Computational Physics - Klausur

16.2.2016

Instructions: Please answer to **six** questions of your choice. Only students that choose the special question number 7 can have the maximum grade 1.0.

Question 1: Generation of Random Numbers

Random number generators as implemented in most libraries typically generate (pseudo-) random numbers x following a uniform probability distribution $u(x) = 1$ in the interval [0,1]. However, often one needs random numbers that follow some non-uniform probability distribution $f(x)$. We have discussed two distinct methods for generating such random numbers.

(a) Which method is best suited for generating random numbers x following the Lorentzian distribution $f(x) = \frac{1}{\pi}\frac{1}{1+x^2}$?

Explain the method and explain why you would choose it over the other in this case. Sketch an algorithm implementing the method.

(b) How can you generate random numbers following a Gaussian distribution

$f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$?

Could you use the same method as in the case of the Lorentzian distribution? Why or why not? Explain the method and sketch the algorithm for implementing the method.

Question 2: Interpolation and Differentiation

(a) Explain the basic idea behind the Lagrange interpolation. What is the problem for interpolation by Lagrange polynomials higher than 1st order?

(b) Numerical Differentiation: How can a derivative be approximated numerically? State the expressions for forward, backward and central differences for derivatives of 1st order. What is the advantage of the central difference method compared to the other two?

Question 3: Monte-Carlo Integration

(a) How is the integral of some function $f(x)$in the interval [a,b] approximated in the simple Monte-Carlo integration scheme? Sketch the algorithm for evaluating $\int_a^b f(x)dx$ by simple Monte-Carlo integration.

(b) How does the error of MC integration scale with the number of evaluation points n? How efficient is Monte-Carlo integration in comparison with other numerical integration methods?

(c) For a d-dimensional integral $\int_{a_1}^{b_1} \dots \int_{a_d}^{b_d} f(x_1, \dots x_d) dx_1 \dots dx_d$, how does the error of MC integration scale with the number of evaluation points? From which dimension d on, MC integration becomes more efficient than other numerical integration methods?

Question 4: Ordinary Differential Equations

(a) Consider then driven harmonic oscillator described by the equation $m\frac{d^2x}{dt^2}=-k\,x-c\frac{dx}{dt}+A(t)$ with the initial condition $x(t=0)=0$. Explain how to convert this equation into a system of first order differential equations.

(b) Which method would you chose to solve numerically this differential equation, and why?

(c) Sketch a program to solve this problem for $A(t)=m\,\omega_0\theta(t-2)$, where θ is the Heaviside step function.

Question 5: Integration

(a) Derive the trapezoidal and Simpson rules for integrating a function in the interval $[x, x+\Delta x]$.

(b) Show how to extend these rules in order to integrate a function in the interval $[a, b]$, when this interval is divided into segments of size Δx . What can we do if the functional has a (integrable) discontinuity in this interval?

(c) Sketch a program that integrates the function $x\cos(x^2)+x^3$ in the interval $[0,1]$ using the trapezoidal method. How do you expect the error to change when we decrease Δx ?

Question 6: Root finding

(a) Explain and compare the bisection, Newton-Raphson, and secant methods to find the zeros of a one-dimensional function in a given interval.

(b) The polynomial x^5-2x+3 has one real root located in the interval $[-5,0]$. Sketch a program to find this root using one of the methods from the previous question.

Question 7: Special question

(a) Explain two different ways that one can use to transform the one-dimensional Schrödinger equation $\left[-\frac{1}{2}\frac{d^2}{dx^2}+v(x)\right]\psi_i(x)=\epsilon_i\,\psi_i(x)$ into a matrix eigenvalue problem.

(b) Show that in one of the cases the resulting matrix is sparse. Give different examples of sparse matrices and how this information can be used to improve the numerical efficiency of the methods.

(c) Explain the Gauss-Jordan elimination method to solve linear systems of equations. Comment on the importance of pivoting to stabilize the algorithm.