Syllabus for Physics Course
Jan.-Apr. Semester

Computational Physics

This is a very hands-on course which will involve a lot of programming assignments. The main aims of the course are two fold:

1. Learning basic methods, tools and techniques of computational physics.
2. Developing practical computational problem solving skills.

Textbooks:

1. Mark Newman, *Computational Physics*, CreateSpace Independent Publishing Platform (2013).
2. Forman Acton, *Real computing made real: Preventing Errors in Scientific and Engineering Calculations*, Dover Publications.
3. Lloyd N. Trefethen and David Bau, *Numerical Linear Algebra*, SIAM.
4. William H. Press, Saul A. Teukolsky, William T. Vetterling and Brian P. Flannery, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*

1 Introduction to computational physics, computer architecture overview, tools of computational physics (3 hours)

What is computational physics? Why do we need it?; *Computer hardware*: basic computer architecture, hierarchical memory, cache, latency and bandwidth; Moore's law, power bottleneck; *Software*: compiled (Fortran, C) vs. interpreted languages (MATLAB, python); software management.; *Parallelization*: MPI; OpenMP; CUDA.

2 Machine representation, precision and errors (1.5 hours)

Representation on a computer: Integer representation; floating-point representation; *Machine precision*; *Errors*: round-off; approximation errors; random errors; errors of the third kind; *Quadratic equations*; *Power series*; *Delicate numerical expressions*; *Dangerous subtractions*; *Preserving small numbers*; *Partial Fractions*; *Cubic equations*; *Sketching functions*;

3 Quadrature and Derivatives (6 hours)

Direct fit polynomials; *Quadrature methods on equal subintervals*; *Newton-Cotes formula*; *Romberg Extrapolation*; *Gaussian quadrature*; *Adaptive step size*; *Special cases*;

4 Solutions of linear and non-linear equations (9 hours)

Simultaneous linear equations: Gauss elimination (pivoting, scaling); LU factorization; Calculating inverse; Tri-diagonal systems; *Eigenvalues and Eigenvectors*: QR Factorization; Gram-Schmidt Orthogonalization; *Real roots of single variable function*; Relaxation method; qualitative behavior of the function; *Closed domain methods (bracketing)*: Bisection; False position method; *Open domain methods*: Newton-Raphson, Secant method; *Complications*; *Roots of polynomials*; *Roots of non-linear equations*;

5 Fourier methods (3 hours)

Fast Fourier transform; Convolution; Correlation; Power spectrum;

6 Random numbers and Monte-Carlo (6 hours)

Random number generators; *Monte-Carlo integration*; *Non-uniform distribution*; *Random Walk*; *Metropolis algorithm*;

7 Ordinary differential equations (9 hours)

Initial value problems: First order Euler method; Second order single point methods; Runge-Kutta methods; Multipoint methods; *Boundary value problems*: Shooting method; equilibrium boundary value method;

8 Partial differential equations (6 hours)

Types of partial differential equations: Hyperbolic, parabolic and elliptic; *Boundary value problems*: Overrelaxation; Gauss-Siedel Method; *Initial value problems*: FTCS method; Numerical stability; Implicit and Crank-Nicolson methods; Spectral methods;