

Computational Physics - Phys 440/740

Course Information, Fall 2008

Aims of the Course

- To expose you to some new and active areas of physics research, which have only become accessible relatively recently thanks to the speed of modern computers. Among the topics to be discussed will be nonlinear dynamics or “chaos” theory, Monte Carlo simulation of statistical systems and quantum field theories, and neural networks. We will also cover computer applications to traditional areas, e.g. electromagnetism and quantum mechanics.
- To help you gain confidence, experience, and sophistication in applying computers to help solve problems of interest to physicists.
- To introduce a few common numerical methods, which are of particular use in engineering and the physical sciences. These include methods for solving ordinary differential equations (including initial value, and boundary/eigenvalue problems), certain partial differential equations, and the Monte Carlo method. Note, however, that this course is not *primarily* a “Numerical Methods” course. The wide availability of numerical software packages (NAG, IMSL, SLATEC, Cernlib, to name a few) as well as the reference *Numerical Recipes* by Press et al., have made the life of the working physicist much easier, and also made it less necessary for us to become great experts in numerical analysis.

There will be an introduction to some of the basic concepts of statistical mechanics. The lecture course will also cover elements of computer science from the physics point of view, including: transistors, digital logic circuits, Turing machines, and quantum computation. If time permits, I will in addition discuss some statistical aspects of data analysis.

Course Requirements

First of all, you must know how to program a computer, in either Fortran 77, C, or C++. This is essential, so you must have either completed CSC 209 or 210 (209 is preferred), or else have my approval of an equivalent background. You must also have completed a semester of Modern Physics (Phys 320), and either one semester of quantum mechanics (Phys 430), or *two* semesters of electromagnetism (Phys 360 and 460).

Text

Computational Physics by Giordano and Nakanishi

Lab Sections and Grading

You will be assigned a set of projects to complete in the course of the semester, and these are to be done both in and, if necessary, out of the lab section. All grading is based on your project reports. There are no exams in this course, either in-class or take home.

Each project sets you a series of computational tasks. The way to get an A in the course is to complete these tasks on time, and comment knowledgeably on the results.

A 3-hour lab section is held once a week in TH123. This will give you a chance to get partly (or completely) through each project, with me hanging around to give advice.

440/740 Differences

Students who take this course for graduate credit (Phys 740) will be expected to tackle, in addition to the “core” 440 projects, a number of additional projects which are more advanced conceptually, and more demanding to implement, than the basic set of projects required for undergraduate credit. The 740 projects are marked with an asterisk in the Project Schedule below.

Languages and Packages

You are free to use C, C++, or Fortran 77. If you are familiar with the Matlab or IDL packages, it is also OK to use those. We will be using the Unix workstations, but you don't have to be proficient in Unix to take this course (I'm no expert myself).

You will be given an introduction to both Latex and Gnuplot, which are installed on our Unix system, and which are freely available for Macs and PC's. Latex is specifically designed for mathematical text-processing, and has become the standard in both physics and mathematics. All project reports are to be written in Latex. Microsoft Word, or the like, is not to be used. Gnuplot is a very easy-to-use plotting package which produces high-quality output, but I do not insist on its use. If you are comfortable with Matlab or IDL, you are free to use their built-in plotting capabilities.

The course will also include an introduction to Matlab (and its freeware clone, Octave). Matlab is a wonderful software package with two disadvantages: First, it is an interpreted language (like Mathematica, Maple, and Basic), which means it is too slow for heavy-duty numerical computation. Nevertheless, for many applications, it is a great tool. Secondly, it is very expensive (\approx \$3000 if you want to buy it for your own PC). Fortunately, you can download Octave for free, and Octave has almost the same capabilities as Matlab.

Project Reports

Project reports are much like lab reports. I will not prescribe a rigid format, but your report should include

- The problem you are trying to solve, or the system you are investigating;
- The equations to be solved numerically;

- The method used to solve those equations (state, if relevant, your choice of parameters);
- Your numerical results, usually displayed graphically;
- Informed commentary on your results.

Do not be too verbose. If it was discussed in class, you may assume that I am familiar with the numerical method you are using, so there is no need to rederive it or explain it in great detail. However, it is usually a good idea to mention the parameters you chose, or some particular way in which you implemented the algorithm.

Commentary is important. After each graph or set of graphs, try to indicate (at least briefly) what your results tell you, what can be learned from them. Do the results agree with some analytic formula, or could they only be produced numerically? Are your results meaningless? If so, do you think that this indicates a program error, or else some inadequacy of the numerical method?

You are encouraged to explore further on your own, if you feel so inclined, and report on the results. Try other parameter ranges, related equations, and so on.

Disability Statement

Students with disabilities who need reasonable accommodations are encouraged to contact the instructor. The Disability Programs and Resource Center (DPRC) is available to facilitate the reasonable accommodations process. The DPRC is located in the Student Service Building and can be reached by telephone (voice/TTY 415-338-2472) or by email (dprc@sfsu.edu).

Project Schedule

Project	Area	Problem or System	Method
0	Introduction		latex, gnuplot
1	Chaos	logistic map	
2	Chaos*	pendulum	Runga-Kutta
3	QM	energy eigenvalues	shooting and matching numerical integration
4	Potentials	Laplace/Poisson eq (elliptic PDEs)	Over-relaxation
5	Waves	hyperbolic PDEs	Discretization
6	QM*	time evolution (parabolic PDEs)	Crank-Nicholson Method
7	Stat Mech	D=1 Ising	Monte Carlo
8	Stat Mech*	random matrices	Matlab
9a	Stat Mech*...	D=2 Ising	Monte Carlo
9b	or - Neural Networks*		

440 Requirements

Projects 0,1,3,4,7

740 Requirements

Projects 0,1,2,3,4,7,
either 5 or 6 or 8,
either 9a or 9b

(* denotes PHYS 740 projects)