Literature

The lecture notes involve topics from a rather wide range of numerics of PDE’s. However, these topics are treated in a very superficial manner. Mathematical justifications are simply omitted. Here, I try to provide you with references to some more adequate material. Majority of the material of the lecture notes is extracted from these sources. Of course there are plenty of other good sources too.

Numerical integration and ODE’s

Concerning ODE’s, I recommend the books of Hairer and his coauthors, namely [HNW93, HW96]. There is also a slightly more advanced book [HLW10] discussing additionally, e.g., symplecticity and KAM-theory, i.e., qualitatively proper methods to time-integrate Hamiltonian systems.

Finite element method

The classical books defining the methods mathematical framework are from P.G. Ciarlet [Cia78]. Another often mentioned book is by D. Braess [Bra97], it is of hand book type and contains many of the main result. The bible of mixed methods is [BF91]. In order to fully understand the saddle point theory, one should get familiar with convex analysis, duality theory, and calculus of variations. Up to my mind, the best exposition is in [ET76, Roc70]. If one wants to avoid functional calculus, many parts of the theory with nice illustrations (concerning only Euclidian finite dimensional spaces) can be found from [BSS06]. These books standard references in related numerical analysis. However, to actually program something using only the material mentioned here might be exhausting and tough to read.

Convection-Diffusion problem and related methods

A wonderful book of Roos, Stynes, and Tobiska is a must read [RST08]. It includes mathematica analysis and addresses high variety of methods (with many references), including both finite difference and the finite element method.
Nonlinear problems, iterative methods

The book of Kelley [Kel95] introduces (besides of linear solvers) all kind of Newton variants and the related convergence analysis. More advanced analysis (related to optimization using duality theory) on a functional level can be found in [IK08].

Adaptive methods and error estimation

There are plenty of good sources, but I (without claiming to be objective) recommend the series of books [NR04, Rep08, MRN14]. The difficulty of the books (as well as the mathematical generality) decreases in time, i.e., the first one is the most advanced and complicated and the last one hopes to be easy to grasp presentation of functional type a posteriori error estimates (in linear elliptic problems). On residual type error estimates the standard references are [Ver96, AO00].

References


Numerical Integration is simply the approximation of integrals and is useful for integrals that cannot be evaluated by the special formulas. One method under it is Romberg Integration. From the methods that was taught in class, it's been observed that this is the only method that eliminates errors (though not all errors are eliminated) through the usage of Richardson Extrapolation as seen in the derivation. Though Composite Simpson's 1/3 Rule outshone Composite Trapezoidal Rule and Romberg, Romberg still holds the trump card for being efficient and also employs the robustness of the Composite Tr Navier-Stokes differential equations used to simulate airflow around an obstruction. In analysis, numerical integration comprises a broad family of algorithms for calculating the numerical value of a definite integral, and by extension, the term is also sometimes used to describe the numerical solution of differential equations. This article focuses on calculation of definite integrals. ODEs of this form are very common in physics, and usually instead of \( y \) we have the velocity, \( v \). In general a system of N ODEs takes the form. where there are N independent variables, and N specified functions \( f_1 \) through \( f_N \). This system of equations requires a set of N initial conditions, which are most commonly of the form \( x_i(0) = x_i(0) \). Solutions to ODEs in scipy are provided by the same scipy.integrate module that provides numerical integration methods; indeed, the techniques for solving ODEs and computing numerical integrals are closely related. The integrate module provides an interface to a powerful and general ODE solver. As with integration, the first step is to define the function specifying the equations to be integrated. Numerical integration of ODE with singular right-hand-side. This problem arised as a self-similar model of a certain gas-dynamic problem. We've made a numerical gas-dynamical solver in a time-space domain and it turned out that a shock wave appears in the solution exactly at the same place the caustic appears (here \( t \) is a self-similar variable). This is a kind of a happy end for us. However, the pure algorithmic issue still exists in the self-similar formulation of the problem. I don't know how to solve an ODE with a finite singularity at the right-hand-side. Thanks to everyone who cared.