\frac{r}{2} & 0 & \ldots & \ldots & 0 \\ -\frac{r}{2} & 1+r & -\frac{r}{2} & & & \\ 0 & -\frac{r}{2} & 1+r & -\frac{r}{2} & & \\ \vdots & & \ddots & \ddots & \ddots & \\ & & & & & -\frac{r}{2} \\ 0 & \ldots & \ldots & 0 & -\frac{r}{2} & 1+r \end{pmatrix} \begin{pmatrix} u_1^{n+1} \\ u_2^{n+1} \\ u_3^{n+1} \\ \vdots \\ u_{J-1}^{n+1} \end{pmatrix} = \begin{pmatrix} \frac{r}{2}\underline{u_0^n} + (1-r)u_1^n + \frac{r}{2}u_2^n + \frac{r}{2}\underline{\underline{u_0^{n+1}}} \\ \frac{r}{2}u_1^n + (1-r)u_2^n + \frac{r}{2}u_3^n \\ \frac{r}{2}u_2^n + (1-r)u_3^n + \frac{r}{2}u_4^n \\ \vdots \\ \frac{r}{2}u_{J-2}^n + (1-r)u_{J-1}^n + \frac{r}{2}\underline{u_J^n} + \frac{r}{2}\underline{\underline{u_J^{n+1}}} \end{pmatrix}$$

The underlined terms on the right-hand side will be known from the boundary conditions. The doubly underlined quantities are "new" at the current time-step and involve the only appearances of $n+1$ on the right-hand side. All the other u approximations at time level $n+1$ are unknown at this stage and appear on the left.

The matrix on the left-hand side of the system has the following properties

- It is **independent of n**. In other words, the same matrix appears at each time-step. (We saw this in the example and exercise above in which the same 2×2 matrix appeared at each of the two time-steps carried out).

- It is **tridiagonal**. That is, the only non-zero entries are either on, or adjacent to, the diagonal. Furthermore, there are only two different values ($\frac{r}{2}$ and $1+r$) which appear. This is good news as far as storage is concerned. Gaussian elimination (seen in HELM 30, for example) works extremely well on tridiagonal matrices.

It is also true that the matrix is **strictly diagonally dominant**. (That is, the diagonal element on each row is greater in size than the sum of the absolute values of the off-diagonal elements on that row.) This means that methods such as Jacobi and Gauss Seidel (see HELM 30 for details) would work very well.

Stability of the Crank-Nicolson scheme

This is the big pay-off when using the Crank-Nicolson method.

Key Point 21

The Crank-Nicolson method is stable for all values of r.

This is excellent news. It means that there is no hideously restrictive constraint on the size of δt.

7. Cost -v- benefit

At a first reading of this Section, it might be tempting to think that the extra effort involved in using Crank-Nicolson (we have to store a set of simultaneous equations, we have to solve them and we have to do this at every time-step) is enough to make the explicit method the winner in a cost-benefit analysis. **But this would be wrong.**

In practical problems involving numerical approximations to parabolic problems the explicit method is rarely good enough. The stability constraint ($r \leq \frac{1}{2}$) imposes such tiny time-steps that it takes a great deal of time for a computer to produce approximations corresponding to even fairly modest values of t. If efficiency is what matters, then Crank-Nicolson beats the explicit approach, and it is worth the extra initial effort formulating a solver (such as those we saw in HELM 30) for the system of equations.

Exercises

1. Consider the function u defined by

   $$u(x,t) = x^3\cos(xt)$$

   Using increments of $\delta x = 0.005$ and $\delta t = 0.01$, and working to 8 decimal places, approximate

   (a) $u_x(2,3)$ with a one-sided forward difference

   (b) $u_{xx}(2,3)$ with a central difference

   (c) $u_t(2,3)$ with a one-sided forward difference

   (d) $u_t(2,3)$ with a central difference.

   State the approximate derivatives to 3 decimal places.

2. The temperature $u(x,t)$ of a metal bar of length $\ell = 3$ at a distance x from one end and at time t is modelled by the partial differential equation

   $$u_t = \alpha u_{xx} \qquad (0 < x < \ell, \quad t > 0)$$

   It is given that the metal has diffusivity $\alpha = 1.6$, that the two ends of the bar are kept at temperature $u = 0$ and that the initial temperature distribution is

   $$u(x,0) = f(x) = x(\ell - x)$$

   Use the explicit difference scheme with $\delta x = 0.75$ and $\delta t = 0.08$ to approximate $u(x,t)$ at $t = \delta t$ and $t = 2\delta t$.

3. The temperature $u(x,t)$ of a metal bar of length $\ell = 1.2$ at a distance x from one end and at time t is modelled by the partial differential equation

   $$u_t = \alpha u_{xx} \qquad (0 < x < \ell, \quad t > 0).$$

   It is given that the metal has diffusivity $\alpha = 2.25$, that the two ends of the bar are kept at temperature $u = 0$ and that the initial temperature distribution is

   $$u(x,0) = f(x) = \sin(\pi x/\ell)$$

   Use the Crank-Nicolson difference scheme with $\delta x = 0.4$ and $\delta t = 0.06$ to approximate $u(x,t)$ at $t = \delta t$ and at $t = 2\delta t$.

Answers

1. The evaluations of u we will need are $u(x,t) = -0.41614684$, $u(x+\delta x,t) = -0.43162908$, $u(x-\delta x,t) = -0.40095819$, $u(x,t+\delta t) = -0.42521885$, $u(x,t-\delta t) = -0.40703321$. It follows that

   (a) $u_x(1,2) \approx \dfrac{-0.43162908 + 0.41614684}{0.005} = -3.096$

   (b) $u_{xx}(1,2) \approx \dfrac{-0.43162908 + 2 \times 0.41614684 - 0.40095819}{0.005^2} = -11.744$

   (c) $u_t(1,2) \approx \dfrac{-0.42521885 + 0.41614684}{0.01} = -0.907$

   (d) $u_t(1,2) \approx \dfrac{-0.42521885 + 0.40703321}{2 \times 0.01} = -0.909$

   to 3 decimal places. (Workings shown to 8 decimal places.)

2. In this case $r = \alpha^2 \delta t/(\delta x)^2 = 0.227556$ so that the numerical scheme can be written

$$u_j^{n+1} = u_j^n + 0.227556(u_{j-1}^2 - 2u_j^n + u_{j+1}^n) = 0.544889u_j^n + 0.227556(u_{j-1}^2 + u_{j+1}^n)$$

   The first stage is to use the given data to find u_j^0

u_0^0	=	0	from the boundary condition
u_1^0	=	$f(\delta x) = f(0.75) = 1.6875$	from the initial condition
u_2^0	=	$f(2\delta x) = f(1.5) = 2.25$	from the initial condition
u_3^0	=	$f(3\delta x) = f(2.25) = 1.6875$	from the initial condition
u_4^0	=	0	from the boundary condition

   The first timestep will find u_j^1. We note that the boundary condition implies that $u_0^1 = u_4^1 = 0$.

$$u_1^1 = 0.544889u_1^0 + 0.227556(u_0^0 + u_2^0) = 0.544889 \times 1.6875 + 0.227556(0 + 2.25) = 1.4315$$
$$u_2^1 = 0.544889u_2^0 + 0.227556(u_1^0 + u_3^0) = 0.544889 \times 2.25 + 0.227556(1.688 + 1.688) = 1.994$$
$$u_3^1 = 0.544889u_3^0 + 0.227556(u_2^0 + u_4^0) = 0.544889 \times 1.6875 + 0.227556(2.25 + 0) = 1.4315$$

   The second timestep will find u_j^2. First we note that the boundary condition implies that $u_0^2 = u_4^2 = 0$.

$$u_1^2 = 0.544889u_1^1 + 0.227556(u_0^1 + u_2^1) = 0.544889 \times 1.4315 + 0.227556(0 + 1.994) = 1.233754$$
$$u_2^2 = 0.544889u_2^1 + 0.227556(u_1^1 + u_3^1) = 0.544889 \times 1.994 + 0.227556(1.432 + 1.432) = 1.738$$
$$u_3^2 = 0.544889u_3^1 + 0.227556(u_2^1 + u_4^1) = 0.544889 \times 1.4315 + 0.227556(1.994 + 0) = 1.233754$$

   where some quantities have been rounded to 6 decimal places.

Answers

3. In this case $r = \alpha\delta t/(\delta x)^2 = 0.84375$ so that the numerical scheme can be written

$$u_j^{n+1} = u_j^n + \frac{0.84375}{2}(u_{j-1}^n - 2u_j^n + u_{j+1}^n + u_{j-1}^{n+1} - 2u_j^{n+1} + u_{j+1}^{n+1})$$

Moving the unknowns to the left of the equation we obtain

$$-0.42188u_{j-1}^{n+1} + 1.84375u_j^{n+1} - 0.42188u_{j+1}^{n+1} = 0.15625u_j^n + 0.42188(u_{j-1}^n + u_{j+1}^n)$$

The first stage is to use the given data to find u_j^0

$$\begin{array}{lcll} u_0^0 & = & 0 & \text{from the boundary condition} \\ u_1^0 & = & f(\delta x) = f(0.4) = 0.86603 & \text{from the initial condition} \\ u_2^0 & = & f(2\delta x) = f(0.8) = 0.86603 & \text{from the initial condition} \\ u_3^0 & = & 0 & \text{from the boundary condition} \end{array}$$

The first time-step will find u_j^1. First we note that the boundary condition implies that $u_0^1 = u_3^1 = 0$. Two uses of the stencil give

$$\begin{array}{rcl} -0.42188u_0^1 + 1.84375u_1^1 - 0.42188u_2^1 = 0.15625u_1^0 + 0.42188(u_0^0 + u_2^0) & = & 0.50067 \\ -0.42188u_1^1 + 1.84375u_2^1 - 0.42188u_3^1 = 0.15625u_2^0 + 0.42188(u_1^0 + u_3^0) & = & 0.50067 \end{array}$$

The implicit nature of this method means that we have to do some extra work to complete the time-step. We must now solve the simultaneous equations

$$\begin{pmatrix} 1.84375 & -0.42188 \\ -0.42188 & 1.84375 \end{pmatrix} \begin{pmatrix} u_1^1 \\ u_2^1 \end{pmatrix} = \begin{pmatrix} 0.50067 \\ 0.50067 \end{pmatrix}$$

In this case there are only two unknowns and it is a simple matter to solve the pair of equations to give $u_1^1 = 0.35212$ and $u_2^1 = 0.35212$.
The second time-step will find u_j^2. First we note that the boundary condition implies that $u_0^2 = u_3^2 = 0$. Two uses of the stencil give

$$\begin{array}{rcl} -0.42188u_0^2 + 1.84375u_1^2 - 0.42188u_2^2 = 0.15625u_1^1 + 0.42188(u_0^1 + u_2^1) & = & 0.20357 \\ -0.42188u_1^2 + 1.84375u_2^2 - 0.42188u_3^2 = 0.15625u_2^1 + 0.42188(u_1^1 + u_3^1) & = & 0.20357 \end{array}$$

The implicit nature of this method means that we have to do some extra work to complete the time-step. We must now solve the simultaneous equations

$$\begin{pmatrix} 1.84375 & -0.42188 \\ -0.42188 & 1.84375 \end{pmatrix} \begin{pmatrix} u_1^2 \\ u_2^2 \end{pmatrix} = \begin{pmatrix} 0.20357 \\ 0.20357 \end{pmatrix}$$

In this case there are only two unknowns and it is a simple matter to solve the pair of equations to give $u_1^2 = 0.14317$ and $u_2^2 = 0.14317$.