Karlsruhe Institute of Technology
Institute for Theoretical Condensed Matter physics (TKM)
Institute of Nanotechnology (INT)
Prof. Dr. Jörg Schmalian, Dr. Peter Schmitteckert, Dr. Andreas Poenicke, Benedikt Schönauer
WS 2014/2015
27.10.2014
http://www.tkm.kit.edu/lehre/ws2014_1827.php

Exercise Sheet No. 0
"Computational Condensed Matter Theory"

1 Matrices

a) Initialize a 10×10 tridiagonal matrix D, with the value 2 on the diagonal, and 1 on super- and sub-diagonals. Calculate the trace, determinant, eigenvalues and eigenvectors of the matrix D.

b) Construct a 10×10 band matrix H, with the value 2 on the diagonal, and $H_{k,k+2} = i$, $H_{k,k-2} = -i$. Calculate the transpose and complex transpose matrix of H and show that H is hermitian.

c) Calculate the inverse of D and use the result to solve the equation

$$\hat{D}\vec{x} = \vec{y}, \qquad \vec{y} = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)^T$$

and calculate $\vec{x}$. Which alternative method does Matlab propose to calculate $\vec{x}$?

Useful functions: `colon(:)`, `det`, `trace`, `diag`, `ones`, `eig`, `eye`, `toeplitz`.

2 Sparse matrices

We will often encounter huge matrices where only a few entries differ from zero. Matlab offers special methods and functions to handle these sparse matrices efficiently:

a) Initialize a 10000×10000 tridiagonal matrix C with random real values for the diagonal and off-diagonal entries. Visualize the structure of the matrix to ensure it has the correct structure.

b) Calculate the 5 largest and smallest eigenvalues of C as well as the corresponding eigenvectors.

Useful functions: `rand`, `spdiags`, `eigs`.

3 Visualization

a) Visualize the time development of an underdamped harmonic oszillator. The amplitute is given by $x(t) = e^{-\zeta t}\cos(\sqrt{1-\zeta^2}t)$. Draw several curves (for different $0 < \zeta < 1$) in one graph.

b) Plot the function $f(x,t) = \Re\left[e^{i(x-\omega_0 t)}\right], \omega_0 = 2\pi$ at $t = 0$ using 100 sampling points $0 \leq x_i \leq 2\pi$. Use a `for`-loop to increase t in steps of $\Delta t = 0.01$ from $t = 0$ to $t = 15$ and always draw $f(x,t)$. Use the command `drawnow` to enforce an immediate redrawing of the graph. (This is a simple way to generate basic animations)

Useful functions: `colon(:)`, linspace, `plot`, `hold on`, `for...end`, `drawnow`.

Please turn ...

$\boxed{4}$ Visualization of fields

a) There are many possibilities to visualize two-dimensional scalar fields. Use Matlab to visualize the functions

$$\Phi_1(x,y) = xy, \quad \Phi_2(x,y) = \frac{1}{1+r^6}, \quad \Phi_3(x,y) = e^{-r^2} \quad \text{with} \quad r = \sqrt{x^2+y^2}$$

in at least two different ways. Play with different color schemes.

b) In physics also vector fields are common. Have look on the the different ways to visualize 2d vector fields in Matlab and draw the functions

$$\mathbf{E}_1(x,y) = (-y,x), \quad \mathbf{E}_2(x,y) = -\nabla r, \quad \mathbf{E}_3(x,y) = -(y,x).$$

Useful functions: `meshgrid`, `mesh`, `surf`, `pcolor`, `colormap`, `contour`, `quiver`.

$\boxed{5}$ Functions and scripts

a) The potential of a point charge q at $\mathbf{x_0}$ is given by $\Phi(\mathbf{x} = \frac{q}{|\mathbf{x}-\mathbf{x_0}|}$. Write a function `potential` with the x- and y-coordinates of an grid, as well as the position (x_0, y_0) and charge q of the point charge as arguments. The function should return a (discretized) field of the potential.
The function should be called like e.g.

```
[x,y] = meshgrid(-5:0.2:5,-5:0.2:5);
resultat = potential(x,y,-1.5,2.5,1);
surf(x,y,resultat);
```

to draw the potential of an point charge with $q = 1$ located at the position $(-1.5, 2.5)$.

b) Write a function `potential2` which calculates the potential of a system, now with several point charges. Beside the coordinates of the grid, the function should take a list of coordinates and charges as parameter. This list has the structure of a matrix:

$$\begin{pmatrix} x_1 & y_1 & q_1 \\ x_2 & y_2 & q_2 \\ x_3 & y_3 & q_3 \\ & \vdots & \end{pmatrix}$$

meaning a charge q_1 is at (x_1, y_1), a charge q_2 at (x_2, y_2), etc.

Thus the function might be called like e.g.

```
[x,y] = meshgrid(-5:0.2:5,-5:0.2:5);
resultat = potential2(x,y,[2.5 2.5 1; -2.5 2.5 -1; -2.5 -2.5 1; 2.5 -2.5 -1]);
surf(x,y,resultat);
```

The new function should use the function `potential` of exercise 5 b).