Technical University of Denmark

Assignment 2

Authors:

Miguel Suau de Castro (s161333)
Michal Baumgartner (s161636)

March 16, 2017

Contents

1 Pure parabolic model problem

In this section we shall study and analyze two different approaches for solving parabolic differential equations. Concretely we will focus on the heat diffusion problem, which can be modeled by the following partial differential equation and its initial and boundary conditions.

$$
\begin{aligned}
u_t &= \kappa u_{xx} \\
u(x, 0) &= \eta(x) \\
u(0, t) &= g_L(t) \\
u(0, t) &= g_R(t)
\end{aligned}
\tag{1}
$$

The first thing we should notice is that, a part from having the usual spatial dependence, the equation also describes how the heat distribution evolves with time. Hence, if we want to solve this equation numerically we need to discretize both time and space, which means that the solution will be represented by a two dimensional grid.

1.1 Matlab implementation of the θ-scheme

Applying euler approximation to the time derivative and the second order central finite difference to the right hand side in equation **??**, an explicit scheme is obtained.

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

where h and k represent the distance between two grid points in space and time respectively.

A more general scheme for solving this problem can be formulated as:

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

where $0 \le \theta \le 1$

It is easy to see that when $\theta = 0$ equation 3 is equivalent to 2. However for any other value of θ the scheme becomes implicit but also more accurate. [[[[The solution at one point in space relies on both its neighbors at the current time and at the previous.]]]!!!

The scheme takes the form of system of equations which can be written in matrix form as:

which is nothing but a system of ordinary differential equations and thus, the solution at a given time can be obtained by inverting the tridiagonal matrix in the left hand side of equation **??**. Consequently, using a loop the global solution can be computed sequentially at every point in the time grid.

The code snippet below shows the implementation of this scheme in MATLAB where the function given by equation 2 in the assignment specifications is used to compute the initial and boundary conditions.

```matlab
function U = parabolicSolver(boundaryFun,h,k,theta,mu)
M = ceil(2/h);
N = ceil(1/k);
A0 = diag(-2*ones(M+1,1)) + diag(ones(M,1),-1) + diag(ones(M,1),1);
A0(1,:) = zeros(1,M+1);
A0(end,:) = zeros(1,M+1);
I = eye(M+1);
x = linspace(-1,1,M+1);
t = linspace(0,1,N+1);
U(:,1) = boundaryFun(x,0);
g(:,1) = [boundaryFun(-1,0); zeros(M-1,1); boundaryFun(1,0)];

for k=2:N
    g(:,k) = [boundaryFun(-1,t(k)); zeros(M-1,1); boundaryFun(1,t(k))
        ];
    U(:,k) = (I-theta*mu*A0)\((I+(1-theta)*mu*A0)*U(:,k-1)+g(:,k)-g
        (:,k-1));
end


end
```

1.2 Analytical analysis

Once the the scheme for solving the problem numerically has been built, it might be interesting to make an analysis on the system and draw some conclusions about how large the grid should be for the solver to be stable, what is its order of consistency or under what circumstances it satisfies the discrete maximum principle.

1.2.1 Forward time central space

As mentioned, for $\theta = 0$ equation 3 corresponds to the explicit forward time and central space scheme. We first check that the finite difference approximation is consistent, that is, the local truncation error approaches to 0 as the grid size increases.

$$\tau(x,t) = \frac{u(x,t+k) - u(x,t)}{k} - \frac{\kappa}{h^2}(u(x-h,t) - 2u(x,t) + u(x+h,t)) \quad (4)$$

by applying the Taylor series expansions about u(x,t)

$$\tau(x,t) = \left(u_t + \frac{1}{2}ku_{tt} + \frac{1}{6}k^2 u_{ttt} + ...\right) - \kappa\left(u_x x + \frac{1}{12}h^2 u_{xxxx} + ...\right) \quad (5)$$

and since we know that $u_t = \kappa u_x x$ and $u_t t = \kappa^2 u_x x$ then

$$\tau(x,t) = \left(\frac{\kappa^2}{2}k - \frac{\kappa}{12}h^2\right) u_{xxxx} + \mathcal{O}(k^2 + h^4) \quad (6)$$

where the dominant term depends on k and h^2 which means that the scheme is consistent of order $\mathcal{O}(k + h^2)$.

Besides, for $\mu = 1/6$

$$\tau(x,t) = h^2\left(\frac{\kappa}{12} - \frac{\kappa}{12}\right) u_{xxxx} + \mathcal{O}(k^2 + h^4) \quad (7)$$

the first term in the right hand side of equation 7 disappears and the order becomes $\mathcal{O}(k^2 + h^4)$.

Secondly, in order to find the stability of the method we need to check when the eigenvalues of the matrix A in the right hand side of equation **??** lie within the unit circle.

According to 2.23 in Randall Leveque the eigenvalues of A are given by

$$\lambda_p = 2(cos(p\pi h) - 1) \qquad for \qquad p = 1, 2, ..., m \quad (8)$$

Therefore, the system would be absolutely stable when $|1 + \mu\lambda_p| \leq 1$ and since the smallest λ_p is approximately -4 we have that the region is defined by:

$$-2 \leq -4\mu \leq 0 \quad (9)$$

which means that for the system to be stable the following relation must be satisfied

$$\frac{\kappa k}{h^2} \leq \frac{1}{2} \quad (10)$$

4

1.2.2 $\theta = 1/2 + h^2/(12k\kappa)$

According to what we saw when developing the θ-scheme we should expect a higher order for this particular choice of θ than in the previous case. The LTE is in this case

$$
\tau(x,t) = \frac{u(x,t+k) - u(x,t)}{k} - \frac{\kappa}{h^2}\left(\left(\frac{1}{2} - \frac{h^2}{12k\kappa}\right)(u(x-h,t)-2u(x,t)+u(x+h,t))+\right.
$$
$$
\left.\left(\frac{1}{2} + \frac{h^2}{12k\kappa}\right)(u(x-h,t+k) - 2u(x,t+k) + u(x+h,t+k))\right) \quad (11)
$$

by taking Taylor expansions in the expression above

$$
\tau(x,t) = \left(u_t + \frac{1}{2}ku_{tt} + \frac{1}{6}k^2 u_{ttt} + ...\right) - \frac{\kappa}{h^2}\left(\left(\frac{1}{2} + \frac{h^2}{12k\kappa}\right)(h^2 u_{xx} + \frac{1}{12}h^4 u_{xxxx} + ...)+\right.
$$
$$
\left.\left(\frac{1}{2} + \frac{h^2}{12k\kappa}\right)(h^2 u_{xx} + h^2 u_{xxt} + ...)\right) \quad (12)
$$

and since we know that $u_t = \kappa u_x x$, $u_{tt} = \kappa^2 u_{xx}$ and $u_{xxt} = \kappa u_{xxxx}$ then

$$
\tau(x,t) = \left(\frac{\kappa h^2}{2} - \frac{h^2}{12}\right)u_{xxxx} + \frac{k^2}{6}u_{ttt} \quad ... \quad (13)
$$

where the dominant term depends on k^2 and h^2 which means that the scheme is consistent of order $\mathcal{O}(k^2 + h^2)$.

1.3 Numerical evaluation

After deriving the analytical expressions of the method's convergence, we shall check the rates obtained for the two values of θ. To do so, we use the solver implemented in MATLAB, using a loop we vary the time and the space grids accordingly so that the system remains within the stability region, and we compute the local truncation error (LTE) at every iteration.

For the choice of $\theta = 0$, figure **??** represents the variation of the LTE when changing the size of h and k. In this case, we set the value of k to be equal to $\frac{h^2}{2}$ so that the eigenvalues of the system lie within the unit circle. With the inten The dashed lines, which represent the theoretical order of the system are parallel to the empirical curves $O(h^2)$ and $O(k)$.

5

When $k = h^2/6$ we see that the the local truncation error is upper bounded by the theoretical limits of $O(h^4)$ and $O(k^2)$ (figure **??**).

On the other hand, for the especial choice of $\theta = 1/2 + h^2/(12k\kappa)$ the plots also follow the analytical expressions obtained in the previous section.

A   Appendix

A.1   2-point Boundary Value Problems