有限元方法

Finite Element Methods

Chapter 0: Introduction

主讲人: 李琦

liqihao@chd.edu.cn

School of Science, Chang'an University

课程详情

- **课程名称**：偏微分方程有限元素法
- **英文名称**：Finite Element Method for PDE
- **课程编号**：s1203106001
- **课程学时**：60 课时 (第 1-15 周)
- **选课对象**：微分方程数值解方向研究生

课程目的

通过本课程的学习

- 使学生掌握有限元方法的数学理论基础
- 并能够独立地开展有限元方法理论和实践方面的研究
- 为今后的科研工作打下必要的理论基础
- 可以理解并使用现有程序/软件研究问题
- 开发出自己的程序包

教学方式及要求

- 教师授课为主
- 学生详细讲解实现过程
- 学生上机编程，实现数值算法，提交实验报告

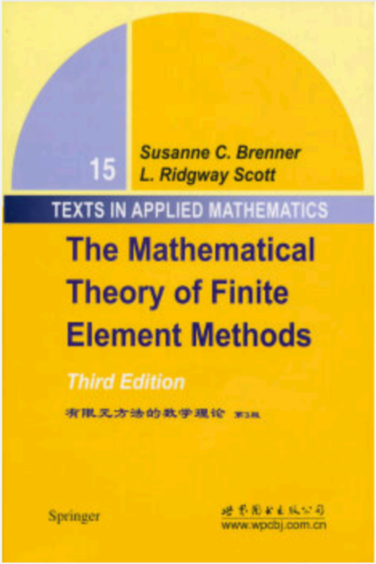
先修课程

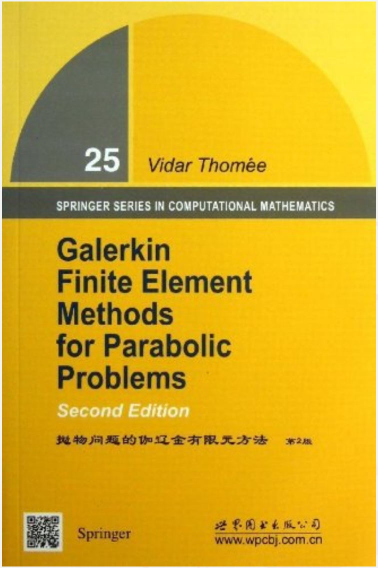
1. 数学分析、高等代数
2. 泛函分析、数学物理方程、数值分析
3. 熟练的编程技能, 例如: Matlab、Python、Julia、$\cdots$

相关教材

1. Brenner, Susanne, and Ridgway Scott. **The Mathematical Theory of Finite Element Methods**. Vol. 15. Springer Science & Business Media, 2007.
2. Thomée, Vidar. **Galerkin Finite Element Methods for Parabolic Problems**. Vol. 25. Springer Science & Business Media, 2007.
3. Boffi, Daniele, Franco Brezzi, and Michel Fortin. **Mixed finite element methods and applications**. Vol. 44. Heidelberg: Springer, 2013.
4. Ciarlet, Philippe G. **The Finite Element Method for Elliptic Problems**. Society for Industrial and Applied Mathematics, 2002.
5. Larson, Mats G., and Fredrik Bengzon. **The Finite Element Method: Theory, Implementation, and Applications**. Vol. 10. Springer Science & Business Media, 2013. （MATLAB Implementation）
6. Chen, Zhangxin. **Finite Element Methods and Their Applications**. Springer Science & Business Media, 2005.
7. 王烈衡，徐学军. **有限元方法的数学基础**. 北京：科学出版社. 2004.
8. 李荣华，刘播. **微分方程数值解法（第四版）**. 北京：高等教育出版社. 2009.

教材图片


15
Susanne C. Brenner
L. Ridgway Scott
TEXTS IN APPLIED MATHEMATICS
The Mathematical Theory of Finite Element Methods
Third Edition
有限元方法的数学理论 第3版
Springer
世界图书出版公司
www.wpcbj.com.cn


25
Vidar Thomée
SPRINGER SERIES IN COMPUTATIONAL MATHEMATICS
Galerkin Finite Element Methods for Parabolic Problems
Second Edition
抛物问题的伽辽金有限元方法 第2版
Springer
世界图书出版公司
www.wpcbj.com.cn

考核方式

- 采用期末考试 + 实验报告 (作业) 的形式综合评定
- 期末考试 (50%)
- 实验报告 (上机作业) (50%)

注

- 作业、实验报告和课程论文需要用 LaTeX 排版书写，
- (1) 以 PDF 或者 tex 源文件打包邮件提交，或者 (2) 以 Overleaf 分享链接的方式提交，
- 以上提交包含所有程序和文档 tex 源文件。

答疑与文件共享

- FEM 微信群

学习资源

- 国家天元数学东北中心 https://space.bilibili.com/393390076
  - 有限元基础编程-何晓明 (Missouri University of Science & Technology)
  - FEALPy 中的偏微分方程数值解程序设计与实现-魏华祎 (湘潭大学)
- deal.II 视频教程 https://www.bilibili.com/video/BV12x411d7Uv
- NGSolve 视频教程 https://www.youtube.com/playlist?list=PL_5FauasEdy01ftt2BH9XCsDjgW_moaBi
- NGSolve 交互教程 https://jschoeberl.github.io/iFEM/intro.html
- FEToy https://www.bilibili.com/video/BV1Eb4y1v7Z5?spm_id_from=333.999.0.0
- FEniCS
- FreeFEM
- Firedrake
- ...
- 欢迎补充

起源-来自维基百科

Development of finite element method can be traced back to the work by **Alexander Hrennikoff** (Russia, 1896 —1984) and **Richard Courant** (German American mathematician, 1888 –1972) in the early 1940s. Another pioneer was Ioannis Argyris.

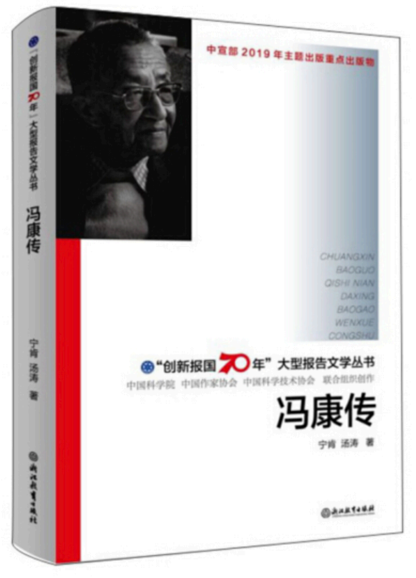
In China, in the later 1950s and early 1960s, based on the computations of dam constructions, **Kang Feng** proposed a systematic numerical method for solving partial differential equations. The method was called the **finite difference method based on variation principle**, which was another independent invention of the finite element method.

Although the approaches used by these pioneers are different, they share one essential characteristic: mesh discretization of a continuous domain into a set of discrete sub-domains, usually called **elements**.

中国计算数学之父

- 英文 Wikipedia https://en.wikipedia.org/wiki/Feng_Kang
- 中文 Wikipedia https://zh.wikipedia.org/wiki/%E5%86%AF%E5%BA%B7
- 袁亚湘维护的主页 http://lsec.cc.ac.cn/~yyx/fk.html
- 百度百科

中国计算数学之父



- 有限元法的创始
  20 世纪 50 年代末，冯康在解决大型水坝计算问题的集体研究实践的基础上，独立于西方创造了一整套解微分方程问题的系统化、现代化的计算方法，当时命名为**基于变分原理的差分方法**，即现时国际通称的有限元方法。
- 哈密顿方程和哈密顿算法（即辛几何算法或辛几何格式）

中国计算数学之父

冯康先生的著名观点:

- “在理论上等价，在计算上未必等效”
- “让离散系统尽可能多地保持原连续系统的内在性质”

已经成为科学与工程计算领域的指南针。

Some important numerical methods

- **Finite Difference Method (FDM)**

  FDM is very easy to understand and apply. They may not be so convenient to handle general geometries of complex domains and general boundary conditions (BCs). Convergence often has to use the **classical functional spaces** $C^k(\Omega)$.

- **Finite Volume Methods (FVM)**

  FVM is based on local properties of the PDEs, and constructed by means of **two sets of triangulations of the physical domain Ω: primal and dual triangulations**. This brings some inconvenience for the mesh generations. In general, FVMs can preserve important physical conservations involved.

- **Spectral Methods**

  Spectral methods are very easy to understand and apply, and may converge with very higher order accuracy when the solutions are smooth. But usually they can deal with PDEs only on **regular domains with simple BCs**, such as rectangular domains or spherical domains. They result in algebraic systems which are **very dense**, with much larger condition numbers than the ones by FEMs and FDMs, so the discrete systems are often highly ill-conditioned.

Finite Element Methods

Finite element methods (FEMs) can easily handle different boundary conditions and general geometries of complex domains. A very general theory has been developed for **convergence and stability analysis of FEMs**, which are applicable to the convergence and stability analysis of many other popular numerical methods, such as finite difference methods (FDMs), finite volume methods (FVMs) and spectral methods. The resulting discrete systems of linear and nonlinear algebraic equations are **very sparse**, and can be solved by effective preconditioning iterative methods, for instance, domain decomposition methods and multigrid methods.

In general, FEM should not be viewed as a normal numerical method. More accurately speaking, it should be viewed as a **numerical methodology**. It is so fundamental that many other popular numerical methods, such as FDMs, FVMs and Spectral Methods, often have to borrow the convergence analysis tools of FEMs in order to achieve some elegant convergence in certain Sobolev spaces.

Procedures for applying FEMs

Mathematical models (PDEs)

$\Longleftrightarrow$

variational problems (also called weak formulation, integral equations)

$\Longrightarrow$

approximation by FEMs

$\Longleftrightarrow$

systems of linear or nonlinear algebraic equations $Au = F$ or $A(u)u = F$, which are then solvable either by direct or indirect methods on computers.

One simple example

Consider the following one-dimensional two-point boundary value problem:

$$
\begin{aligned}
& -(a(x)u_x)_x + c(x)u = f(x), \quad x \in (0,1) \\
& u(0) = \alpha, \quad u(1) = \beta.
\end{aligned}
$$

Variational principle. For any $v \in C^1(0,1)$ such that $v(0) = 0$ and $v(1) = 0$, using the integration by parts formula we obtain

$$
\int_0^1 (a(x)u_x v_x + c(x)uv)\, dx = \int_0^1 f(x)v\, dx, \quad \forall v \in H_0^1(0,1)
$$

where the space $H_0^1(0,1)$ is defined by

$$
\begin{aligned}
H^1(0,1) &= \{v \in L^2(0,1);\ v_x \in L^2(0,1)\}, \\
H_0^1(0,1) &= \{v \in H^1(0,1);\ v(0) = v(1) = 0\}.
\end{aligned}
$$

Galerkin approximation. Construct two finite dimensional spaces V^h and V_0^h respectively to approximate $H^1(0,1)$ and $H_0^1(0,1)$:

$$V^h = \{\phi_0, \phi_1, \cdots, \phi_N, \phi_{N+1}\} \subset H^1(0,1), \quad V_0^h = \{\phi_1, \cdots, \phi_N\} \subset H_0^1(0,1)$$

Then we can formulate:

Galerkin method. Find $u_h \in V^h$ such that $u_h(0) = \alpha$, $u_h(1) = \beta$ and

$$\int_0^1 (a(x)u_h'(x)v_h'(x) + c(x)u_h(x)v_h(x))\, dx = \int_0^1 f(x)v_h(x)\, dx, \quad \forall v_h \in V_0^h.$$

Linear system. Let $u_h = \sum_{j=1}^N u_j\phi_j + \alpha\phi_0 + \beta\phi_{N+1}$ and $v_h = \phi_i$, $(i = 1, 2, \cdots, N)$, then we derive

$$Au = F.$$

Such system of algebraic equations can be solved by either direct methods: e.g., Gauss elimination, Cholesky factorization; or indirect methods (also called iterative methods): e.g., Jacobi, Gauss-Seidal, SOR methods, CG or PCG methods; Secant method, Newton' s method or quasi-Newton' s method (when the discrete system is nonlinear).

Matrix A. A is usually a full matrix, so the storage of A can be very large, and solving $Au = F$ can be very expensive and unstable (since A is usually ill-conditioned).

FEMs. V^h and V_0^h are chosen to be piecewise polynomials with local supports, based on a partition of the domain $[0, 1]$:

$$\mathcal{T}^h : 0 = x_0 < x_1 < \cdots < x_{N+1} = 1,$$

for instance, we may take $\phi_i(x)$ such that

$$\phi_i(x_j) = \delta_{ij}, \quad \forall i, j.$$

Then we can see the following nice situation: A is extremely sparse, so it needs very small storage, and many preconditioned efficient iterative solvers exist for solving the resultant discrete systems.