

1 Numerical Methods

1.3 Parabolic Partial Differential Equations (7 units)

Part II Numerical Analysis is useful but not essential, since the required background can readily be found in references [1, 2, 3], and elsewhere.

1 Formulation

For times $0 \leq t < \infty$ we wish to solve the diffusion equation

$$\theta_t = \theta_{xx}$$

on the interval $0 \leq x \leq 1$, with boundary conditions

$$\theta(0, t) = f(t) \quad \text{and} \quad \theta(1, t) = 0 \quad \text{for } 0 \leq t < \infty, \quad \text{where } f(t) = (3t - 2)t,$$

and with initial condition

$$\theta(x, t) = 0 \quad \text{for } t \leq 0, \quad 0 \leq x \leq 1.$$

This is the (non-dimensionalised) initial-value problem for the conduction of heat down a lagged bar when the temperature of one end varies in time. The aim is to study the performance of three simple finite-difference methods applied to this problem, for which the numerical solutions can be compared with an analytic one.

2 Analytic solution

Question 1

- (i) To find an analytic solution of the problem first write

$$\theta(x, t) = f(t)(1 - x) + \phi(x, t).$$

Next find the governing equation, boundary conditions and initial condition for $\phi(x, t)$. Thence, with justification, solve for ϕ in terms of a Fourier sine series in x .

- (ii) Deduce, either from the Fourier sine series or otherwise, that as $t \rightarrow \infty$

$$\phi(x, t) \rightarrow \alpha(x)t + \beta(x), \tag{1}$$

where the functions $\alpha(x)$ and $\beta(x)$ are to be identified.

- (iii) Write a program to compute the analytic solution by summing a finite number of terms of the series, or otherwise.
- (iv) Plot θ against x at a few judiciously chosen values of t to illustrate the evolution in time.
- (v) How have you satisfied yourself that the solution has been computed to ‘sufficient’ accuracy?
- (vi) Discuss the evolution of the temperature in terms of the physics.

3 Numerical Methods

Divide $0 \leq x \leq 1$ into N intervals, each of size $\delta x \equiv 1/N$. The aim is to march the solution forward in time for various time steps δt . We consider three schemes.

- (i) Approximate θ_t by a forward difference in time and θ_{xx} by a spatial central difference at the current time, which gives the numerical scheme

$$\frac{\theta_n^{m+1} - \theta_n^m}{\delta t} = (\delta^2 \theta)_n^m \equiv \frac{\theta_{n+1}^m - 2\theta_n^m + \theta_{n-1}^m}{(\delta x)^2},$$

where θ_n^m is an approximation to $\theta(n\delta x, m\delta t)$.

- (ii) Approximate θ_t instead by a central difference in time, so that

$$\frac{\theta_n^{m+1} - \theta_n^{m-1}}{2\delta t} = (\delta^2 \theta)_n^m.$$

In this case you will need scheme (i) in order to make the first step.

- (iii) Modify scheme (i) to

$$\frac{\theta_n^{m+1} - \theta_n^m}{\delta t} = \rho (\delta^2 \theta)_n^{m+1} + (1 - \rho) (\delta^2 \theta)_n^m$$

with $0 < \rho \leq 1$. This is now an *implicit* method, and at each step ($N + 1$) simultaneous equations have to be solved for the θ_n^{m+1} .

Remarks

- (a) The matrix of the simultaneous equations is tridiagonal. Therefore the system may be solved quickly and efficiently by exploiting the sparsity. Your code *should make use of the sparsity*, e.g. the matrix should be stored in an efficient way, and needless multiplications by zero avoided. If you are using MATLAB then `help sparse`, `help spdiags` and `help speye` should help.
- (b) You can check that aspects of your program are working by setting $\rho = 0$ and comparing with the output of scheme (i).

Question 2 It is convenient to introduce the Courant number $\nu = \delta t / (\delta x)^2$.

- (i) First run each finite-difference scheme with $N = 5$ and $\nu = \frac{1}{2}$ and, in the case of scheme (iii), $\rho = \frac{1}{2}$. Plot the solution for representative times. In particular, tabulate and plot the numerical solution θ_n^m , the analytic solution $\theta(n\delta x, m\delta t)$ and the error $\theta_n^m - \theta(n\delta x, m\delta t)$ at $t = 0.1$.
- (ii) Next investigate a range of values of your choice for the parameters ν (for all schemes) and ρ (for scheme (iii)). You might like to start by considering $\nu = \frac{1}{12}, \frac{1}{6}, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}$ and 1, and $N = 5, 20, 80$. In the case of scheme (iii) also consider $\delta t = \mu\delta x$ (i.e. $\nu = \mu/\delta x$) for appropriate values of the constants μ and ρ .
- (iii) Discuss the accuracy and the stability of each scheme, and how these properties vary with N , ν and ρ . For instance, are your results consistent with the theoretical order of accuracy of each scheme, e.g. see [1, 2, 3]? Statements about accuracy and stability should be supported by *selective* reference to your numerical results, displayed as short tables and/or graphs. Relevant theoretical results should be cited *briefly*.

Comment on, and explain, any interesting features, e.g. do you notice anything about the error in the case of scheme (i) with $\nu = \frac{1}{6}$, scheme (iii) with particular choices of ρ and ν , and scheme (iii) with $\rho = \frac{1}{2}$ and $\delta t = \mu\delta x$?

- (iv) Explain, *with justification*, which scheme and parameter values you would recommend to achieve a given level of accuracy using the *least* computing resources. In particular, you should consider the total operation count to achieve a given level of accuracy.
- (v) For your recommended scheme and parameter values, demonstrate that the numerical solution tends to the asymptotic limit (1) as $t \rightarrow \infty$.

References

- [1] Ames, W.F. *Numerical Methods for Partial Differential Equations*, Academic Press.
- [2] Iserles, A. *A First Course in the Numerical Analysis of Differential Equations*, CUP.
- [3] Smith, G.D. *Numerical Solution of Partial Differential Equations: Finite Difference Methods*, OUP.