Uppsala Universitet
Institutionen för informationsteknologi
Avdelningen för teknisk databehandling

Analysis of Numerical Methods

Compulsory assignments

The assignments in Analysis of Numerical Methods are compulsory. There is one assignment for each module of the course. The results should be presented at the concluding seminar of each module. *The results of each student group must be presented, even if the group has not been able to solve the assignment completely.*

The presentation consists of two parts, one written and one oral. During the seminar, each group member should be prepared to orally present the results of the group. The written presentation (one per student group) is to be handed in at the beginning of the seminar, and shall contain:

- A brief statement of the problem to be solved.
- A description of how the group solved the problem. (This should focus on how the group approached the problem, and on the methods and arguments used.)

For the experiments, supply in addition:

- Program listings.
- Presentation of the results of the experiments (graphs, tables, or what else may be appropriate for each experiment).
- Comments to the results. Relate the results to the theory, and point at how the results did (or did not) agree with what you had expected from a theoretical point of view.

If the group is unable to solve the assignment, partly or completely, the presentation should focus on the difficulties encountered, and on how the group attempted to address these.

Advise from the teachers is offered according to the "open door" principle. The teachers *encourage* the students to come and discuss the problems, since this is an important part of the learning process. You are also encouraged to do your own experiments with the assignments.

Experiments with difference methods

Each assignment has two parts, one experiment and one theoretical problem.

To make a *numerical experiment* means to apply a numerical method to a model problem with known solution. The parameters of the method are varied, and the computed results are evaluated. This is used as a complement to a theoretical analysis.

Numerical experiments can also be used for *demonstrating* different properties of numerical methods. In this course, experiments will be used for this purpose. In order to avoid time consuming programming, Matlab will be used.

Below, the general setting of the experiments of the three first assignments is described. The details of each experiment are given in the subsequent sections.

- In the three first experiments, the model problem is:

$$
\begin{aligned}
u_t &= u_x, \qquad 0 < x < 1, \quad 0 < t, \\
u(x,0) &= \sin 2\pi x,
\end{aligned}
$$

  with additional boundary conditions. The difference method to be studied is the Leap-Frog method (which is described in the literature). For simplicity, we use the *exact* solution at the first *two* time levels.

- The parameters to be varied in the experiments are the step sizes (Δx in space and Δt in time). Unless another time limit is specified, the computations are to be continued until we reach a time level where $t \approx 5$. The presentation of the experimental results shall consist in a plot of the grid function v, for different time levels, with the analytic solution u drawn in the same graph for comparison. Moreover, $\|v\|$ och $\|v-u\|$ should be computed for the same time levels. The discrete $\mathcal{L}_2$ norm should be used (see the literature). *You* decide which time levels to present. If nothing interesting happens to the numerical solution, then it is sufficient to show the solution and norms for the final time level.

- The results of the experiments shall be evaluated with respect to the *quality* of the numerical solution:

  - How good is the solution compared to the analytical one?
  - How is the quality of the solution affected by changes in the step sizes?
  - How is the quality of the solution affected by changes in the *relation* between the step sizes, the parameter $\lambda = \Delta t/\Delta x$?

- Note in particular if unexpected phenomena show up.

  - At what time level could you see the first signs of the phenomenon?
  - At what position in space did the first signs of the phenomenon show up?
  - (How) does the phenomenon propagate?

Assignment 1:
Basic concepts

Experiment

We begin by an experiment where the data are chosen so that the solution is nice. The emphasis is on getting used to working with Matlab. Nevertheless, make an evaluation of the experimental results. The questions about the choice of parameter values are interesting in the nice cases too.

Use periodic boundary conditions: $u(x,t) = u(x+1,t)$. Try five different choices of parameter values:

a) $\Delta x = 0.1$, $\Delta t = 0.049$

b) $\Delta x = 0.1$, $\Delta t = 0.098$

c) $\Delta x = 0.05$, $\Delta t = 0.0245$

d) $\Delta x = 0.05$, $\Delta t = 0.049$

e) $\Delta x = 0.1$, $\Delta t = 0.1$

Theory

Leap-Frog, which is used in the experiment, has order of accuracy (2, 2). Make a theoretical analysis that proves this.

What happens when $\Delta t = \Delta x$?

Assignment 2:
Stability of initial value problems

Experiment

(i) Solve the same problem as in Experiment 1, but choose the following parameter values:

a) $\Delta x = 0.1$, $\Delta t = 0.102$

b) $\Delta x = 0.05$, $\Delta t = 0.051$

c) $\Delta x = 0.025$, $\Delta t = 0.0255$

Change the initial data to a sawtooth wave: $u = z$ for $z < \frac{1}{2}$ and $u = 1 - z$ for $z > \frac{1}{2}$, where $z = x - \lfloor x \rfloor$. Solve again the problem with the parameters above.

(ii) Change the boundary condition to $u(1, t) = 0$. This condition is sufficient for the PDE problem. Leap-Frog, however, needs an additional boundary condition, at $x = 0$. We choose to compute a value at $x = 0$, by extrapolation from the interior of the domain: $v_0^{n+1} = 2v_1^n - v_2^{n-1}$. (Note: In Matlab the index starts from 1 and then we must change the boundary condition accordingly.)

The boundary conditions are chosen so that the analytical solution is less regular than before. This causes an oscillation in the numerical solution. In order to handle this, we introduce an additional, *dissipative* term in Leap-Frog:

$$v_j^{n+1} = v_j^{n-1} + 2\Delta t D_0 v_j^n - \delta(\Delta x)^4 (D_+ D_-)^2 v_j^{n-1}.$$

Use this method at the grid points $x_2, \ldots, x_{N-2}$, and the ordinary Leap-Frog method at the grid points x_1 and x_{N-1}. (Use the original initial data). The aim is to find an optimal δ for a given λ. Choose $\Delta x = 0.05$ and $\Delta t = 0.04$. Try different choices of δ between 0 and 0.1. Compute until $T \approx 0.8$ and use the original initial condition $u(x, 0) = sin(2\pi x)$.

Theory

Experiment 2(i) shows that Leap-Frog is unstable for $\lambda > 1$. (Theoretically, we know that Leap-Frog is unstable also for $\lambda = 1$.) Now, let us modify the method, keeping the approximation in time, but changing to a fourth-order approximation in space. This yields the Leap-Frog (2, 4) method, which for our model problem has the following form:

$$\frac{u_j^{n+1} - u_j^{n-1}}{2\Delta t} = aD_0 \left(I - \frac{h^2}{6} D_+ D_- \right) u_j^n.$$

a) Use the Fourier method to investigate the stability of this method.

b) Assume that we want to apply Leap-Frog (2, 4) to the mixed initial boundary value problem below. At what points in space is it necessary to introduce additional boundary conditions and suggest conditions for these points.

$$\begin{aligned} u_t &= u_x, \qquad 0 \leq x < 1, \quad 0 < t \leq T \\ u(x,0) &= f(x) \\ u(1,t) &= 0. \end{aligned}$$

Assignment 3:
Stability of initial boundary value problems

Experiment

(i) For a stable difference method, the speed of convergence is given by the order of accuracy. The theoretical analysis of the order of accuracy is carried out under the assumption that the solution has a sufficient number of continuous derivatives. Thus, the theoretical speed of convergence may be higher than the one actually achieved in practice.

One way of experimentally investigating the speed of convergence is the following:

- Solve the problem with step sizes Δx och Δt, until $t = T$. Let ε_1 be the error norm at the final time level.
- Divide the step sizes by two, and solve the problem again, until $t = T$. Let ε_2 be the error norm at the final time level.
- By comparing ε_1 and ε_2, we can find the actual value of $\min(p, q)$, where p is the order of accuracy in time, and q is the order of accuracy in space.

Conduct experiments according to this strategy, for the two cases:

a) periodic boundary conditions

b) $u(1, t) = 0$, $v_0^{n+1} = 2v_1^n - v_2^{n-1}$.

Let $T \approx 0.8$. Choose the initial step sizes as you like. (Repeat the experiment for other choices of initial step sizes, to see if this choice matters.)

(ii) For non-periodic problems, the difference method needs more boundary conditions than the PDE problem. Let $u(1, t) = \sin 2\pi(1 + t)$, and choose as extra boundary condition:

a) $v_0^{n+1} = v_1^{n+1}$

b) $v_0^{n+1} = 2v_1^{n+1} - v_2^{n+1}$

c) $v_0^{n+1} = v_1^n$

d) $v_0^{n+1} = 2v_1^n - v_2^{n-1}$

Use the same strategy as in **(i)**. Let $T \approx 5$. Begin with $\Delta x = 0.1$, $\Delta t = 0.098$. Study both stability and convergence speed.

Theory

Show that Leap-Frog with straight extrapolation

$$v_0^{n+1} = v_1^{n+1}$$

is unstable. Use the GKSO theory.

Challenge

Show that Leap-Frog with straight extrapolation of higher order

$$(hD_+)^p v_0^{n+1} = 0, \qquad p \geq 2,$$

is unstable.

Assignment 4:
Efficiency of difference methods

Experiment

The problem

$$\begin{aligned} u_t &= 0.4u_{xx}, \qquad 0 < x < 1 \\ u(0,t) &= 0 \\ u(1,t) &= 1 \\ u(x,0) &= \sin(5\pi x/2) \end{aligned}$$

has a steady state solution when $t \to \infty$.

a) Use the Crank-Nicolson scheme,

$$u_j^{n+1} = u_j^n + 0.4\Delta t D_+ D_- \left(\frac{u_j^n + u_j^{n+1}}{2} \right),$$

for finding the steady state solution. Let $\Delta x = 0.01$ and iterate forwards in time until the norm of the difference between the solutions of the two latest iterations is smaller than 10^{-4}. Take large time steps. You will need to solve a triangular system at each time level. Declare you Matlab matrices to be `sparse`. Then, Matlab will solve the systems efficiently. Since the matrix is constant, the LU factorization can be done *once*, before the time-marching begins. Subsequently, in each time step, the system can be solved by means of a forward and a back substitution.

Try to find the Δt that gives the smallest number of iterations. Measure the execution time (use the Matlab functions `clock` and `etime`).

Change the initial condition to $\sin(4\pi x)$, put periodic boundary conditions, and repeat the experiment.

b) Use the Euler method with forward difference,

$$u_j^{n+1} = u_j^n + 0.4\Delta t D_+ D_- u_j^n.$$

Carry out 1000 iterations with $\Delta x = 0.01$, using the largest possible Δt. Note the norm of the difference between the solutions of the final two iterations. Measure the execution time. Repeat the experiment, but continue to iterate until the numerical solution has converged to the steady analytical solution with an error less than 10^{-4}. Measure the execution time for the computation.

Theory

In order to conduct the experiment above, the value of Δt has to be chosen such that the respective difference methods become stable. Perform the stability analysis for the case with periodic boundary conditions.

Challenge

Theoretically derive the Δt that would give the minimal number of iterations with Crank-Nicolson, in the case $u(x, 0) = \sin(4\pi x)$, periodic boundary conditions. How many iterations would then be needed for convergence to the accuracy specified above?

Carry out the same analysis for the Euler *backward* scheme, i.e., the implicit version of Euler's method.

Assignment 5:
Application to nonlinear fluid problem

Experiment

Air flow (simplified to one space dimension) can be described by the hyperbolic system

$$\begin{pmatrix} \rho \\ m \\ e \end{pmatrix}_t + \begin{pmatrix} m \\ \rho u^2 + p \\ (e+p)u \end{pmatrix}_x = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix},$$

where the unknowns are ρ (air density), m (momentum density) och e (total energy density). The air velocity u is computed as $u = m/\rho$. Under the assumption of an ideal gas, the pressure p is given by

$$p = (\gamma - 1)(e - \frac{1}{2}m^2/\rho),$$

where $\gamma = 1.4$. That system is called the 1D Euler equations and is an example for a conservation law. The conservation of mass, momentum and energy is described. The system is in conservative form, i.e. $U_t + F(U)_x = 0$, where the vectors U and $F(U)$ can be identified from the system of equations above.

Assume the following initial data:

$$U(0,x) = \begin{cases} (8,0,25)^T & x < 0 \\ (1,0,2.5)^T & x > 0 \end{cases}$$

Solve this problem on the interval $-1 < x < 1$, until time $T = 0.4$. Use the Lax-Friedrichs scheme

$$U_j^{n+1} = \frac{1}{2}(U_{j+1}^n + U_{j-1}^n) - \Delta t D_0 F(U_j^n)\,,$$

and with approximately 400 grid points in space. Use straight extrapolation as boundary condition at both boundaries. Try experimentally to find a value of Δt that generates a good solution with a minimum of smeared discontinuities. For stability, the value of Δt should be chosen such that $\Delta t \leq 0.4\Delta x$.

Below, the solution to the problem is shown at some time $t > 0$. Note that you cannot expect to get such sharp resolution with the method used in our experiment.

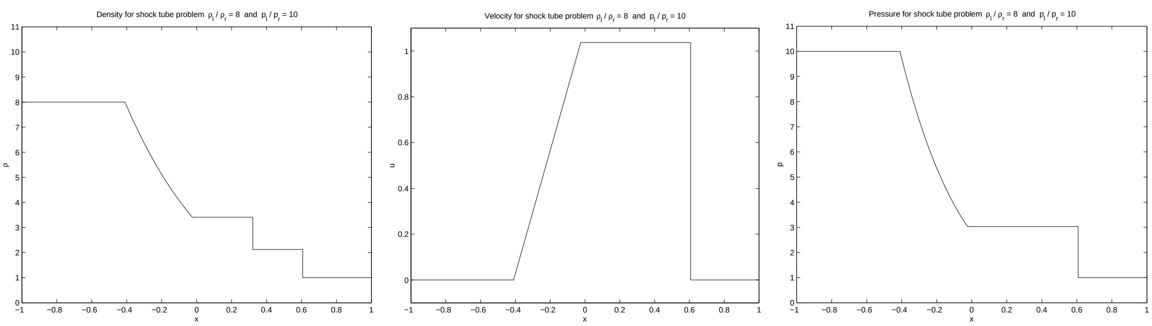

Fig. Solution to the air flow problem

Theory

The upper limit of Δt recommended above is motivated by the fact that the Jacobian matrix $\partial F/\partial U$ for this problem has spectral radius approximately 2.5 (you need not compute the spectral radius). Hyperbolic systems can be modeled by the model problem $w_t + aw_x = 0$, where a is the spectral radius of the Jacobian matrix (needs not but may with pleasure be motivated). Therewith, analyse stability for the Lax-Friedrichs scheme. You may assume periodic boundary conditions.

Challenge

Show that the Lax-Friedrichs scheme can be written in conservation form. Show that the Lax-Friedrichs scheme discretizes the parabolic PDE $U_t + F(U)_x = \delta\, U_{xx}$, where δ is the artificial viscosity. Identify δ. What is the condition for consistency of the Lax-Friedrichs scheme?

Remark

The solution with the Lax-Friedrichs scheme does not have overshoots, because the scheme is total variation diminishing (TVD). But the method is rather dissipative. Starting from TVD schemes, more accurate methods have been developed, which have had great success in computational fluid dynamics. A 2nd order TVD finite volume method is shortly described in assignment 2 of the course Numerical Analysis II (W), fall 2002, cf.
`http://www.it.uu.se/edu/course/homepage/numW2/ht02/assignment2.html`.
There is more information on the homepage of the NGSSC CFD course
`http://user.it.uu.se/~bernd/cfd03/module_5.html`.

Assignment 6:
Iterative methods for elliptic problems

Experiment

The model problem

$$\begin{aligned} -\phi'' &= (2\pi)^2\sin(2\pi x) + (4\pi)^2\sin(4\pi x), \qquad 0 < x < 1, \\ \phi(0) &= \phi(1) = 0, \end{aligned}$$

is to be solved with the difference approximation specified in Chapter 12 of the compendium, Multigrid Methods, chapter 2. For the solution of the resulting linear system, we make a comparison between damped Jacobi with $\omega = 1/2$, and TGM with S_1, R och P as in Chapter 12, Multigrid Methods, chapter 3. Below, u denotes the exact solution to the linear system, i.e., $u = A^{-1}f$. The approximate solution obtained by the iterative method is denoted by v.

Conduct the following experiments in Matlab. Use the matrix formulation of the methods, and take advantage of Matlab's matrix operations. Make sure that the matrices are declared as `sparse`, so that the operations are carried out efficiently.

a) Solve $Au = f$ with damped Jacobi, $h = 0.1$. Create a random initial vector $v^{(0)}$, and save it so that it can be reused for TGM. First, study the error $v^{(j)} - u$ visually, for $j = 1, 2,..$ until the error curve no longer looks oscillating. How many iterations were needed? Then, go on until $\|v^{(j)} - u\|_2 < 10^{-6}$. Note the number of iterations j.

b) Solve $Au = f$ with TGM, $h_1 = 0.1$. Use $\nu = 1, 2, 3, 4$, and 5, respectively. For each value of ν, perform TGM iterations until $\|v^{(j)} - u\|_2 < 10^{-6}$. How many TGM iterations were required in each case?

c) Repeat a) and b) with $h = 0.05$.

Theory

a) Use Matlab to compute $\|S_1\|_2$ and $\|G(3)\|_2$, for $h = 0.1$ and $h = 0.05$, respectively.

b) Demonstrate that the result in a) is in agreement with the theoretical values given by Formulae (2.6) och (4.20) in chapter 12, Multigrid Methods, of the compendium.

c) Use the values from a) to predict the number of iterations that would be needed, with damped Jacobi and TGM, respectively, in order to reach $\|v^{(j)} - u\|_2 < 10^{-6}$. Consider both $h = 0.1$ and $h = 0.05$.

Challenge

Extend the two-grid method TGM to a multigrid method MGM. Test MGM for solving the model problem with $h = 1/128$ and 7 grids to reach $\|v - u\|_2 < 10^{-6}$? Compare the computing times for TGM and MGM. Draw conclusions. Is the improvement, i.e. $\|v^{(k+1)} - v^{(k)}\|_2$, a good measure to check convergence for TGM and MGM?