Course Outline

3. Course Description

Computational Science I. This course provides an overview of, and practical experience in, using algorithms for solving numerical problems arising in applied sciences. In particular, the two dominant themes in the course are: (1) the theory and development of numerical algorithms, and (2) programming applied to scientific computation. Mathematical analysis and the scientific problem-solving environment provided by Python will be used to investigate the applicability, efficiency and accuracy of a number of basic algorithms that arise in many scientific problems. Topics include basic numerical linear algebra, interpolation, solution of nonlinear equations, quadrature and, if time allows, the solution of initial-value problems for ordinary differential equations and optimization.

4. Learning Outcomes

Upon completing this course students will be able to implement a number of basic computational algorithms in Python. They will be able to evaluate the efficiency and accuracy of these algorithms using theoretical arguments as well as visualization, and will be able to combine them to perform more complex computations. They will have the ability to numerically approximate and visualize solutions to the mathematical problems they may encounter in many of their third and fourth year courses.

6. Outline of Topics in the Course (tentative)

Week	Topic	Python
1	Introduction	
1-2	Solving nonlinear equations	Functions, loops, lists, branching, matplotlib, pyplot
3-4	Solving systems of linear and nonlinear equations	Arrays, scipy.linalg
5-6	Computational complexity	Logarithmic plotting
6-8	Interpolation and least squares	scipy.interpolate
8-10	Integration and differentiation	scipy.integrate
10-12	Additional Topics	