Technische Universität München
Institut für Informatik
Dr. Miriam Mehl
Dr. Sergey Pankratov

WT 2002/2003

Introduction to Scientific Computing

Final Exam, Febraury 14th 2003

Name: ________________________________

- The exam consists of four problems and is divided into two parts:
  - The first part (problem 1) is without materials any like notes, books etc. at all. You have 20 minutes to accomplish this part.
  - For the second part (problems 2 through 4), you may use books, lecture notes etc., but no electrical devices like calculators, mobile phones, ... You have 70 minutes for this part.

  You have to hand in your answers for part one before you receive the second part of the questions and may start using media.
- Part one consists of page 2, part two is on pages 3 to 7. Please check your copy for completeness!
- Please, insert your answers in the gaps on the worksheet – the space provided should be sufficient for the expected answers. Please advise if you need some more sheets.
- The level of difficulty of the questions is quite different, and we did not sort them with respect to difficulty. Therefore, if you have problems with some question, just proceed to the next part before loosing too much time.

1) General questions

a) What are the three possible tasks of modelling and simulation? **(3 pts)**

- *stating if there is a solution for a given problem (e.g. graph theory),*
- *computing some scenario (e.g. fluid dynamics),*
- *optimization of parameters in technical applications (e.g. reactors, cooling processes)*

alternative answer:

- *gaining insight in the mechanisms of known processes,*
- *predicting unknown processes,*
- *optimizing technical processes.*

b) Give short descriptions of the following three complications for numerical methods for ODEs. Indicate for each of them, whether it can be removed by more suitable numerical methods. If so, name the possible remedies. **(6pts)**

- *ill-conditioned problems:*
  - *small changes of input data cause big changes in the* exact *solution of the model (discontinuous dependency of output from input),*
  - *cannot be improved by numerical methods, inherent problem of the model!*
- *instabilities:*
  - *big errors (often oscillating) in the approximative numerical solution of a model caused by agglomeration of small errors over many discretization steps,*
  - *can be avoided by suitable discretization of the model equations (e.g. being careful with central differences)*
- *stiffness:*
  - *unsatisfying approximative numerical solutions for discretization stepsize above a certain limit due to properties of the model,*
  - *can be removed in most cases by replacing explicit discretization schemes by implicit ones*

c) Why are Jacobi- and Gauß-Seidel-Iterations called 'smoothers'? How is this property exploited in multigrid methods?

- *High frequency parts (with respect to the stepsize of the grid) are reduced very fast, whereas low frequeny parts are reduced much slower. Thus, after a few iterations, we get a still quite big, but very smooth error.*

- *The multigrid idea is based on the following two observations:*
  - *Smooth errors can be represented well on a coarser grid, too.*
  - *On a coarser grid, the error is less smooth with respect to the stepsize of the grid and can thus be reduced faster by a smoother on this grid than on the original fine grid.*

  Thus, multigrid combines iteration steps of smoothers on different grid levels (with different stepsizes) to get faster convergence to the exact solution.

2) Continous Models: PDE

Consider the two-dimensional Laplacian equation

$$p_{xx} + p_{yy} = 0.$$

a) Use the assumption

$$p(x, y) = p_1(x) \cdot p_2(y)$$

and derive two separate ordinary differential equations for p_1 and p_2.

$p_{xx}(x, y) = p_1''(x) \cdot p_2(y), \quad p_{yy} = p_1(x) \cdot p_2''(y),$

$\Rightarrow p_1''(x) \cdot p_2(y) + p_1(x) \cdot p_2''(y) = 0 \quad \forall x, y,$

$\Rightarrow \frac{p_1''(x)}{p_1(x)} = -\frac{p_2''(y)}{p_2(y)} \quad \forall x, y,$

$\Rightarrow$ *there is a constant* c *with* $\frac{p_1''(x)}{p_1(x)} = c$ *and* $\frac{p_2''(y)}{p_2(y)} = -c,$

$\Rightarrow p_1''(x) = c \cdot p_1(x), \quad p_2''(y) = -c \cdot p_2(y).$

b) Transform the second order ODE

$$y''(x) = ay(x)$$

into a system of first order ODEs.

Define the vector

$$\begin{pmatrix} y(x) \\ z(x) \end{pmatrix} := \begin{pmatrix} y(x) \\ y'(x) \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} y'(x) \\ z'(x) \end{pmatrix} = \begin{pmatrix} z(x) \\ ay(x) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ a & 0 \end{pmatrix} \begin{pmatrix} y(x) \\ z(x) \end{pmatrix}.$$

c) Compute the eigenvalues and eigenvectors of the resulting system matrix from b).

eigenvalues $\lambda_{1,2}$*:*

$$\begin{vmatrix} -\lambda & 1 \\ a & -\lambda \end{vmatrix} = \lambda^2 - a,$$

$$\Rightarrow \lambda_1 = \sqrt{a}, \quad \lambda_2 = -\sqrt{a}.$$

eigenvectors:

for λ_1*:* $\begin{pmatrix} -\sqrt{a} & 1 \\ a & -\sqrt{a} \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Rightarrow \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} = \alpha \begin{pmatrix} 1 \\ \sqrt{a} \end{pmatrix}.$

for λ_2*:* $\begin{pmatrix} \sqrt{a} & 1 \\ a & \sqrt{a} \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Rightarrow \begin{pmatrix} e_1 \\ e_2 \end{pmatrix} = \alpha \begin{pmatrix} 1 \\ -\sqrt{a} \end{pmatrix}.$

d) If λ_1 and λ_2 are the eigenvalues computed in c), $\vec{x}_1$ and $\vec{x}_2$ are the respective eigenvectors, the solution of the system of ODEs from b) is

$$\begin{pmatrix} y \\ z \end{pmatrix} = \alpha \vec{x}_1 e^{\lambda_1 x} + \beta \vec{x}_2 e^{\lambda_2 x}.$$

With the help of this result, derive a solution of the two-dimensional Laplacian equation.

Using the result of c), we get the solution of the second order differential equation from b):

$y(x) = \alpha e^{\sqrt{a}x} + \beta e^{-\sqrt{a}x}, \quad \alpha, \beta$ *real numbers.*

As for p_1 *and* p_2*, we computed the same type of second order differential equation as given for* y*, we get analoguously:*

$p_1(x) = \alpha e^{\sqrt{c}x} + \beta e^{-\sqrt{c}x}, \quad p_2(y) = \gamma e^{\sqrt{-c}y} + \delta e^{-\sqrt{-c}y}, \alpha, \beta, \gamma, \delta$ *real numbers.*

Thus, the solution for the original Laplacian equation for p *is*

$p(x, y) = (\alpha e^{sqrtcx} + \beta e^{-sqrtcx}) \cdot \left(\gamma e^{\sqrt{-c}y} + \delta e^{-\sqrt{-c}y}\right), \quad \alpha, \beta, \gamma, \delta$ *real numbers.*

3) Finite Differences and Fast Iterative Solvers for SLE

Consider the one-dimensional Laplacian equation

$$y'' = 0 \text{ in }]0; 1[, y(0) = y(1) = 0.$$

a) Give the exact solution of this ODE:

$y(x) = 0 \quad \forall x \in [0; 1].$

b) Use the standard 3-point-stencil

$$y_i'' \approx \frac{y_{i-1} - 2y_i + y_{i+1}}{h^2}$$

on an equidistant grid with stepsize $h = \frac{1}{4}$ and establish the associated discretized equations. Write the equations in the form $Ax = b$:

$$16 \begin{pmatrix} -2 & \vdots & 1 & \vdots & 0 \\ \cdots & & \cdots & & \cdots \\ 1 & \vdots & -2 & \vdots & 1 \\ \cdots & & \cdots & & \cdots \\ 0 & \vdots & 1 & \vdots & -2 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 0 \\ \cdots \\ 0 \\ \cdots \\ 0 \end{pmatrix}$$

c) Fill in the Jacobi iteration for the above SLE in the following scheme:

$$\vec{y}^{(k+1)} = y^{(k)} + \begin{pmatrix} -1 & \vdots & \frac{1}{2} & \vdots & 0 \\ \cdots & & \cdots & & \cdots \\ \frac{1}{2} & \vdots & -1 & \vdots & \frac{1}{2} \\ \cdots & & \cdots & & \cdots \\ 0 & \vdots & \frac{1}{2} & \vdots & -1 \end{pmatrix} \vec{y}^{(k)} = \begin{pmatrix} 0 & \vdots & \frac{1}{2} & \vdots & 0 \\ \cdots & & \cdots & & \cdots \\ \frac{1}{2} & \vdots & 0 & \vdots & \frac{1}{2} \\ \cdots & & \cdots & & \cdots \\ 0 & \vdots & \frac{1}{2} & \vdots & 0 \end{pmatrix} \vec{y}^{(k)}.$$

d) Starting with

$$\vec{y}^{(0)} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix},$$

perform the first two Jacobi iterations for the SLE from a). Derive a general formula for the iterates $y^{(k)}$.

Hint: You will find two formulas, one for all $y^{(2m)}$ and one for all $y^{(2m+1)}$, $m = 0, 1, 2, \ldots$.

$$\vec{y}^{(1)} = \begin{pmatrix} \frac{1}{2} \\ \cdots \\ 1 \\ \cdots \\ \frac{1}{2} \end{pmatrix}, \vec{y}^{(2)} = \begin{pmatrix} \frac{1}{2} \\ \cdots \\ \frac{1}{2} \\ \cdots \\ \frac{1}{2} \end{pmatrix},$$

$$\vec{y}^{(2m)} = 2^{-m} \begin{pmatrix} 1 \\ \cdots \\ 1 \\ \cdots \\ 1 \end{pmatrix}, \vec{y}^{(2m+1)} = 2^{-m} \begin{pmatrix} 1 \\ \cdots \\ \frac{1}{2} \\ \cdots \\ 1 \end{pmatrix}.$$

How many iterations do you have to perform if you prescribe an accuracy limit $\epsilon = 2^{-10}$ (per component of $\vec{y}$)?

If we copmpare the formula for $\vec{y}^{2m}$ and $\vec{y}^{2m+1}$ with the exact solution $\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$, we get that we stay below the accuracy limit for $m \geq 10$. Thus, we have to perform 20 iterations.

e) Now, try to solve the SLE from a) with the help of a two grid method:

- Using $\vec{y}^{(1)}$ from d), compute the residual after one Jacobi iteration:

$$r^{(1)} = b - A\vec{y}^{(1)} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} - 16 \begin{pmatrix} -2 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -2 \end{pmatrix} \begin{pmatrix} \frac{1}{2} \\ 1 \\ \frac{1}{2} \end{pmatrix} = \begin{pmatrix} 0 \\ \cdots \\ 16 \\ \cdots \\ 0 \end{pmatrix}.$$

- Restrict the residual to the coarse grid with stepsize $H = \frac{1}{2}$ according to the following formula ('Full Weighting'):

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \rightarrow \left(\frac{1}{4}x_1 + \frac{1}{2}x_2 + \frac{1}{4}x_3 \right).$$

$$r_g^{(1)} = \left(\frac{1}{4} \cdot 0 + \frac{1}{2} \cdot 16 + \frac{1}{4} \cdot 0 \right) = (8).$$

- Establish the coarse grid equation for the correction c_g (using the same discretization scheme as for the fine grid):

  $4\,(-2)\,c_g = (8)$

- Solve the coarse grid equation:

  $c_g = (-1)$

- Interpolate the resulting correction using linear interpolation:

$$\vec{c} = \begin{pmatrix} \frac{1}{2} \cdot (0-1) \\ -1 \\ \frac{1}{2} \cdot (-1+0) \end{pmatrix} = \begin{pmatrix} -\frac{1}{2} \\ \cdots \\ -1 \\ \cdots \\ -\frac{1}{2} \end{pmatrix}.$$

- Compute the new fine grid approximation:

$$\vec{y}^{MG} = \vec{y}^{(1)} + \vec{c} = \begin{pmatrix} 0 \\ \cdots \\ 0 \\ \cdots \\ 0 \end{pmatrix}.$$

How many iterations of this two grid method do you have to perform to reach the accuracy limit $\epsilon = 2^{-10}$ (per component of $\vec{y}$)?

We get the exact solution already after the first iteration of the two-grid scheme.

4) Grid Generation

Consider the following computational domain with given grid points (+).

Sketch the construction of the Delaunay triangulation in the above graph. Use different colours or different line types (dashed, dotted, solid, ...) to distinguish the Voronoi diagram from the resulting grid.