ME690 **Spring 2005**

Advanced Computational Numerical Analysis of Partial Differential Equations

The purpose of this course is to provide to Engineering/Physical Science graduate students a background in numerical methods that will prepare them to at least begin computational work on essentially arbitrarily difficult problems in partial differential equations (PDEs). The course begins with classical, but fairly advanced, material on the solution of both steady-state and time-dependent PDEs with a focus on single linear equations posed on rectangular domains. Following this preparation, *nonlinear* problems, and *systems* of PDEs are treated. Finally, the restriction of rectangular domains is removed via an introduction to *grid generation*.

Prerequisites for the course are EGR/MA 537 and expertise in programming with a high-level language—preferably Fortran.

Course Outline

I. Solution of Elliptic PDEs

- A. Basic theory of linear fixed-point iteration
- B. Successive overrelaxation (SOR)
- C. Alternating direction implicit (ADI) methods
- D. Incomplete LU decompositions (ILU)
- E. Preconditioning
- F. Introduction to conjugate gradient methods
- G. Multigrid methods
- H. Domain decomposition

II. Time-Splitting Methods for Evolution Equations

- A. ADI methods—again
- B. Locally one-dimensional (LOD) techniques
- C. General Douglas & Gunn time splitting

III. Some Miscellaneous Topics

- A. Discretizations for general self-adjoint form operators
- B. Treatment of mixed derivatives
- C. The "cell-Re" problem
- D. Nonlinear equations
- E. Systems of PDEs

IV. Introduction to Grid Generation

- A. Basics of (applicable) differential geometry
- B. Grid generation via PDEs
- C. Algebraic grid generation

There is no required textbook, but course lecture notes will be made available as a downloadable PDF file from the course website http://courses.engr.uky.edu/ME/me690.

Course grades will be determined on the basis of performance on two extensive homework assignments (each requiring implementation and testing of several different algorithms and then comparing the results), and on a term project that can be viewed as a take-home final exam.