Math 498, Winter 2008

Some Suggested Project Topics

1. Find some physical phenomenon that you are interested in or that you are studying in another class that is modeled using Monte Carlo methods or ordinary differential equations. Implement the Monte Carlo simulation or solve the system of ODE's using one or more of the methods that we discuss in class, or related methods that you find in the literature. Report on the physical phenomenon being modeled, on how you solved the problem numerically, on how accurate your solution is, and on how efficient your algorithm is. In short, say why you chose the method that you chose. Turn in a written report and give a short talk about your work.

   One excellent source of problems is the Mathematical Contest in Modeling. See

   `http://www.comap.com/undergraduate/contests/mcm/`

   Some of the contest problems may interest you. You can look at other people's solutions to these problems as well, but I suggest that you come up with your own, although you can use other papers as references *provided they are cited.*

2. A list of the Top Ten Algorithms of the 20th Century was compiled by Jack Dongarra and Francis Sullivan in the Jan./Feb. 2000 issue of *Computing Science in Engineering.* Choose one of these algorithms, report on what it is and where it is used, and, if appropriate, implement the algorithm or find an implementation somewhere and use it to solve some problem of interest. Your report should go beyond what is done in class in describing uses and implementations of the algorithm.

3. *The SIAM 100-Dollar, 100-Digit Challenge.* In the Jan./Feb. 2002 issue of *SIAM News*, Nick Trefethen published a list of ten numerical problems, each of whose answers was a real number. He challenged readers to solve the problems to ten correct digits each, using whatever methods they chose. Since then, many of the solutions have been computed to far more than the original ten decimal places requested, and many solvers have posted their solutions on the web. See, for example,

   `http://web.comlab.ox.ac.uk/oucl/work/nick.trefethen/hundred.html`

   In fact, a book has been written about the solutions: *The SIAM 100-Digit Challenge*, by Bornemann, Laurie, Wagon, and Waldvogel.

   Choose one or two of these problems and either attempt to solve them yourself or look at what others have done and report on their solutions. Or you could do both: First devise your own solution method and see how many decimal places of accuracy you can obtain; then look at what others have done to see if they had any better ideas.

4. SVD and text matching: The SVD can be used for text matching by search engines. Google does not use this for text matching. A project could develop how the SVD is used in this context and give examples, as well as discuss the strengths and weaknesses of this method. The book *Understanding Search Engines: Mathematical Modeling and Text Retrieval* by Michael W. Berry and Murray Browne is a strong introduction to these topics.

5. Monte Carlo methods and queuing theory: Monte Carlo methods can be used to model queues. There is a little bit about this in the course notes, but much more information can be found in other sources. These models can be used to answer questions like the following: If there are multiple servers, should all customers wait in one line for the first available server (as is usually done in airports), or should customers choose a line and wait for that particular server (as is usually done in supermarkets)? Of course, to answer these questions one must decide on a measure of efficiency. (Minimize the average customer waiting time or the maximum customer waiting time, etc.) These same questions are relevant to computing questions such as how to assign jobs to processors when there are multiple users competing for service. Choose an appropriate situation to model (One that occurred to us after a few lunches at the HUB: Can the waiting time in the Subway line at the HUB be shortened through a better distribution of labor?) and run a Monte Carlo simulation to solve your problem. Report on your results and conclusions. Also give an indication of how accurate you believe your results are.

A typical project length is about 5-10 pages of text, along with supporting figures, data, and perhaps computer programs. Computer programs, if included, should be relegated to an appendix so as not to disturb the flow of the text. You should also be selective about what output results are included in the text. It is best to present just a small amount of output and explain it well. Graphs and other visual presentations are usually much better than long lists of numbers. Figuring out how best to present data is an art in itself.

Your paper should contain an abstract – a short overview of the problem addressed, the methods used, and the results obtained – usually about 100 words or so. Your paper should contain a clear problem description with background information about why this problem is important or interesting. It should describe the numerical methods used to solve the problem, and the results obtained. If you wish, you might discuss what further improvements might be made if you were to continue working on this problem. You **must** include a bibliography listing all references that you used, e.g., books, journal articles, web pages, etc. Put a citation in the paper at the point where you use material from another source, and if you take a sentence or paragraph directly from another source, it should be in quotes, as well as referenced. The format of your paper depends very much on the type of problem that you work on, but these sections should be included in all papers.

You may work alone or with one or two partners. You are strongly encouraged (but not required) to present your work to the class. We have set aside the time scheduled for our final for project presentations. You may also present your work at an earlier date in order to get feedback before turning in the final paper.