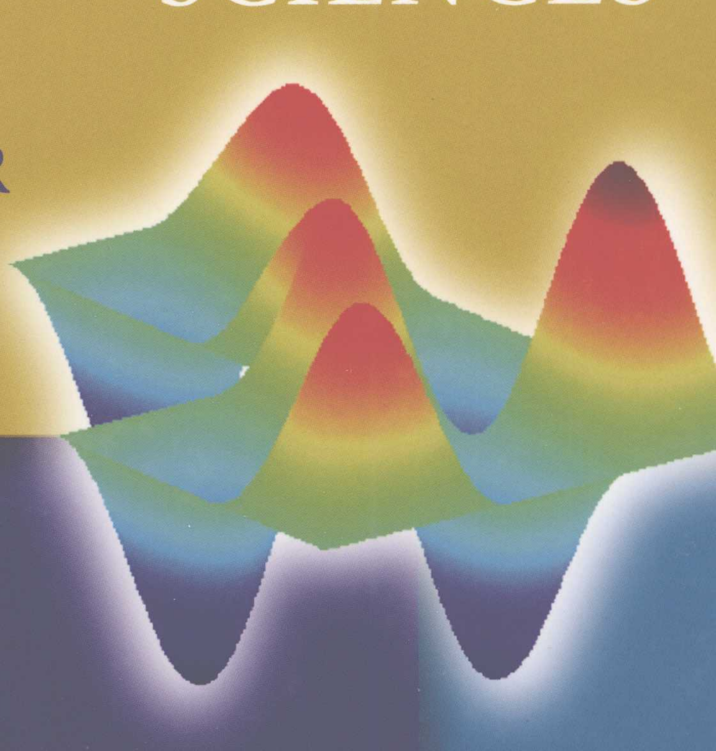
FINITE ELEMENTS

Computational ENGINEERING SCIENCES

A. J. BAKER



WILEY

Finite Elements ⇕ Computational Engineering Sciences

A. J. Baker

Professor Emeritus
The University of Tennessee
USA

WILEY

A John Wiley & Sons, Ltd., Publication

Finite Elements
⇕
Computational Engineering Sciences

This edition first published 2012

Registered office

John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom

For details of our global editorial offices, for customer services and for information about how to apply for permission to reuse the copyright material in this book please see our website at www.wiley.com.

Library of Congress Cataloguing-in-Publication Data

Baker, A. J., 1936-
Finite elements : computational engineering sciences / A.J. Baker.
p. cm.
Includes bibliographical references and index.
ISBN 978-1-119-94050-0 (cloth)
1. Finite element method. I. Title.
TA347.F5B3423 2012
620.001'51825–dc23

2012011745

A catalogue record for this book is available from the British Library.

ISBN: 9781119940500

Set in 10/12.5 pt, Palatino-Roman by Thomson Digital, Noida, India
Printed in Singapore by Ho Printing Singapore Pte Ltd

Yogi Berra is quoted,
"If you come to a fork in the road, take it,"
so I did and ended up here.

Preface

The computer revolution has profoundly impacted how engineers and scientists conduct professional activities. In the early 1960s, a computer fully occupied and amply heated (!) a space the size of a classroom. The PC, introduced in the mid-1970s, was a "toy." Yet by the millennia, Linux clusters of cheap gigahertz–gigabyte PCs could execute truly large-scale computational simulations. Indeed, the "desktop Cray," fantasized in ~1980, was here and was truly inexpensive!

The companion maturation of theory and practice in the computational engineering sciences has been an evolutionary (not revolutionary!) process. It remains highly fragmented by discipline, even though computational fluid dynamics (CFD) and computational structural mechanics (CSM) emerged simultaneously from the research laboratory in the late 1950s. The former relied on finite difference (FD) methods to convert theory to computable form. Conversely, the latter's classical *virtual work* foundation enabled a calculus-based finite element (FE) theory implementation of the underlying variational principle extremum. Finally, in chemical engineering collocation methods were developed for process simulations, and at first glance these theories appear absolutely "linearly independent."

Research now completed has proven that practically all developments supporting the *computational engineering sciences* can be formulated from the *extremum* of the mathematician's *weak form theory* termed a *weak statement* (WS). The weak form process enables *theorization* to be completed in the *continuum*, using calculus, vector field theory, and modern approximation concepts. When finished, the discrete implementation of the theory extremum can be formed using FE, FD, and/or finite volume (FV) procedures. The FE implementation is typically guaranteed *optimal* in its performance, that is, accuracy, asymptotic convergence rate, and so on. Furthermore, FE methodology leads to *precise constructions* devoid of heurism, since integral–differential calculus is used rather than difference algebra to generate the algebraic statement amenable to computing.

This text develops discrete implementations of WS theory for a diverse variety of problem statements in the *computational engineering sciences*. Unique to the FE discrete development, the resulting *algorithms* are immediately stated in *computable* form via a transparent, object-oriented programming syntax. The engineering science problem classes developed herein include

- heat conduction
- structural mechanics
- mechanical vibrations
- heat transfer, with convection and radiation
- fluid mechanics
- heat/mass convective transport

The text is organized into twelve chapters. Following an introduction, and some very pertinent overview material, an elementary heat conduction *tutorial* clearly illustrates all element matrix constructs, the "famous" assembly algorithm and the concept of error estimation and measurement. Subsequent chapter pairs develop expository one-dimensional, then general n-dimensional FE WS implementations in each continuum engineering sciences discipline.

The sequence of developments serves to illustrate, examine, and generalize the available theoretical error estimates, with the concept of a *norm* central to this process. In moving to the convection–diffusion problem class, a sequence of Taylor series manipulations leads to *modified* conservation principle expressions, expressed in the *continuum*, which collectively improve asymptotic convergence rate coupled with annihilation of significant order discretization-induced *phase lag* and *dispersive* error mechanisms.

Incisive computer lab experiments complement each development, with principle focus to gain a firm usable understanding of approximation error mechanisms as influenced by data *nonsmoothness*, problem *nonlinearity*, stability, dispersion error and boundary conditions, each impacted by the selected FE basis completeness degree. The n-dimensional computer experiments focus on refinements for error nuances associated with nonconvex boundaries, phase lag, and artificial *numerical diffusion*. An intervening brief chapter clearly identifies the connections between FD, FV, and FE discrete implementations for a Poisson equation in n-dimensions.

Engineers are clearly of the opinion that, "theory is fine, but show me the numbers!," which requires theory conversion to code practice. Since the FE-implemented WS theory is highly organized, the algorithm statement in any discipline ends up constituted of six, and *only* six, types of data to convert theory to practice. Capitalizing on *object-oriented* concepts, these six data types are organized into a *template* such that the *computing statement*, including explicit nonlinearity, is unambiguously expressible.

In summary, this text fully develops modern FE discrete algorithms for the computational engineering sciences with applications aimed to available and emergent *problem solving environments* (PSEs). Its organization and content has evolved from two decades of teaching the subject at UT. This text fully obsoletes the predecessor 1991 text *Finite Elements* 1-2-3, marketed with a "spaghetti" Fortran PC code on a 5.25 inch floppy disk.

All computer lab exercise MATLAB® .m files along with the specifically written MATLAB® toolbox *FE*m*PSE* are available for download from www.wiley.com/go/baker/finite. The .mph files for the COMSOL design studies may be downloaded from their user community web site www.comsol.com/community/exchange/?page=2. University faculty interested in presenting the internet-enabled academic course from which this text was generated will find complete support materials available at www.wiley.com/go/baker/finite.

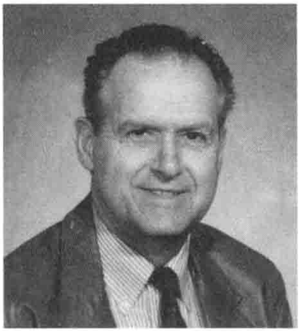
Many colleagues and graduate students have contributed to the creation and refinement of text content. My thinking formality on the subject has benefited from a multidecade collegial association with Prof. J. Tinsley Oden. I owe a deep debt of gratitude to my Computational Mechanics Corp. co-founders Paul Manhardt, who invented the template concept, and Joe Orzechowski, who assimilated templates into reliable computational syntax for mainly CFD applications.

The dissertation research of Dr. Jin Kim, Dr. Subrata Roy, Dr. David Chaffin, Dr. Alexy Kolesnikov, and Dr. Sunil Sahu collectively formalized the improved theoretical and practical understanding of FE algorithm performance nuances detailed herein. Dr. Zac Chambers and Dr. Marcel Grubert along with Messrs. Mike Taylor and Shawn Ericson contributed significantly to polishing these fundamental underlying precepts to pedagogical acceptability.

A. J. Baker
Knoxville, TN
January 2012

Note: All color originals are accessible at www.wiley.com/go/baker/finite.

About the Author



A. J. Baker, PhD, PE, left commercial aerospace research to join the University of Tennessee College of Engineering in 1975, to lead academic research in the exciting new field of CFD (computational fluid dynamics). Now Professor Emeritus and still Director, UT CFD Laboratory (http://cfdlab.utk.edu), his professional career started as a mechanical engineer with Union Carbide Corp. The challenges there prompted resigning after 5 years to enter graduate school full time in 1963 with the goal to "learn what a computer was and could do." The introduction involved *driving* an IBM 1620 with 5 kB memory and no disk pack! A 1967 summer job with Bell Aerospace Company required assessing the *first* publication claiming unsteady heat conduction was amenable to finite element analysis. This led to the 1968 Bell Aerospace technical memorandum, "A Numerical Solution Technique for a Class of Two-dimensional Problems in Fluid Dynamics Formulated via Discrete Elements," a truly pioneering expose in the fledgling FE CFD field. Finishing his dissertation in 1970, he joined Bell Aerospace as Principal Research Scientist to pursue full-time finite element methods in CFD. NASA Langley contracts with summer appointments at ICASE led to a visiting professorship at Old Dominion University, 1974–1975, from which he moved directly to UT forming Computational Mechanics Consultants, Inc., with two Bell colleagues, to assist converting academic FE CFD research progress into computing practice.

FE ⇔ Computational Engineering Sciences with hands-on computing:
This is the first *introductory* level text to fully integrate the underlying *theory* with *hands-on* computer experiments supported by the MATLAB® and COMSOL® Problem Solving Environments (PSEs). You may download all .m and .mph files supporting each suggested computer experiment, also eight topical lectures for video-streaming on your PC available from www.wiley.com/go/baker/finite. The academic course engendering the text technical content became totally distance-enabled on Internet in 2005. Academics interested in presenting this course at their institution may acquire the complete academic support material at www.wiley.com/go/baker/finite.

Notations

a	expansion coefficient
A	plane area; one-dimensional FE matrix prefix; coefficient
A	generic square matrix
[A]	factored global matrix
b	coefficient; boundary condition subscript; body force component, generic column matrix
{b}	global data matrix
B	two-dimensional FE matrix prefix
B	body force, structural FE matrix
c	coefficient; specific heat
C	three-dimensional FE matrix prefix, constant, Courant number
d	coefficient; FE matrix indicator
D	diagonal matrix, diffusion coefficient
[DIFF]	global diffusion matrix
DOF	approximation degrees-of-freedom
e	element-dependent; unit vector component, error
$e(\cdot)$	error, a function of $(\cdot)$
e^N	approximation error
e^h	discrete approximation error
eta_{ji}	coordinate transformation data
E	energy seminorm (subscript), elastic modulus
E	Hooke's law matrix
F	radiation viewfactor
F	applied force, flux on $\partial\Omega$
f	kinetic flux vector
FD	finite difference
FE	finite element
FV	finite volume
{F}	homogeneous form of a discretized weak statement
g	gravity magnitude
g	gravity

G	elastic shear modulus, amplification factor, Gebhart factor
Gr	Grashoff number
GWS	Galerkin weak statement
h	discretization (superscript), heat transfer coefficient, measure
H	Gauss quadrature weight; Hilbert space
[BC]	boundary condition matrix
i	summation index, mesh node, imaginary unit
$\hat{\mathbf{i}}$	unit vector parallel to x
I	moment of inertia; element matrix summation index
[I]	identity (diagonal) matrix
j	summation index, mesh node
$\hat{\mathbf{j}}$	unit vector parallel to y
J	template summation index
[J]	coordinate transformation jacobian
[JAC]	jacobian
k_{ij}	element of the [DIFF] and/or [STIFF] matrix
k	thermal conductivity, basis degree, index, diffusion coefficient,
k	spring constant
$\bar{k}$	average value of conductivity
$\hat{\mathbf{k}}$	unit vector parallel to z
K	template matrix summation index, viewfactor kernel
ℓ	element length; summation index
$\ell(\cdot)$	differential equation on $\partial\Omega$
L	domain span, length measure, lower triangular matrix, lagrangian
$\mathcal{L}(\cdot)$	differential equation on Ω
m	integer
m_i	point mass
mGWS	Taylor series-modified Galerkin weak statement
mPDE	Taylor series-modified conservation principle PDE
M	elements in Ω^h; moment; matrix prefix; particle system mass
M	iteration matrix
[MASS]	global mass matrix
n	index; normal subscript; dimension of domain Ω; integers, normal coordinate, time index (subscript)
$\hat{\mathbf{n}}$	outward pointing unit vector normal to $\partial\Omega$
N	matrix prefix
N	summation termination; approximation (superscript), iteration matrix
NC	natural coordinate basis
$\{N_k\}$	finite element basis of degree k non-D non-dimensional
p	load (data); pressure, iteration index
P	point load; Gauss quadrature order
$\{P\}$	computational matrix, distributed load DOF
Pa	non-D parameter on Ω
Pb	non-D parameter on $\partial\Omega$

Pr	Prandtl number
q	generalized dependent variable
Q	discretized dependent variable; heat added
$\{Q\}$:	approximation DOF matrix
r	reference state subscript; radius
Re	Reynolds number
R^+	the positive real axis
$\Re^n$	Euclidean space
{RES}	global matrix statement residual
s	source term on Ω; heat added, tangent coordinate
$\mathbf{s}$	unit vector tangent to $\partial\Omega$
S	finite element assembly operator; entropy
SOR	successive over-relaxation
{S}	computational matrix
t	time
T	temperature, kinetic energy
T_c	convection heat transfer exchange temperature
T_r	radiation heat transfer exchange temperature
$\mathbf{T}$	surface traction vector
T^N	approximate temperature solution
TE	truncation error
TP	tensor product basis
TS	Taylor series
$\mathbf{u}$	displacement vector; velocity vector
U	upper triangular matrix
u	velocity x component; speed
U	discretized speed DOF, phase velocity (speed)
[VEL]	global fluid convection matrix
v	velocity y component
V	shear force; volume; potential energy
$\mathbf{V}$	velocity
w	weight function; fin thickness; velocity z component
W	weight; work done by system
WF	weak form
WS	weak statement
x	generic unknown
x, x_i	cartesian coordinate, coordinate system $1 \leq i \leq n$
$\bar{x}$	transformed local coordinate
X	discrete cartesian coordinate
y	displacement; cartesian coordinate
Y	discrete cartesian coordinate
z	cartesian coordinate
Z	thickness ratio; discrete cartesian coordinate
$(\cdot)$	scalar (number)

$\{\cdot\}$	column matrix
$\{\cdot\}^T$	row matrix
$[\cdot]$	square matrix
$\|\cdot\|$	norm
$\cup$	union (non-overlapping sum)
$\cap$	intersection
$\det[\cdot]$	matrix determinant
sym	symmetric
α	coefficient
β	coefficient
γ	shear strain, coefficient
δ_{ij}	Kronecker delta
δQ	iterate
Δ	discrete increment
ε	normal strain, emissivity
ϕ	electric potential, flow potential
$\phi(\cdot)$	trial space function; potential function
$\boldsymbol{\Phi}$	potential function
$\boldsymbol{\Phi}_\beta(\mathbf{x})$	test space
$\boldsymbol{\Psi}_\alpha(\mathbf{x})$	trial space
$\boldsymbol{\eta}$	coordinate system in transform space
η_i	tensor product coordinate system
κ	thermal diffusivity, wave number
$\boldsymbol{\kappa}_{\alpha\beta}$	element of a square matrix
λ	Lamé parameter, wavelength
μ	Lamé parameter, dynamic viscosity
υ	Poisson ratio, kinematic viscosity
$O(\cdot)$	order of $(\cdot)$
π	pi (3.1415926 . . .)
θ	time integration implicitness factor
Θ	potential temperature
ρ	density, absorbtivity
$d\sigma$	differential element on $\partial\Omega$
$d\tau$	differential element on Ω
τ	normal stress
ω	frequency
Ω	domain of differential equation
Ω_e	finite element domain
$\boldsymbol{\Omega}^h$	discretization of Ω
$\partial\Omega$	boundary of Ω
ζ_α	natural coordinate system
$d(\cdot)/dx$	ordinary derivative
$\partial(\cdot)/\partial x$	partial derivative
∇	vector derivative
∇^2	laplacian derivative operator

Contents