# INTRODUCTION to
# FINITE ELEMENT METHODS

CARLOS A. FELIPPA

Department of Aerospace Engineering Sciences
and Center for Aerospace Structures
University of Colorado
Boulder, Colorado 80309-0429, USA

Last updated Fall 2004

# Preface

This textbook presents an Introduction to the computer-based simulation of linear structures by the Finite Element Method (FEM). It assembles the "converged" lecture notes of **Introduction to Finite Element Methods** or IFEM. This is a core graduate course offered in the Department of Aerospace Engineering Sciences of the University of Colorado at Boulder.

IFEM was first taught on the Fall Semester 1986 and has been repeated every year since. It is taken by both first-year graduate students as part of their M.S. or M.E. requirements, and by senior undergraduates as technical elective. Selected material in Chapters 1 through 3 is used to teach a two-week introduction of Matrix Structural Analysis and Finite Element concepts to junior undergraduate students who are taking their first Mechanics of Materials course.

Prerequisites for the graduate-level course are multivariate calculus, linear algebra, a basic knowledge of structural mechanics at the Mechanics of Materials level, and some familiarity with programming concepts learnt in undergraduate courses.

The course originally used Fortran 77 as computer implementation language. This has been gradually changed to *Mathematica* since 1995. The changeover is now complete. No prior knowledge of *Mathematica* is required because that language, unlike Fortran or similar low-level programming languages, can be picked up while "going along." Inasmuch as *Mathematica* supports both symbolic and numeric computation, as well as direct use of visualization tools, the use of the language is interspersed throughout the book.

## Book Objectives

"In science there is only physics; all the rest is stamp collecting" (Lord Kelvin). The quote reflects the values of the mid-XIX century. Even now, at the dawn of the XXIth, progress and prestige in the natural sciences favors fundamental knowledge. By contrast, engineering knowledge consists of three components:[1]

1. Conceptual knowledge: understanding the framework of the physical world.

2. Operational knowledge: methods and strategies for formulating, analyzing and solving problems, or "which buttons to push."

3. Integral knowledge: the synthesis of conceptual and operational knowledge for technology development.

The language that connects conceptual and operational knowledge is mathematics, and in particular the use of mathematical models. Most engineering programs in the USA correctly emphasize both conceptual and operational components. They differ, however, in how well the two are integrated. The most successful curricula are those that address the tendency to "horizontal disconnection" that bedevils engineering students suddenly exposed to a vast array of subjects.

Integral knowledge is unique to the engineering profession. Synthesis ability is a personal attribute that cannot be coerced, only encouraged and cultivated, the same as the best music programs do not

---

[1] Extracted from: B. M. Argrow, Pro-active teaching and learning in the Aerospace Engineering Sciences Curriculum 2000, internal report, University of Colorado, February 2001.

automatically produce Mozarts. Studies indicate no correlation between good engineers and good students.[2] The best that can be done is to provide an adequate (and integrated) base of conceptual and operational knowledge to potentially good engineers.

Where does the Finite Element Method (FEM) fit in this framework?

FEM was developed initially, and prospered, as a computer-based simulation method for the analysis of aerospace structures. Then it found its way into both design and analysis of complex structural systems, not only in Aerospace but in Civil and Mechanical Engineering. In the late 1960s it expanded to the simulation of non-structural problems in fluids, thermomechanics and electromagnetics. This "Physical FEM" is an *operational tool*, which fits primarily the operational knowledge component of engineering, and draws from the mathematical models of the real world. It is the form emphasized in the first part of this book.

The success of FEM as a general-purpose simulation method attracted attention in the 1970s from two quarters beyond engineering: mathematicians and software entrepreneurs. The world of FEM eventually split into applications, mathematics, and commercial software products. The former two are largely housed in the comfortable obscurity of academia. There is little cross-talk between these communities. They have different perpectives. They have separate constituencies, conferences and publication media, which slows down technology transfer. As of this writing, the three-way split seems likely to continue, as long as there is no incentive to do otherwise.

This book aims to keep a presentation balance: the physical and mathematical interpretations of FEM are used eclectically, with none overshadowing the other. Key steps of the computer implementation are presented in sufficient detail so that a student can understand what goes on behind the scenes of a "black box" commercial product. The goal is that students navigating this material can eventually feel comfortable with any of the three "FEM communities" they come in contact during their professional life, whether as engineers, managers, researchers or teachers.

## Book Organization

The book is divided into four Parts. The first three are of roughly similar length.

**Part I: The Direct Stiffness Method**. This part comprises Chapters 1 through 11. It covers major aspects of the Direct Stiffness Method (DSM). This is the most important realization of FEM, and the one implemented in general-purpose commercial finite element codes used by practicing engineers. Following a introductory first chapter, Chapters 2-4 present the fundamental steps of the DSM as a matrix method of structural analysis. A plane truss structure is used as motivating example. This is followed by Chapters 5-10 on programming, element formulation, modeling issues, and techniques for application of boundary conditions. Chapter 11 deals with relatively advanced topics including condensation and global-local analysis. Throughout these chapters the physical interpretation is emphasized for pedagogical convenience, as unifying vision of this "horizontal" framework.

**Part II: Formulation of Finite Elements**. This part extends from Chapters 12 through 19. It is more focused than Part I. It covers the development of elements from the more general viewpoint of the variational (energy) formulation. The presentation is inductive, always focusing on specific elements and progressing from the simplest to more complex cases. Thus Chapter 12 rederives the

---

[2] As evaluated by conventional academic metrics, which primarily test operational knowledge. One difficulty with teaching synthesis is that good engineers and designers are highly valued in industry but rarely comfortable in academia.

plane truss (bar) element from a variational formulation, while Chapter 13 presents the plane beam element. Chapter 14 introduces the plane stress problem, which serves as a testbed for the derivation of two-dimensional isoparametric elements in Chapter 15 through 18. This part concludes with an overview of requirements for convergence.

**Part III: Computer Implementation**. Chapters 20 through 29 deal with the computer implementation of the finite element method. Experience has indicated that students profit from doing computer homework early. This begins with Chapter 5, which contains an Introduction to *Mathematica*, and continues with homework assignments in Parts I and II. The emphasis changes in Part III to a systematic description of components of FEM programs, and the integration of those components to do problem solving.

**Part IV: Structural Dynamics**. This part, which starts at Chapter 30, is under preparation. It is intended as a brief introduction to the use of FEM in structural dynamics and vibration analysis, and is by nature more advanced than the other Parts.

### Exercises

Most Chapters are followed by a list of homework exercises that pose problems of varying difficulty. Each exercise is labeled by a tag of the form

```
[type:rating]
```

The type is indicated by letters A, C, D or N for exercises to be answered primarily by analytical work, computer programming, descriptive narration, and numerical calculations, respectively. Some exercises involve a combination of these traits, in which case a combination of letters separated by + is used; for example A+N indicates analytical derivation followed by numerical work. For some problems heavy analytical work may be helped by the use of a computer-algebra system, in which case the type is identified as A/C.

The rating is a number between 5 and 50 that estimates the degree of difficulty of an Exercise, in the following "logarithmic" scale:

5    A simple question that can be answered in seconds, or is already answered in the text if the student has read and understood the material.

10    A straightforward question that can be answered in minutes.

15    A relatively simple question that requires some thinking, and may take on the order of half to one hour to answer.

20    Either a problem of moderate difficulty, or a straightforward one requiring lengthy computations or some programming, normally taking one to six hours of work.

25    A scaled up version of the above, estimated to require six hours to one day of work.

30    A problem of moderate difficulty that normally requires on the order of one or two days of work. Arriving at the answer may involve a combination of techniques, some background or reference material, or lenghty but straightforward programming.

40    A difficult problem that may be solvable only by gifted and well prepared individual students, or a team. Difficulties may be due to the need of correct formulation, advanced mathematics, or high level programming. With the proper preparation, background and tools these problems may be solved in days or weeks, while remaining inaccessible to unprepared or average students.

50    A research problem, worthy of publication if solved.

Most Exercises have a rating of 15 or 20. Assigning three or four per week puts a load of roughly 5-10 hours of solution work, plus the time needed to prepare the answer material. Assignments of difficulty 25 or 30 are better handled by groups, or given in take-home exams. Assignments of difficulty beyond 30 are never assigned in the course, but listed as a challenge for an elite group.

Occasionally an Exercise has two or more distinct but related parts identified as items. In that case a rating may be given for each item. For example: [A/C:15+20]. This does not mean that the exercise as a whole has a difficulty of 35, because the scale is roughly logarithmic; the numbers simply rate the expected effort per item.

## Selecting Course Material

The number of chapters has been coordinated with the 28 lectures and two midterm exams of a typical 15-week semester course offered with two 75-minute lectures per week. The expectation is to cover one chapter per lecture. Midterm exams cover selective material in Parts I and II, whereas a final exam covers the entire course. It is recommended to make this final exam a one-week take-home to facilitate computer programming assignments. Alternatively a final term project may be considered. The experience of the writer, however, is that term projects are not useful at this level, since most first-year graduate students lack the synthesis ability that develops in subsequent years.

The writer has assigned weekly homeworks by selecting exercises from the two Chapters covered in the week. Choices are often given. The rating may be used by graders to weight scores. Unlike exams, group homeworks with teams of two to four students are recommended. Teams are encouraged to consult other students, as well as the instructor and teaching assistants, to get over gaps and hurdles. This group activity also lessen schedule conflicts common to working graduate students.

Feedback from course offerings as well as advances in topics such as programming languages resulted in new material being incorporated at various intervals. To keep within course coverage constraints, three courses of action were followed in revising the book.

**Deleted Topics**. All advanced analysis material dealing with variational calculus and direct approximation methods such as Rayleigh-Ritz, Galerkin, least squares and collocation, was eliminated by 1990. The few results needed for Part II are stated therein as recipes. That material was found to be largely a waste of time for engineering students, who typically lack the mathematical background required to appreciate the meaning and use of these methods in an application-independent context.[3] Furthermore, there is abundant literature that interested students may consult should they decide to further their knowledge in those topics for self-study or thesis work.

**Appendices**. "Refresher" material on vector and matrix algebra has been placed on Appendices A through D. This is material that students are supposed to know as a prerequisite. Although most of it is covered by a vast literature, it was felt advisable to help students in collecting key results for quick reference in one place, and establishing a consistent notational system.

**Starred Material**. Chapter-specific material that is not normally covered in class is presented in fine print sections marked with an asterisk. This material belong to two categories. One is extension to the basic topics, which suggest the way it would be covered in a more advanced FEM course. The other includes general exposition or proofs of techniques presented as recipes in class for expedience or time constraints. Starred material may be used as source for term projects or take-home exams.

---

[3] This is a manifestation of the disconnection difficulty noted at the start of this Preface.

The book organization presents flexibility to instructors in organizing the coverage for shorter courses, for example in a quarter system, as well as fitting a three-lectures-per-week format. For the latter case it is recommended to cover two Chapters per week, while maintaining weekly homework assignments. In a quarter system a more drastic condensation would be necessary; for example much of Part I may be left out if the curriculum includes a separate course in Matrix Structural Analysis, as is common in Civil and Architectural Engineering.

## Acknowledgements

# Chapter Contents