Proseminar Numerische Mathematik I, SS 04
1. Programmieraufgabe, abzugeben bis 23. März 2004

Schreiben Sie eine Funktion zur Lösung eines linearen Gleichungssystems

$$Ax = b$$

für eine quadratische Matrix $A \in \mathbb{R}^{n \times n}$. Dabei soll es möglich sein, zwischen

a) der Benutzung der MATLAB Funktion `lu` mit Pivotstrategie,

b) der Verwendung von \ ,

c) einer selbst implementierten LR-Zerlegung ohne Pivotsuche,

d) einer Cholesky-Zerlegung unter Benutzung der MATLAB Funktion `chol` sowie

e) einer von Ihnen implementierten Cholesky-Zerlegung für eine Tridiagonalmatrix

zu wählen. Außerdem soll es möglich sein, das System für mehrere rechte Seiten b zu lösen, ohne die Zerlegung zu wiederholen. Zur Ersparnis von Speicherplatz soll in (c) und (e) die Matrix A dabei mit den Elementen von L und gegebenenfalls R überschrieben werden.

Als Hilfe sei hier als Musterbeispiel ein Funktionskopf angegeben:

```
% LES Linear Equation Solve.
%   [X,time]=LES(A,b,flag) computes the solution
%   to the linear system
%                        A*X=B,
%   where A must be a square non-singular matrix and B a
%   right-hand side with appropriate dimension. The solution
%   is stored in the variable X. The variable time contains
%   the CPU time measured in ms. With the variable flag the
%   user should choose one of the following methods for
%   computing X:
%
%   flag='a': Use MATLAB routine lu with partial pivoting.
%   flag='b': Apply MATLAB operation \.
%   flag='c': Utilize the own LU-factorization without pivoting.
%   flag='d': Use the MATLAB routine chol.
%   flag='e': Apply the own Cholesky solve for tridiagonal
%             matrices.
function [X,time]=les(A,B,flag)
```

Sei $v \in L^2([-1,1])$ eine verrauschte Funktion. Sei u eine entrauschte Abschätzung von v mit $u \in L^2([-1,1])$ und $u' \in L^2([-1,1])$, die durch die Minimierung von

$$J(u) = \int_{-1}^{+1} [u(x) - v(x)]^2 dx + \mu \int_{-1}^{+1} [u'(x)]^2 dx$$

bestimmt wird. Mit einem äquidistanten Gitter $x_i = -1 + 2i/n$, $i = 0, \ldots, n$, $h = 2/n$, sei J_h eine Annäherung von J:

$$J_h(\boldsymbol{u}) = h \sum_{i=1}^{n} (u_i - v_i)^2 + h\mu \sum_{i=1}^{n-1} [(u_{i+1} - u_i)/h]^2$$

wobei $\boldsymbol{u} = \{u_i\}$, $u_i \approx u((x_{i-1} + x_i)/2)$, $\boldsymbol{v} = \{v_i\}$, und $v_i \approx v((x_{i-1} + x_i)/2)$. Die Optimalitätsgleichung $\nabla_{\boldsymbol{u}} J_h = 0$ wird durch die Matrixgleichung $A\boldsymbol{u} = \boldsymbol{v}$ charakterisiert, wobei:

$$A = I + \frac{\mu}{h^2} D, \qquad D = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix}$$

so definiert wird. Weitere Details befinden sich auf:

`http://www.kfunigraz.ac.at/imawww/keeling/an3-p19.ps`.

Lösen Sie nach $\boldsymbol{u}$ für folgendes $\boldsymbol{v}$ auf:

```
n  = 101;                                          % input dimension
pn = 0.1;                                          % choose the noise level
h  = 2.0/n;                                        % width of grid cells
x  = -1.0 + 0.5*h + h*(0:(n-1));                   % define cell centers
v0 = [zeros(1,fix(n/2)),ones(1,n-fix(n/2))]';      % define a step function
v  = v0 + pn*randn(n,1);                           % add noise
D  = 2*diag(ones(1,n)) - diag(ones(1,n-1),-1) - diag(ones(1,n-1),1);
D(1,1)=1;                                          % define the matrix D
D(n,n)=1;
mu = 0.1;                                          % choose smoothing level
A  = eye(n) + mu*D/h^2;                            % define the matrix A
u  = LES(A,v,flag);                                % solve system with flag
plot(x,v0,'b',x,v,'r',x,u,'g');                    % plot results
```

und zeigen Sie den Effekt von μ. Vergleichen Sie die CPU-Zeiten (MATLAB-Funktion `cputime`) für dieses Beispiel bei beiden Cholesky-Zerlegungen und unterschiedlichen Werten von n.

Die Programme sollen *modular* geschrieben und auf Englisch kommentiert werden. Zum Beispiel sehen Sie:

`http://www.kfunigraz.ac.at/imawww/keeling/prg/tipps.txt`
`http://www.kfunigraz.ac.at/imawww/keeling/prg/beispiel/`