Numerical Methods for Partial Differential Equations - Notes - v0.2.0-dev

260236

November 2024

Preface

Every theory section in these notes has been taken from the sources:

- Course slides. [1]

About:

GitHub repository

These notes are an unofficial resource and shouldn't replace the course material or any other book on numerical methods for partial differential equations. It is not made for commercial purposes. I've made the following notes to help me improve my knowledge and maybe it can be helpful for everyone.

As I have highlighted, a student should choose the teacher's material or a book on the topic. These notes can only be a helpful material.

Contents

1 Basic Concepts

In this course, we introduce numerical methods for the solution of **Partial Differential Equations** (PDEs), with focus on the **Finite Element** (FE) **method**[1] and the use of the computer for the construction of the PDEs numerical solution.

We will consider the numerical approximation of elliptic and parabolic PDEs by considering their variational formulation, Galërkin and FE approximations in 1D/2D/3D, the theoretical properties and practical use of the methods, algorithmic aspects, and interpretation of the numerical results.

Advanced topics include the approximation of saddle-point PDEs (Stokes equations), vectorial, nonlinear, and multiphysics differential problems, domain decomposition methods exploiting the properties of the PDEs, and the introduction to parallel computing for the FE method, i.e., in the *High Performance Computing* (HPC) framework.

Finally, the course will feature the use of the `deal.II` software library, a C++ open source FE library, and ParaView for the visualization of numerical solution and scientific computing data.

[1]The finite element method (FEM) is a popular method for numerically solving differential equations arising in engineering and mathematical modeling. Typical problem areas of interest include the traditional fields of structural analysis, heat transfer, fluid flow, mass transport, and electromagnetic potential. Computers are usually used to perform the calculations required. With high-speed supercomputers, better solutions can be achieved, and are often required to solve the largest and most complex problems. (source)

1.1 Mathematical Models and Scientific Computing

Definition 1: Mathematical Model

A **Mathematical Model** is a **set of** (algebraic or differential) **equations that is able to represent the features of a complex system or process**.

Why do they exist?

Models are **developed** to:

- Describe
- Forecast
- Control

The **behavior or evolution of such systems**.

We are interested in the physics models. **Physics-based models** are those **mathematical models that are derived from physical principles** (like conservation laws of mass, momentum, energy, etc.) **and that encode natural laws of leading to (differential) equations whose solutions are often represented in the form of functions**. However, the analytical solution of such models is rarely available in closed form, for which numerical approximation methods are instead employed.

Definition 2: Numerical Modelling

Numerical Modelling indicates **sets of numerical methods that determine an approximate solution of the original** (often infinite-dimensional) **mathematical model**, by turing it into a *discrete problem* (algebraic, finite-dimensional), whose dimension (size) is typically very large.

Definition 3: Scientific Computing

Scientific Computing is **a branch** of Mathematics **that numerically solves** (differential) **mathematical models by building approximate solutions though the use of a calculator**.

For numerical models of large size, parallel architectures for calculators and the HPC framework are typically used.

❓ Why did we introduce mathematical models and physical models?

Because they are connected and used together. Mathematical models are conventionally used altogether with theoretical (mathematical) models and experimental tests. Unfortunately, in several cases theoretical models are not available (like in Computational Medicine) or experimental tests are not meaningful or cannot be performed (for example, for nuclear testing). Physics-based models have witnessed an increasing role in the modern society in virtue of the massive developments of Scientific Computing and computational tools.

Since a large amount of data is becoming available from multiple sources nowadays, data-driven models are fundamentals. **Data-driven models** are those mathematical models built from meaningful data that do not rely on physical principles, because the latter are not available or are not reliable, and whose construction calls for statical learning methods.

Physics-based mathematical models (**mathematical problems**) are a fundamental pillar in the understanding and prediction of several physical phenomena and processes (**physical problems**). However, these mathematical models lead to problems that can rarely be solved analytically, or in an exact way (**exact solution**), especially for PDEs: with only a few exceptions, it is not possible to write their solution explicitly.

Numerical methods and numerical approximation techniques (**numerical problems**) serve the purpose to determine an **approximate solution** of a mathematical model. When the calculator is used to determine such approximate solution, the latter is called **numerical solution** (see the Figure 1).

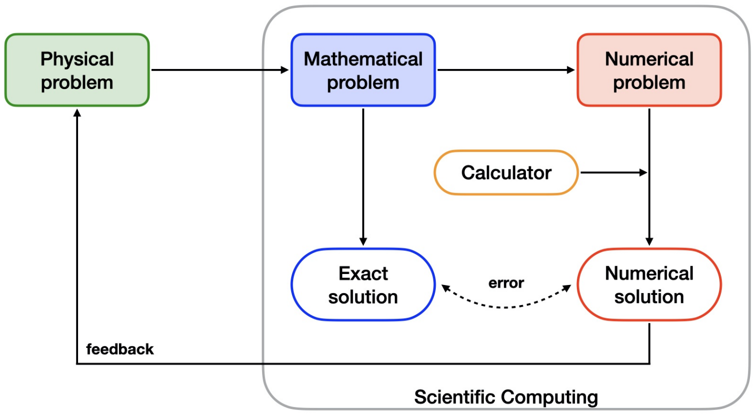

Figure 1: Scientific Computing.

References

[1] Quarteroni Alfio Maria. Numerical methods for partial differential equations. Slides from the HPC-E master's degree course on Politecnico di Milano, 2024.

Index