

Electric Machines

Theory and Analysis Using the
Finite Element Method

DIONYSIOS ALIPRANTIS
OLEG WASYNCZUK

Electric Machines

Offering a new perspective, this textbook demystifies the operation of electric machines by providing an integrated understanding of electromagnetic fields, electric circuits, numerical analysis, and computer programming. It presents fundamental concepts in a rigorous manner, emphasizing underlying physical modeling assumptions and limitations, and provides detailed explanations of how to implement the finite element method to explore these concepts using Python. It includes explanations of the conversion of concepts into algorithms, and algorithms into code, and examples building in complexity, from simple linear-motion electromagnets to rotating machines. Over 100 theoretical and computational end-of-chapter exercises test understanding, with solutions for instructors and downloadable Python code available online.

Ideal for graduates and senior undergraduates studying electric machines, electric machine design and control, and power electronic converters and power systems engineering, this textbook is also a solid reference for engineers interested in understanding, analyzing, and designing electric motors, generators, and transformers.

Dionysios Aliprantis is Professor of Electrical and Computer Engineering at Purdue University, West Lafayette, Indiana. He has developed and taught numerous undergraduate and graduate courses on electric machines, power electronics, and power systems. He is a Senior Member of the IEEE.

Oleg Wasynczuk is Professor of Electrical and Computer Engineering at Purdue University, West Lafayette, Indiana, and Chief Technical Officer of PC Krause and Associates, Inc. He is a Fellow of the IEEE.

“This book provides an innovative approach that is very relevant, particularly from the electric machine designer’s perspective. After providing relevant background to vector and variational calculus, the finite element method and electromagnetic fields, their application to the analysis of electric machines at the design stage is illustrated by application to actual machines. The theoretical material presented is supplemented with numerous solved examples, building blocks of computer code, and exercise problem sets throughout the book to facilitate understanding for the reader. In this respect it is unique, as the common books on electric machines treat the material from the circuit analysis view rather than the basic field perspective as done here.”

Om Malik, University of Calgary

“This text provides a valuable explanation of the theoretical and practical issues of solving for electric and magnetic fields in devices that perform energy conversion and position control. From fundamental Maxwell’s equations to actual industrial-grade software technics of the finite element method including nonlinearities such as hysteresis, the text gives a path to understanding and implementation.”

Peter Sauer, University of Illinois at Urbana-Champaign

Electric Machines

Theory and Analysis Using the Finite Element Method

DIONYSIOS ALIPRANTIS

Purdue University

OLEG WASYNCZUK

Purdue University



CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

One Liberty Plaza, 20th Floor, New York, NY 10006, USA

477 Williamstown Road, Port Melbourne, VIC 3207, Australia

314–321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre,
New Delhi – 110025, India

103 Penang Road, #05–06/07, Visioncrest Commercial, Singapore 238467

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of
education, learning, and research at the highest international levels of excellence.

www.cambridge.org

Information on this title: www.cambridge.org/highereducation/isbn/9781108423748

DOI: [10.1017/9781108529280](https://doi.org/10.1017/9781108529280)

© Dionysios Aliprantis and Oleg Wasynczuk 2023

This publication is in copyright. Subject to statutory exception
and to the provisions of relevant collective licensing agreements,
no reproduction of any part may take place without the written
permission of Cambridge University Press.

First published 2023

Printed in the United Kingdom by TJ Books Ltd, Padstow, Cornwall, 2023

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

ISBN 978-1-108-42374-8 Hardback

Additional resources for this publication at www.cambridge.org/Aliprantis_Wasynczuk

Cambridge University Press has no responsibility for the persistence or accuracy of
URLs for external or third-party internet websites referred to in this publication
and does not guarantee that any content on such websites is, or will remain,
accurate or appropriate.

To Nadia, Alcmene, and Christopher (DA)

To my family (OW)

Contents

	<i>Preface</i>	<i>page xi</i>
	<i>Nomenclature</i>	<i>xv</i>
1	Review of Vector Calculus and Electromagnetic Fields	1
1.1	Overview of Maxwell's Equations	2
1.2	Vectors	3
1.2.1	Dot Product	4
1.2.2	Cross Product	8
1.2.3	Vectors in Python	11
1.2.4	Triple Products	13
1.3	Scalar and Vector Fields	15
1.4	Integrals of Scalar and Vector Fields	20
1.4.1	Line Integrals	20
1.4.2	Surface Integrals	23
1.4.3	Volume Integrals	26
1.5	Differential Operators on Scalar and Vector Fields	27
1.5.1	Gradient	27
1.5.2	Divergence	30
1.5.3	Curl	33
1.5.4	The Laplace Operator	41
1.5.5	Useful Identities	42
1.6	Integral Laws	47
1.6.1	The Divergence Theorem or Gauss' Law	47
1.6.2	Stokes' Theorem	52
1.6.3	Green's Identities	53
1.6.4	Green's Theorem in Two Dimensions	55
1.6.5	Helmholtz's Theorem	56
1.7	Case Studies on the Magnetostatic Field	63
1.7.1	Field from a Circular Loop and the Magnetic Dipole	64
1.7.2	Torque on a Magnetic Dipole	66
1.7.3	Field from a Distribution of Magnetic Dipoles	66
1.7.4	The Relationship between B and H in Materials	68
1.7.5	Two-Dimensional Problems	71
1.7.6	Axisymmetric Problems	78

1.7.7	An Elementary Three-Dimensional Problem Solved Using the Method of Moments	81
1.8	Summary	92
1.9	Further Reading	92
	Problems	92
	References	93
2	Variational Form of Maxwell's Equations, Energy, and Force	96
2.1	Calculus of Variations	97
2.1.1	The Euler–Lagrange Equation for a One-Dimensional Problem	98
2.1.2	The Euler–Lagrange Equation for a Two-Dimensional Problem	102
2.1.3	Problems in the Time Domain	104
2.2	Solving Maxwell's Equations with Variational Calculus	107
2.2.1	Functional for Poisson's Equation	107
2.2.2	The Electrostatic Problem	108
2.2.3	The Magnetostatic Problem	112
2.3	Energy Transfer Pathways	120
2.3.1	Conservation of Energy and Poynting's Theorem of Energy Flow	121
2.3.2	Energy Transfer through a Rotating Magnetic Field	122
2.3.3	Energy Balance in Conductors	140
2.3.4	The Coupling Field Energy	146
2.3.5	Core Loss	148
2.3.6	Circuit-Based Energy Flow Analysis	154
2.3.7	Mechanical Energy Transfer: Electromagnetic Force and Torque	162
2.4	Force and Torque Calculations from the Fields	178
2.4.1	Preliminary Modeling Assumptions	178
2.4.2	Torque Calculation Based on a Virtual Distortion of the Air Gap	181
2.4.3	Conservation of Electromagnetic Momentum: The Electro-magnetic (Maxwell's) Stress Tensor	182
2.4.4	Force and Torque from Conservation of Energy	200
2.5	Summary	214
2.6	Further Reading	214
	Problems	214
	References	216
3	The Finite Element Method	219
3.1	Solution of Laplace's Equation Using the FEM	220
3.1.1	Functional Stationarity and the Weak Form	221
3.1.2	Single-Element Preliminary Calculations	221
3.1.3	Functional Contribution from a Single Element	223
3.1.4	Assembly of Global System Equation	225
3.2	Solution of Linear Poisson's Equation Using the FEM	230
3.2.1	Setting up the System of Equations	230
3.2.2	Functional Minimization and the Weak Form	234

3.2.3	Minimization of a Quadratic Functional	235
3.2.4	Handling of Boundary Conditions	237
3.3	Building Algorithms for Linear Magnetics	238
3.3.1	Data Structures	238
3.3.2	Building the Stiffness Matrix	242
3.3.3	A First FEM Solver	247
3.4	Calculation of Flux Linkage and Inductance	257
3.4.1	The Average MVP Method for Flux Linkage Calculations	257
3.4.2	Inductance Calculations Using the Flux Linkage	261
3.4.3	Converting Between Physical and Modeled Current Density	262
3.5	Solution of Nonlinear Poisson's Equation Using the FEM	273
3.5.1	Fixed-Point Method	275
3.5.2	Newton–Raphson Method	276
3.6	Building Algorithms for Nonlinear Magnetics	283
3.6.1	Modeling Nonlinear Materials	283
3.6.2	Building the Gradient and Hessian Matrix	288
3.6.3	FEA Program Implementation with Nonlinear Magnetics	291
3.7	Devices with Permanent Magnets	293
3.8	FEA of Axisymmetric Problems	294
3.8.1	Magnetically Linear Axisymmetric Problems	294
3.8.2	Nonlinear Axisymmetric Problems	296
3.8.3	Flux Linkage Calculation	296
3.9	Galerkin's Method	297
3.9.1	Distinguishing Features of Galerkin's Method	297
3.9.2	Energetic Interpretation of Galerkin's Method	300
3.9.3	Galerkin's Method for the Nonlinear Poisson Equation in Magnetic Devices	302
3.10	Summary	306
3.11	Further Reading	307
	Problems	307
	References	309
4	Electric Machine FEA Implementation Guidelines	311
4.1	Force and Torque Calculation	311
4.1.1	Coenergy Method	312
4.1.2	Maxwell Stress Tensor Methods	315
4.1.3	Virtual Displacement Method	321
4.1.4	An Air-Gap MVP-Based Method	330
4.2	Rotation without Remeshing	333
4.3	Wound-Rotor Synchronous Machines	336
4.3.1	Elementary Two-Phase Round-Rotor Synchronous Machine: Fundamentals of Distributed Windings	337
4.3.2	Elementary Three-Phase Salient-Rotor Synchronous Machine: Basics of Two-Reaction Theory	356

4.3.3	Analysis in qd Variables Using Park's Transformation	363
4.3.4	Derivation of a WRS qd Equivalent Circuit from FEA	376
4.4	Permanent-Magnet Synchronous Machines	381
4.4.1	Elementary Surface-Mounted PMSM: Principles of Operation	381
4.4.2	Derivation of a PMSM qd Equivalent Circuit from FEA	387
4.4.3	Interior PMSM: Fundamentals of qd -Current Control	390
4.5	Switched Reluctance Machines	403
4.5.1	Magnetostatic Analysis	409
4.6	Exploiting Periodicity	413
4.6.1	Analysis of a Single Pole-Pair (Periodic Conditions)	414
4.6.2	Analysis of a Single Pole (Antiperiodic Conditions)	417
4.7	Summary	419
4.8	Further Reading	420
	Problems	420
	References	423
5	Problems in the Time Domain	425
5.1	Physics of Induced Currents	425
5.1.1	Electric Field in Two-Dimensional Problems	426
5.1.2	Motional EMF in Two-Dimensional Problems	427
5.2	Formulation of Two-Dimensional FEA in the Time Domain	427
5.3	Eddy Current Problems with Analytical Solutions	429
5.3.1	Eddy Currents in a Semi-Infinite Plate	429
5.3.2	Elliptical Model of Magnetic Hysteresis	435
5.3.3	The Complex Poynting Vector	437
5.3.4	The Semi-infinite Plate as a Conductor	444
5.3.5	Semi-infinite Plate with Twin Return Conductor	447
5.3.6	Circular Conductor	448
5.4	Phasor-Based FEA	449
5.4.1	Phasor-Based FEA in Lossy Materials	451
5.4.2	Calculating the B -Field Phasors and the Core Loss	452
5.5	Higher-Frequency Winding Model	463
5.6	Time-Stepping FEA	465
5.6.1	Time Stepping Using Newton's Method	466
5.6.2	Time Stepping with External Circuit (No Eddy Currents, No Motion)	468
5.6.3	Time Stepping with External Circuit and Eddy Currents (No Motion)	477
5.6.4	Time Stepping With Motion	478
5.7	Summary	481
5.8	Further Reading	481
	Problems	481
	References	482
	Notes	483
	Index	487

Preface

Ἀρχὴ γὰρ λέγεται μὲν ἥμισυ παντός
“For it is said that the beginning is the half of every work”
Plato¹

This book is about devices that convert electrical to mechanical energy and vice versa. In the former case, we have an electric motor; in the latter case, an electric generator. Motors, generators, and transformers (which lack moving parts) are what we commonly call “electric machines.” A tremendous number of electric machines have been invented over the last 200 years, so it would be futile to attempt a detailed analysis of each different type. Instead, we should provide engineers with an understanding of how common operating principles are obtained from classical laws of electromagnetism, regardless of rating, size, or application domain.

Hence, in this text everything is explained based on the electromagnetic field permeating a machine, which is governed by Maxwell’s partial differential equations. So the field plays the leading role, but the supporting role unquestionably goes to the finite element method (FEM) that computes the field. The FEM is a game changer because it allows us to take a look inside real machines with unparalleled resolution and accuracy. This textbook presents a fields-based theory of electric machines jointly with the FEM and its implementation using the Python programming language, in a bid to approach this subject in a pedagogical and, hopefully, more entertaining manner.

Electric Machines in the Twenty-First Century

The scientific area of electromechanical energy conversion is as vibrant as ever. Cost-effective, efficient, and reliable electric machines play a key role in addressing modern society’s grand challenge of electrifying our energy and transportation infrastructures to combat climate change. Rotating generators still produce most of our electricity worldwide, while the penetration of semiconductor-based photovoltaic generation remains relatively low. In addition, the proliferation of new hydro, wind, and marine energy ap-

plications has increased demand for innovative generator designs. On the other hand, it is estimated that electric motors, such as pumps or fans in industrial, commercial, or residential settings, consume ultimately half of the electricity in the power grid. Stricter regulations concerning motor efficiencies are thus driving advances in motor designs and power electronics. In parallel with this unprecedented power-grid transformation, the transportation sector is undergoing rapid electrification. The commercial success of hybrid and all-electric vehicles, trains, marine vessels, drones, and next-generation aircraft hinges on continuous advancements in motor technology.

The design of electric machinery often begins with empirical rules of thumb or simple formulas that are derived after approximations. This approach is often adequate for obtaining “first-order” estimates of basic dimensions and parameters that come close to meeting nominal specifications. Nevertheless, novel machine designs and stringent performance requirements are challenging conventional wisdom and the validity of such methods. For more accuracy, industry practitioners rely on finite element analysis (FEA), which is the application of the FEM theory to a particular problem. Numerous sophisticated software packages have been developed that help us conduct FEA over multiple physical domains (e.g., electromagnetic, thermal, and mechanical). FEA is often embedded in an optimization loop that adjusts geometric and material parameters to maximize performance. However, FEA-based machine design is computationally demanding even with today’s technology, but advances in optimization algorithms and computing hardware are changing this landscape very rapidly.

This text explains the operation of electric machines under a new light, bridging electromagnetic fields, electric circuits, numerical analysis, and computer programming. We do not shy away from presenting fundamental concepts in a rigorous fashion. In doing so, we emphasize underlying physical modeling assumptions and limitations. And all this is tightly integrated with nitty-gritty details of implementing the FEM using a modern language (Python) for programming on today’s hardware. We have included examples covering several major classes of electric machines, which increase in complexity as new material is presented. From a simple linear-motion and constant-permeability electromagnet, we progress to devices with nonlinear ferromagnetic materials, and then introduce devices with rotational motion. We eventually reach the point where it becomes possible to analyze realistic permanent-magnet and wound-rotor synchronous machines, switched reluctance machines, and induction machines in the steady state and during transients. Each chapter includes a set of practice exercises, which are of a theoretical or programming nature.

Overview of Contents

The primary focus of this textbook is on the *electromagnetic* analysis of electric machines rather than their thermal or mechanical aspects. In **Chapter 1**, we recall relevant concepts from mathematics (vector calculus) and physics (classical electromagnetism). The reader may have encountered these during the course of an undergraduate degree. Nevertheless, they are reviewed to help progress smoothly throughout the material.

Chapter 1 also highlights basic Python syntax, which is the programming language that we use to implement the FEM.

We begin **Chapter 2** with a concise presentation of the calculus of variations. It has been our experience that this material is new to electrical engineering students. We prove that Maxwell's equations can be equivalently formulated as a variational problem, namely, the minimization of an energy-related functional, which inherently accounts for interface conditions due to material discontinuities. This approach, which forms the basis of the FEM, also provides useful physical insights into the principles of operation of electrical machinery. The second part of Chapter 2 is thus dedicated to an energy-oriented analysis of the magnetic field. These considerations naturally extend into theoretical results regarding the creation of electromagnetic force from the fields.

In **Chapter 3**, we introduce the FEM gently using elementary two-dimensional (2-D) problems involving Laplace and Poisson equations. We then adapt the method to the analysis of simple magnetic devices. We provide a thorough treatment of both magnetically linear and nonlinear problems, while emphasizing programming details. The chapter ends with further insights about Galerkin's method, setting forth the theoretical foundations of the FEM.

Practical FEA implementation guidelines for electric machines are offered next in **Chapter 4**. These include force and torque calculations, handling moving boundaries due to rotation without remeshing, and exploiting multi-pole symmetry by enforcing periodic boundary conditions. Detailed FEA examples are provided for wound-rotor synchronous machines, surface-mounted and interior permanent-magnet synchronous machines, and switched reluctance machines. These serve as a means to explain various important concepts, such as the magnetomotive force of distributed windings, the calculation of flux linkage and torque waveforms, Park's transformation in qd variables, and the derivation of equivalent circuits from FEA results.

Chapter 5 presents the analysis of time-varying fields with phasor-based FEA (under sinusoidal steady-state conditions) as well as time-stepping FEA (under arbitrary transients). This material begins with an overview of the physics of induced eddy currents. Finally, we provide details of coupling FEA models with external circuit equations, which is necessary in the analysis of squirrel-cage induction machines, transformer inrush currents, or the short-circuit transients of a synchronous generator.

The end-of-chapter problems (roughly 20 problems in each chapter) are a key pedagogical feature of our book and are tightly integrated with the material. The solutions manual accompanying the text provides answers to exercises that can be solved analytically, such as derivations and mathematical proofs. The programming exercises involve replicating results found in the main text; the reader is asked to write Python programs gluing together the various building blocks of code that are provided. We do not provide solutions to these exercises because there are many ways that such programs can be implemented, and we certainly do not wish to limit the creativity of the reader. Boxes are also sprinkled around the book, highlighting the most significant theoretical results and key equations.

Who Could Benefit by Reading this Book

This book is suitable for graduate-level students pursuing advanced engineering degrees with specialization in electric power systems, power electronics, or machines; and for practicing engineers who may be interested in understanding, analyzing, and designing electric motors, generators, and transformers. An undergraduate student at the senior level should have all the prerequisite knowledge to pick up this book, although our treatment is certainly more demanding than one typically encounters in introductory machine texts. The book contains a wealth of background information on vector calculus, variational calculus, and basic electromagnetism. Based on our experience with teaching this material, it is helpful to devote a few weeks at the beginning of a semester (or an independent study) to review these subjects before proceeding with the FEM and the analysis of electric machines.

The readers of this textbook will be fully capable of programming their own 2-D FEA program using Python, which is freely available and widely used by the scientific community. Apart from the personal satisfaction and sense of achievement gained from reaching this milestone demonstrating a solid grasp of the underlying concepts, such a tool could be useful for educational or research activities. Furthermore, a custom 2-D FEA program could feature the required flexibility for rapidly testing novel machine concepts that may be difficult to analyze using commercial software. This type of analysis (in contrast to a full-scale 3-D study) is lightweight and more than sufficient for obtaining accurate results over a broad range of applications. A reader will also gain a deeper understanding of what takes place “under the hood” of commercial FEA packages, and will be better informed regarding the limitations imposed by our imperfect modeling of the associated physical phenomena.

By trying to make the FEM more accessible, we aim to impart an appreciation for the numerical analysis of electromechanical devices, and perhaps to inspire further study of this subject. Our hope is that this book will serve for the years to come a new generation of engineers who are dedicated to changing our world for the better.

Nomenclature

This is a list of the main variables and mathematical operators encountered throughout this text. Ambiguity in notation may be resolved from context.

Greek-Letter Variables

α	Element basis function
α	Resistivity temperature coefficient
α	Scaling parameter of variation
$\alpha_p, \beta_p, \gamma_p, \delta_p$	Coefficients of p -pole-pair air-gap magnetic field
β	Generalized trapezoidal rule parameter
δ	Dirac delta function
δ	Skin depth
δ_{ij}	Kronecker delta
Δ	Area of triangle
ϵ	Permittivity
ζ	Radii ratio (inner over outer)
$\vec{\eta}, \eta$	Variation shape vector, scalar (test function)
θ	Angular or linear position
θ	Polar angle
$\hat{\theta}$	Polar angle unit vector
θ_r	Rotor angle
λ	Flux linkage of winding
μ	Permeability
μ_0	Permeability of free space
μ_h	Hysteretic permeability
μ_r	Relative permeability
ν	Reluctivity
ξ	Relaxation factor
ρ	Charge density
ρ	Resistivity
ρ_m	Magnetic charge density
σ	Conductivity
σ	Maxwell stress tensor
τ	Thickness

$\vec{\tau}, \tau$	Torque vector, scalar value
$\tilde{\tau}$	Torque per unit depth
ϕ	Azimuthal angle
$\hat{\phi}$	Hat function
$\hat{\phi}$	Azimuthal angle unit vector
ϕ_{pi}, ϕ_{po}	Phase angle of p -pole-pair air-gap magnetic potential at inner, outer boundary
φ	Scalar potential
Φ	Magnetic flux
$\tilde{\Phi}$	Magnetic flux phasor
χ_m	Magnetic susceptibility
ψ	Flux linkage of filament
ω	Electrical frequency
ω	Surface charge density
$\vec{\omega}, \omega$	Angular velocity vector, scalar value
ω_m	Rotor angular velocity (mechanical)
ω_p	Angular velocity of p -pole-pair field (mechanical)
Ω	Region in space (the domain of a vector or scalar field)

Latin-Letter Variables

A	Cross-sectional area
\mathbf{A}	Array of magnetic potential values
\vec{A}, A	Magnetic potential vector, magnitude
\tilde{A}, \tilde{A}	Magnetic potential complex vector, phasor
\tilde{A}, \tilde{A}	Modified magnetic potential function, array of values (for axisymmetric problems)
A_p	Function of p -pole-pair air-gap magnetic potential
A_{pi}, A_{po}	Amplitude of p -pole-pair air-gap magnetic potential at inner, outer boundary
\mathbf{b}	Source vector for linear FEA
\vec{B}, B	Magnetic flux density vector, magnitude
\tilde{B}, \tilde{B}	Magnetic flux density complex vector, phasor
$B_{pr}, B_{p\phi}$	Radial, tangential component of p -pole-pair air-gap B -field
B_{pri}, B_{pro}	Amplitude of radial component of p -pole-pair air-gap B -field at inner, outer boundary
D	Density
\mathcal{D}	Domain in 2-D or 3-D space
\vec{D}, D	Electric displacement field vector, magnitude
e	Error
\vec{E}, E	Electric field vector, magnitude
\tilde{E}, \tilde{E}	Electric field complex vector, phasor
f	Frequency
\vec{f}	Force density vector
F	Scalar field

$\vec{\mathbf{F}}$	Vector field
$\vec{\mathbf{F}}, F$	Force vector, magnitude
$\tilde{\mathbf{F}}$	Force per unit depth
\mathcal{F}	MMF vector
g	Acceleration of gravity
g	Air-gap width
\mathbf{g}	Gradient
h	Height
h	Time step
\mathbf{H}	Hessian matrix
$\vec{\mathbf{H}}, H$	Magnetic field vector, magnitude
$\tilde{\mathbf{H}}, \tilde{H}$	Magnetic field complex vector, phasor
\mathcal{H}^1	Hilbert space of functions with square integrable first derivatives
i or I	Current
\tilde{i} or \tilde{I}	Current phasor
$\hat{\mathbf{i}}$	x -Axis unit vector of a Cartesian coordinate system
I	Functional
I	Moment of inertia
I_ν	Modified Bessel function of the first kind of order ν
j	The imaginary unit, $j = \sqrt{-1}$
$\hat{\mathbf{j}}$	y -Axis unit vector of a Cartesian coordinate system
\mathbf{J}	Jacobian matrix
$\vec{\mathbf{J}}, J$	Current density vector, magnitude
$\vec{\mathbf{J}}_m, J_m$	Magnetization current density vector, magnitude
k_e	Coefficient of eddy current loss
k_h	Coefficient of hysteresis loss
k_{pf}	Packing factor
k_{st}	Stacking factor
$\hat{\mathbf{k}}$	z -Axis unit vector of a Cartesian coordinate system
\mathbf{K}	Reference frame transformation matrix
$\vec{\mathbf{K}}, K$	Surface current density vector, magnitude
$\vec{\mathbf{K}}_m, K_m$	Magnetization surface current density vector, magnitude
ℓ	Length
L	Inductance
L	Lagrangian
$\vec{\mathbf{L}}$	Angular momentum
\mathcal{L}^2	Space of square integrable functions
m	Energy density of magnetic field
m	Mass
$\vec{\mathbf{m}}$	Magnetic dipole moment
$\vec{\mathbf{M}}, M$	Magnetization vector, magnitude
n	Number of slots
$\hat{\mathbf{n}}$	Unit normal vector
N	Number of turns

p	Basis function coefficient
p	Number of pole-pairs
p	Volumetric power
\bar{p}	Time-averaged volumetric power
\vec{p}	Momentum per unit volume
P	Power
\mathcal{P}	Permeance
\vec{P}	Momentum
P_{mp}	Mechanical power of p -pole-pair field
$P_{r \rightarrow s}$	Air-gap power flow from rotor to stator
q	Basis function coefficient
\mathbf{q}	Array of basis function coefficients
q_e	Electric charge of an electron
Q	Electric charge
Q	Reactive power
r	Basis function coefficient
r	Radial distance
r	Residual
\mathbf{r}	Array of basis function coefficients
\mathbf{r}, \vec{r}	Displacement from the origin vector
$\hat{\mathbf{r}}$	Unit radial vector
$\dot{\mathbf{r}}$	Derivative of displacement vector \mathbf{r} with respect to time
r_i	Inner radius
r_o	Outer radius
R	Radius of a circle through the middle of the air gap
R	Resistance
\mathcal{R}	Reluctance
$\hat{\mathbf{s}}$	Unit tangential vector
s_p	Slip of p -pole-pair field
S	Surface
\mathbf{S}	Complex Poynting vector
\mathbf{S}	Stiffness matrix
\vec{S}	Poynting vector
t	Time
t	Step length
$\hat{\mathbf{t}}$	Unit tangential vector
\vec{t}	Maxwell stress
T	Kinetic energy
\mathbf{T}	Matrix multiplying vector of time derivatives of magnetic potentials
v, V	Voltage
\vec{v}, v	Velocity vector, magnitude
V	Potential energy
V	Region (volume) in 2-D or 3-D space
\tilde{V}	Voltage phasor

w	Solution guess
w	Width
w^*	True solution
W	Work or energy
W_{ag}	Energy stored in the air gap
W_c	Coenergy
W_f	Coupling field energy
X	Reactance
\mathcal{X}^N	N -dimensional trial space
Z	Impedance

Mathematical Operators

$\partial\Omega$	Boundary of region Ω
$\partial y/\partial x$	Partial derivative of y with respect to x
\dot{x}	Derivative of x with respect to time
$\ \vec{\mathbf{a}}\ $	Norm of vector
$\ w\ $	Norm of function
$\ w\ _E$	Energetic norm of function
\mathbf{A}^\top	Transpose of vector or matrix
\bar{f}	Time average of function
∇F	Gradient of scalar field
$\nabla \cdot \vec{\mathbf{F}}$	Divergence of vector field
$\nabla \times \vec{\mathbf{F}}$	Curl of vector field
$\nabla^2 F$	Laplacian of scalar field
$\nabla^2 \vec{\mathbf{F}}$	Laplacian of vector field
$\mathbf{a} \cdot \mathbf{b}$	Dot product
$\vec{\mathbf{a}} \times \vec{\mathbf{b}}$	Cross product
δx	Small change in x
δw	Variation of solution
dx	Differential of x
da	Differential surface area
$d\mathbf{a}$	Differential surface area vector
$d\mathbf{r}$	Differential displacement vector
ds	Differential arc length
dv	Differential volume
dy/dx	Total derivative of y with respect to x
(u, v)	Inner product of functions
$(u, v)_E$	Energetic inner product of functions

1 Review of Vector Calculus and Electromagnetic Fields

Μηδεὶς ἀγεωμέτρητος εἰσὶτω μοι τῇ θύρᾳ

“Let no one ignorant of geometry enter through this door”

Inscription on entrance of Plato’s Academy²

In this chapter, we will be highlighting fundamental concepts of vector calculus, which is the branch of mathematics concerned with the differentiation and integration of vector fields. In doing so, we have assumed that the reader is already familiar with the main concepts of algebra, vectors, mechanics, and classical electromagnetism, most likely in the course of satisfying the requirements of an undergraduate collegiate degree. The material is self-contained to the extent possible, but it is not intended as a rigorous treatment. We will not be attempting to explain ideas such as: What is a derivative and an integral? What is force? What is energy? What is electricity? What is the electromagnetic field? . . . Instead, we will emphasize the underlying mathematics (vector calculus) that enables us to *quantify* and *correlate* the physical variables in space.

Our exposition will progress from simple to more complicated ideas. To refresh our memory and support this material, we will be invoking many examples. Whenever possible, these will be based on electromagnetic field problems. The examples will also include relevant results that are necessary for a fields-based understanding and analysis of electric machines, while underpinning the mathematical formalism that we will adhere to for the remainder of this book.

As we set forth these wonderful concepts, we will not remain on a purely theoretical plane. Rather, we will immediately apply these ideas to perform relatively simple computational tasks using Python. These examples should be accessible by anyone with a basic knowledge of computer programming. We will thereby introduce Python (version 3) syntax and some key capabilities, and familiarize the reader with the programming language that we will use to code the finite element method (FEM) calculations. At the end of this chapter, we shall have a working knowledge of Python.

1.1 Overview of Maxwell's Equations

The equations that govern the electromagnetic field are known as **Maxwell's equations**.³ They involve two vector variables, namely the **electric field** \vec{E} and the **magnetic flux density** \vec{B} . The fields are created by the presence of a **charge density** ρ or a **current density** \vec{J} throughout space. In this first form, Maxwell's equations are applicable for problems in vacuum or free space.

Since we are primarily interested in the analysis of devices where the fields permeate matter, we need an alternate formulation of the equations, namely, a formulation in **macroscopic form**. In such form, the equations should capture the aggregate (i.e., space-averaged over many atoms) behavior of matter that is subject to an electromagnetic field. To this end, two additional fields are introduced. These are the **electric displacement field** \vec{D} and the **magnetic field** \vec{H} , which help us model the effect of electric and magnetic polarization, respectively. We also introduce phenomenological, **constitutive laws**, such as

$$\vec{B} = f(\vec{H}) \quad \text{and} \quad \vec{E} = g(\vec{D}), \quad (1.1)$$

to describe the relationships of fields within materials.

The macroscopic variant of Maxwell's equations comes in two equivalent forms, namely, differential and integral. In **differential form**, the equations are

$$\text{Gauss' }^4 \text{ law for the electric field: } \nabla \cdot \vec{D} = \rho, \quad (1.2a)$$

$$\text{Gauss' law for the magnetic field: } \nabla \cdot \vec{B} = 0, \quad (1.2b)$$

$$\text{Faraday's }^5 \text{ law: } \nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t}, \quad (1.2c)$$

$$\text{Ampère's }^6 \text{ law: } \nabla \times \vec{H} = \vec{J} + \frac{\partial \vec{D}}{\partial t}. \quad (1.2d)$$

In **integral form**, the equations are

$$\text{Gauss' law for the electric field: } \oiint_{\partial V} \vec{D} \cdot d\mathbf{a} = \iiint_V \rho \, dv, \quad (1.3a)$$

$$\text{Gauss' law for the magnetic field: } \oiint_{\partial V} \vec{B} \cdot d\mathbf{a} = 0, \quad (1.3b)$$

$$\text{Faraday's law: } \oint_{\partial S} \vec{E} \cdot d\mathbf{r} = - \iint_S \frac{\partial \vec{B}}{\partial t} \cdot d\mathbf{a}, \quad (1.3c)$$

$$\text{Ampère's law: } \oint_{\partial S} \vec{H} \cdot d\mathbf{r} = \iint_S \vec{J} \cdot d\mathbf{a} + \iint_S \frac{\partial \vec{D}}{\partial t} \cdot d\mathbf{a}. \quad (1.3d)$$

These are supplemented by an equation describing the **conservation of charge**:

$$\nabla \cdot \vec{J} = -\frac{\partial \rho}{\partial t}. \quad (1.4)$$

When the sources are constant or changing “slowly” with time, we obtain simpler

versions of Maxwell's equations by setting $\partial/\partial t = 0$:

$$\nabla \cdot \vec{\mathbf{D}} = \rho, \quad (1.5a)$$

$$\nabla \cdot \vec{\mathbf{B}} = 0, \quad (1.5b)$$

$$\nabla \times \vec{\mathbf{E}} = \mathbf{0}, \quad (1.5c)$$

$$\nabla \times \vec{\mathbf{H}} = \vec{\mathbf{J}}, \quad (1.5d)$$

$$\nabla \cdot \vec{\mathbf{J}} = 0. \quad (1.5e)$$

Clearly, this leads to a decoupling of the equations, so the electric and magnetic fields can be determined separately.

In practice, we encounter two types of problems, differentiated based on the main field source. If the field is created by a static or quasi-static distribution of charge, we obtain an **electrostatic problem**, where $\vec{\mathbf{B}} = \mathbf{0}$ or $\vec{\mathbf{B}} \approx \mathbf{0}$, and

$$\nabla \cdot \vec{\mathbf{D}} = \rho, \quad (1.6a)$$

$$\nabla \times \vec{\mathbf{E}} = \mathbf{0}. \quad (1.6b)$$

If the field is created by a static or quasi-static distribution of current, we obtain a **magnetostatic problem**, where $\vec{\mathbf{E}} = \mathbf{0}$ or $\vec{\mathbf{E}} \approx \mathbf{0}$, and

$$\nabla \cdot \vec{\mathbf{B}} = 0, \quad (1.7a)$$

$$\nabla \times \vec{\mathbf{H}} = \vec{\mathbf{J}}. \quad (1.7b)$$

All this may already seem overwhelming, but we hope that the material that follows will be helpful. In the subsequent sections, we will present background information on vector calculus that will enable us to comprehend what these formulas mean. We will start in §1.2 by defining vectors and mathematical operations on vectors, such as the dot and cross products. We will proceed in §1.3 by defining scalar and vector fields. In §1.4, we will define line, surface, and volume integrals, like the ones appearing in the integral form of Maxwell's equations. §1.5 will set forth differential operators on scalar and vector fields, whereas §1.6 will present various integral laws that are widely employed in vector calculus. We will conclude by putting a focus on the magnetic field in §1.7.

1.2 Vectors

Vector calculus entails the manipulation of **vectors** in three-dimensional (3-D) space, which are commonly used to describe physical quantities like force or velocity. These physical vectors are geometric objects that are associated with a magnitude and a direction. They are denoted using boldface notation and an arrow. For example, the magnetic flux density at a point (x, y, z) may be the vector

$$\vec{\mathbf{B}} = -0.5\hat{\mathbf{i}} + 1.2\hat{\mathbf{j}} + 0.5\hat{\mathbf{k}} \text{ T}, \quad (1.8)$$

where T stands for tesla,⁷ which is the SI⁸ unit for field density. Here, $\hat{\mathbf{i}}$, $\hat{\mathbf{j}}$, and $\hat{\mathbf{k}}$ are commonly used symbols for the three unit vectors of a Cartesian⁹ coordinate system. Unit vectors will be denoted by hats throughout this text. In Equation (1.8), the vector was written as a vector sum of three components. A different way to express a vector is by listing three scalar coefficients in a given coordinate system as a tuple, e.g.

$$\mathbf{B} = (-0.5, 1.2, 0.5), \quad (1.9)$$

or as a column vector

$$\mathbf{B} = \begin{bmatrix} -0.5 \\ 1.2 \\ 0.5 \end{bmatrix}. \quad (1.10)$$

Note that we did not use an arrow. The distinction is subtle, and often these two notations are used interchangeably. For instance, we may also write $\vec{\mathbf{B}} = (-0.5, 1.2, 0.5)$.

The **length** or **magnitude** or Euclidean¹⁰ **norm** of a vector $\vec{\mathbf{v}}$ will be denoted as $\|\vec{\mathbf{v}}\|$. The norm is calculated using the Pythagorean¹¹ theorem. For instance, the magnitude of the above B -vector is

$$\|\vec{\mathbf{B}}\| = \sqrt{(-0.5)^2 + 1.2^2 + 0.5^2} \approx 1.393 \text{ T}. \quad (1.11)$$

We may also define vectors in higher dimensions; however, these may not be representative of physical vector fields. These high-dimensional vectors will be denoted using a boldface symbol but without an arrow. For example, the solution of a linear system of equations with 100 unknowns, which is a column vector of dimension 100×1 , may be denoted as \mathbf{x} . Strictly speaking, placing an arrow on top of a physical vector is not necessary since a boldface symbol should suffice. Nevertheless, this notational distinction is made for added clarity because in this book we will encounter both physical and higher-dimensional vectors stemming from the numerical solution of Maxwell's equations. The Euclidean norm for such vectors is denoted similarly by $\|\mathbf{x}\|$. If \mathbf{x} has n components, then

$$\|\mathbf{x}\| = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}. \quad (1.12)$$

1.2.1 Dot Product

The **dot product** or **inner product** between two vectors $\mathbf{a} = (a_1, a_2, \dots, a_n)$ and $\mathbf{b} = (b_1, b_2, \dots, b_n)$ from the same n -dimensional space (in Cartesian coordinates) is defined algebraically as

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{a}^\top \mathbf{b} = \sum_{i=1}^n a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n. \quad (1.13)$$

The dot product returns a scalar, that is, $\mathbf{a} \cdot \mathbf{b} \in \mathbb{R}$. Note that the vector dimension n is arbitrary. Using this definition, we can see that

$$\mathbf{a} \cdot \mathbf{a} = \|\mathbf{a}\|^2. \quad (1.14)$$

An equivalent geometric definition is

$$\vec{\mathbf{a}} \cdot \vec{\mathbf{b}} = \|\vec{\mathbf{a}}\| \|\vec{\mathbf{b}}\| \cos \theta, \quad (1.15)$$

where θ is the angle between the two vectors, such that $0 \leq \theta \leq 180^\circ$ (see Figure 1.1). The latter definition is easier to visualize in two or three dimensions, hence the arrow notation. Note that

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{b} \cdot \mathbf{a}. \quad (1.16)$$

This is called the **commutative** property. Another important property of the dot product is related to orthogonality: *Two vectors are **orthogonal** if and only if their dot product is zero.*

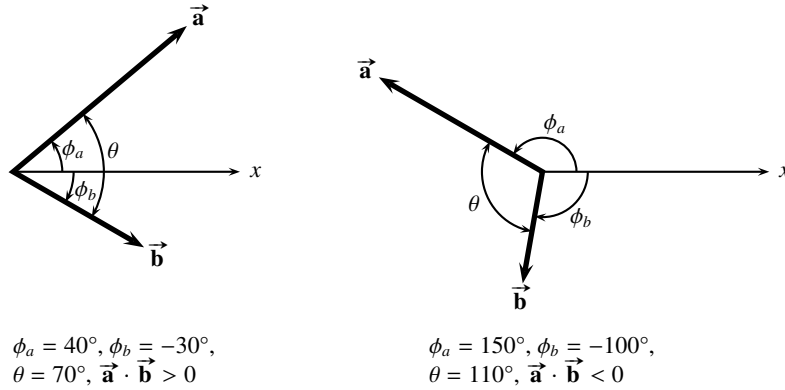


Figure 1.1 Dot product examples. Note that vector angles are positive when measured counterclockwise, and negative in the clockwise direction.

Example 1.1 *Equivalence of dot product definitions.*

Confirm that the algebraic and geometric definitions of the dot product are equivalent.

Proof For simplicity, we will solve this problem in two dimensions. (Even in 3-D space, without loss of generality we can always use a 2-D coordinate system that sits on the plane defined by the two vectors.) The trick is to express the vectors

$$\vec{\mathbf{a}} = a_1 \hat{\mathbf{i}} + a_2 \hat{\mathbf{j}}, \quad (1.17a)$$

$$\vec{\mathbf{b}} = b_1 \hat{\mathbf{i}} + b_2 \hat{\mathbf{j}} \quad (1.17b)$$

using their **polar coordinates**, that is, using their magnitudes and angles:

$$\mathbf{a} = (a_1, a_2) = (\|\vec{\mathbf{a}}\| \cos \phi_a, \|\vec{\mathbf{a}}\| \sin \phi_a), \quad (1.18a)$$

$$\mathbf{b} = (b_1, b_2) = (\|\vec{\mathbf{b}}\| \cos \phi_b, \|\vec{\mathbf{b}}\| \sin \phi_b). \quad (1.18b)$$

Let us limit the vector angles to the interval $(-180^\circ, 180^\circ]$. Without loss of generality,

we will assume that $\phi_a \geq \phi_b$, as in Figure 1.1. Hence

$$\vec{\mathbf{a}} \cdot \vec{\mathbf{b}} = a_1 b_1 + a_2 b_2 \quad (1.19a)$$

$$= \|\vec{\mathbf{a}}\| \|\vec{\mathbf{b}}\| (\cos \phi_a \cos \phi_b + \sin \phi_a \sin \phi_b) \quad (1.19b)$$

$$= \|\vec{\mathbf{a}}\| \|\vec{\mathbf{b}}\| \cos(\phi_a - \phi_b), \quad (1.19c)$$

where $\Delta\phi = \phi_a - \phi_b$ is in the interval $[0^\circ, 360^\circ)$. It suffices to show that $\cos \Delta\phi = \cos \theta$, where the angle between vectors $\theta \in [0^\circ, 180^\circ]$. There are two cases to consider: (i) if $0^\circ \leq \Delta\phi \leq 180^\circ$, then $\theta = \Delta\phi$; (ii) if $180^\circ < \Delta\phi < 360^\circ$, then $\theta = 360^\circ - \Delta\phi$. In both cases, we obtain $\cos \Delta\phi = \cos \theta$. \square

Example 1.2 *Invariance of dot product to axes rotation or reflection.*

Confirm that the algebraic definition of the dot product yields the same answer regardless of the choice of coordinate system.

Proof It is convenient to use 2-D Cartesian coordinate systems lying on the plane defined by the two vectors. We need to verify that the dot product remains the same in a **rotated** and possibly **reflected** coordinate system that maintains the length of the vectors. The more general case where the new coordinate system is such that all three components are present is left as an exercise for the reader. (The geometric dot product definition does not employ the coordinates of the vectors, implying that the result should indeed be independent of choice of axes. Nevertheless, this example is instructive because such coordinate system changes are fundamental to electric machine analysis.) In linear algebra, this is called a **change of basis** from one orthonormal basis to another, also termed an **orthogonal transformation**.

As depicted in Figure 1.2, a vector $\vec{\mathbf{a}}$ can be expressed in terms of the unit vectors of the two coordinate systems as

$$\vec{\mathbf{a}} = a_1 \hat{\mathbf{i}} + a_2 \hat{\mathbf{j}} = a'_1 \hat{\mathbf{i}}' + a'_2 \hat{\mathbf{j}}'. \quad (1.20)$$

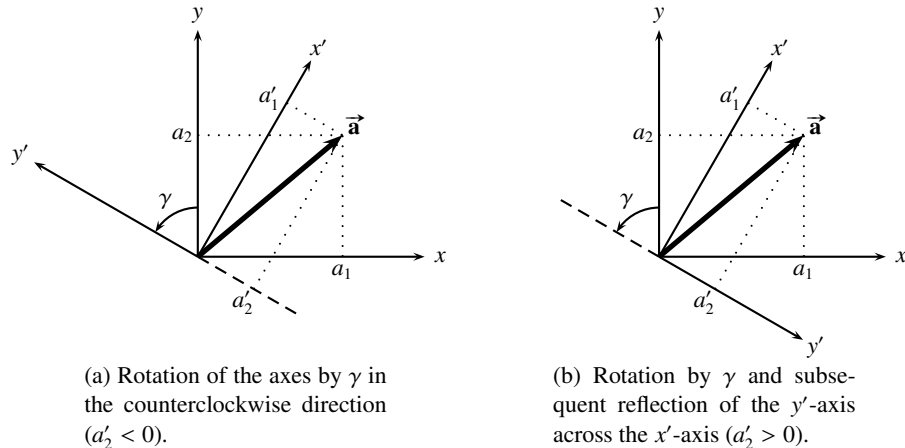


Figure 1.2 Cartesian coordinate system orthogonal transformation: from xy to $x'y'$.

By inspection, we can relate the unit vectors as

$$\hat{\mathbf{i}}' = \cos \gamma \hat{\mathbf{i}} + \sin \gamma \hat{\mathbf{j}}, \quad (1.21a)$$

$$\hat{\mathbf{j}}' = -\sin \gamma \hat{\mathbf{i}} + \cos \gamma \hat{\mathbf{j}} \quad (1.21b)$$

for a pure rotation, or

$$\hat{\mathbf{i}}' = \cos \gamma \hat{\mathbf{i}} + \sin \gamma \hat{\mathbf{j}}, \quad (1.22a)$$

$$\hat{\mathbf{j}}' = \sin \gamma \hat{\mathbf{i}} - \cos \gamma \hat{\mathbf{j}} \quad (1.22b)$$

for a rotation followed by a reflection of the y' -axis. Substitution of these equations in Equation (1.20) yields

$$\vec{\mathbf{a}} = a_1 \hat{\mathbf{i}} + a_2 \hat{\mathbf{j}} = (a'_1 \cos \gamma \mp a'_2 \sin \gamma) \hat{\mathbf{i}} + (a'_1 \sin \gamma \pm a'_2 \cos \gamma) \hat{\mathbf{j}}. \quad (1.23)$$

By equating coefficients, we obtain

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \cos \gamma & -\sin \gamma \\ \sin \gamma & \cos \gamma \end{bmatrix} \begin{bmatrix} a'_1 \\ a'_2 \end{bmatrix} \quad (1.24)$$

and

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \cos \gamma & \sin \gamma \\ \sin \gamma & -\cos \gamma \end{bmatrix} \begin{bmatrix} a'_1 \\ a'_2 \end{bmatrix}, \quad (1.25)$$

respectively, for the two cases. It is convenient to introduce a 2×2 **transformation matrix** \mathbf{M} that defines the transformation from the original to the new coordinates, which is found by taking the inverse of the above matrices. For case (a), which involves only a rotation, we have

$$\begin{bmatrix} a'_1 \\ a'_2 \end{bmatrix} = \underbrace{\begin{bmatrix} \cos \gamma & \sin \gamma \\ -\sin \gamma & \cos \gamma \end{bmatrix}}_{\mathbf{M}_a} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}, \quad (1.26)$$

whereas for case (b), which involves rotation and reflection, we have

$$\begin{bmatrix} a'_1 \\ a'_2 \end{bmatrix} = \underbrace{\begin{bmatrix} \cos \gamma & \sin \gamma \\ \sin \gamma & -\cos \gamma \end{bmatrix}}_{\mathbf{M}_b} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}. \quad (1.27)$$

The components of a second vector $\vec{\mathbf{b}}$ are transformed in the same manner. Hence, the dot product using the new coordinates becomes

$$\vec{\mathbf{a}} \cdot \vec{\mathbf{b}} = a'_1 b'_1 + a'_2 b'_2 = \begin{bmatrix} a'_1 & a'_2 \end{bmatrix} \begin{bmatrix} b'_1 \\ b'_2 \end{bmatrix} \quad (1.28a)$$

$$= \left(\begin{bmatrix} a_1 & a_2 \end{bmatrix} \mathbf{M}^\top \right) \left(\mathbf{M} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \right) \quad (1.28b)$$

$$= \begin{bmatrix} a_1 & a_2 \end{bmatrix} (\mathbf{M}^\top \mathbf{M}) \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}, \quad (1.28c)$$

where \mathbf{M}^\top is the **transpose** of \mathbf{M} . It can be readily verified that $\mathbf{M}^\top \mathbf{M} = \mathbb{I}$, the identity matrix, for both cases. Hence, the dot product is invariant to orthogonal transformations.

□

1.2.2 Cross Product

We will define the **cross product** or **vector product** between two vectors \vec{a} and \vec{b} in 3-D space. The cross product yields a vector that satisfies the following conditions: (i) it is perpendicular to both \vec{a} and \vec{b} ; (ii) its direction is decided by the right-hand rule; and (iii) its magnitude is equal to the area of the parallelogram defined by \vec{a} and \vec{b} as its sides, as illustrated in Figure 1.3. We can write, therefore

$$\vec{a} \times \vec{b} = \|\vec{a}\| \|\vec{b}\| \sin \theta \hat{n}. \quad (1.29)$$

Here, θ is the angle between the two vectors, so that $0 \leq \theta \leq 180^\circ$. Also, \hat{n} is a unit normal vector that is perpendicular to the plane defined by \vec{a} and \vec{b} , and points to a direction dictated by the right-hand rule. See Figure 1.4 for two examples. Note that

$$\vec{a} \times \vec{b} = -\vec{b} \times \vec{a}. \quad (1.30)$$

Hence, the cross product is **anti-commutative**. In contrast to the dot product, the magnitude of the cross product becomes zero when the vectors are parallel or antiparallel (i.e., when $\theta = 0$ or $\theta = 180^\circ$, respectively). Examples of formulas from physics that employ the cross product are the torque, $\vec{\tau} = \vec{r} \times \vec{F}$, and the Poynting¹² vector, $\vec{S} = \vec{E} \times \vec{H}$.

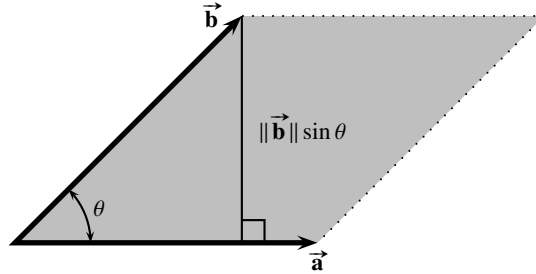


Figure 1.3 Geometric interpretation of the cross-product magnitude as the area of a parallelogram (base times height) formed by two vectors.

In Cartesian coordinates, the cross product can be evaluated as a determinant. If $\vec{a} = (a_1, a_2, a_3)$ and $\vec{b} = (b_1, b_2, b_3)$, then

$$\vec{a} \times \vec{b} = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix}. \quad (1.31)$$

Using a cofactor expansion along the top row, this determinant evaluates to

$$\vec{a} \times \vec{b} = \begin{vmatrix} a_2 & a_3 \\ b_2 & b_3 \end{vmatrix} \hat{i} - \begin{vmatrix} a_1 & a_3 \\ b_1 & b_3 \end{vmatrix} \hat{j} + \begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix} \hat{k} \quad (1.32a)$$

$$= (a_2 b_3 - a_3 b_2) \hat{i} + (a_3 b_1 - a_1 b_3) \hat{j} + (a_1 b_2 - a_2 b_1) \hat{k}. \quad (1.32b)$$

Strictly speaking, this is not a determinant of an actual matrix, but merely a mnemonic.

A special case of particular interest for us is the multiplication of vectors that sit on

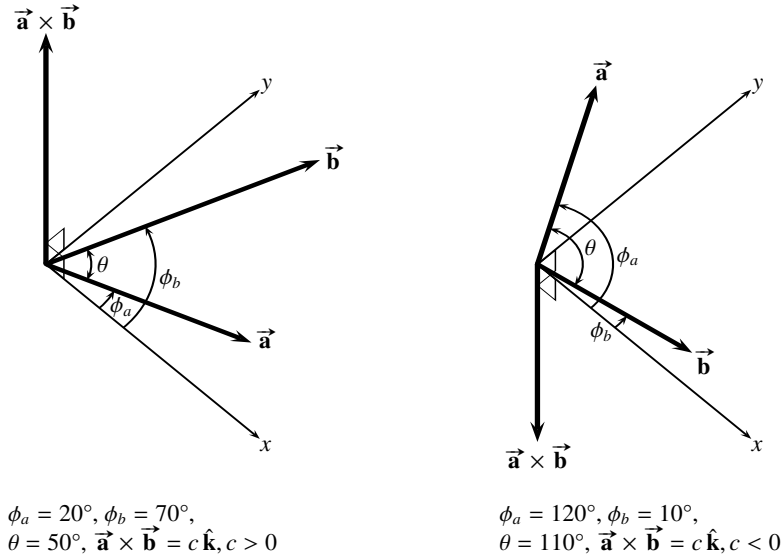


Figure 1.4 Cross-product examples. The vectors \vec{a} and \vec{b} are on the x - y plane.

the x - y plane, as shown in Figure 1.4. This means that $a_3 = b_3 = 0$, so the first two terms in the previous expression vanish, and we are left with

$$\vec{a} \times \vec{b} = (a_1 b_2 - a_2 b_1) \hat{k}. \quad (1.33)$$

Example 1.3 *Cross product as a determinant.*

Confirm the validity of the formal determinant expression (1.31).

Proof Note that the standard basis vectors are defined such that

$$\hat{i} \times \hat{j} = \hat{k}, \quad (1.34a)$$

$$\hat{j} \times \hat{k} = \hat{i}, \quad (1.34b)$$

$$\hat{k} \times \hat{i} = \hat{j}. \quad (1.34c)$$

This is the usual case for right-handed systems. Also, recall that the vector product is **distributive**, i.e.

$$\vec{a} \times (\vec{b} + \vec{c}) = \vec{a} \times \vec{b} + \vec{a} \times \vec{c}. \quad (1.35)$$

Evaluating the cross product yields

$$\vec{a} \times \vec{b} = (a_1 \hat{i} + a_2 \hat{j} + a_3 \hat{k}) \times (b_1 \hat{i} + b_2 \hat{j} + b_3 \hat{k}). \quad (1.36)$$

Multiplying out the terms will lead to a sum of nine cross products. Three of these terms will be zero, since $\hat{i} \times \hat{i} = \hat{j} \times \hat{j} = \hat{k} \times \hat{k} = \vec{0}$. The remaining six terms will contain cross products as in Equations (1.34a)–(1.34c), or in reverse order. Using the

anti-commutative property (1.30) will lead to Equation (1.32b), which is the expansion of the determinant in Equation (1.31). \square

Example 1.4 *The area of a triangle.*

Derive a formula for the area of a triangle based on the coordinates of its vertices.

Proof This result is invoked repeatedly in the FEM, so it is worthwhile taking a closer look. The proof hinges on the definition of the cross product. Consider the triangle ABC on the x - y plane, shown in Figure 1.5. The triangle vertices are located at (x_1, y_1) , (x_2, y_2) , and (x_3, y_3) . In the ensuing analysis, it is necessary to assume that the vertices are ordered *counterclockwise*. Hence, if we define the vectors \vec{c} and \vec{a} as the sides AB and BC of the triangle, respectively, then $\vec{c} \times \vec{a}$ will be pointing along the positive z -axis. (To see this, you may translate \vec{c} mentally so that its origin coincides with the origin of \vec{a} .) In turn, this implies that $\|\vec{c} \times \vec{a}\|$, which is the area of the parallelogram $ABCD$, equals the z -component of $\vec{c} \times \vec{a}$, as given by Equation (1.33). Denoting the triangle area by Δ , we have

$$\Delta = \frac{1}{2} \|\vec{c} \times \vec{a}\| = \frac{1}{2} (\vec{c} \times \vec{a}) \cdot \hat{k} \quad (1.37a)$$

$$= \frac{1}{2} (c_1 a_2 - c_2 a_1) \quad (1.37b)$$

$$= \frac{1}{2} [(x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1)] . \quad (1.37c)$$

What if we had chosen another pair of sides? For instance, take \vec{b} to be the side CA . Then, $\vec{b} \times \vec{c}$ points towards the positive z -axis as well, and

$$\Delta = \frac{1}{2} \|\vec{b} \times \vec{c}\| = \frac{1}{2} (\vec{b} \times \vec{c}) \cdot \hat{k} \quad (1.38a)$$

$$= \frac{1}{2} (b_1 c_2 - b_2 c_1) \quad (1.38b)$$

$$= \frac{1}{2} [(x_1 - x_3)(y_2 - y_1) - (y_1 - y_3)(x_2 - x_1)] . \quad (1.38c)$$

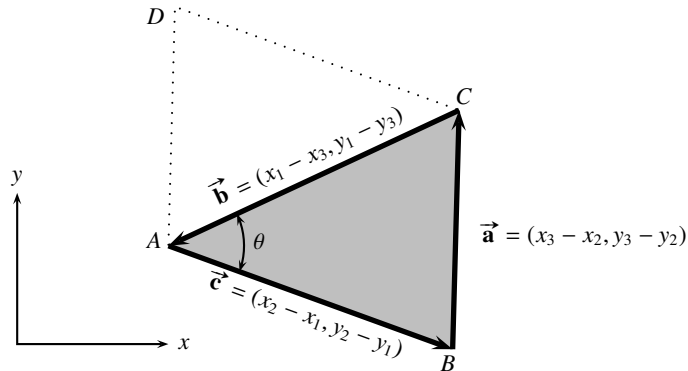


Figure 1.5 Calculating the area of a triangle on the x - y plane based on the cross product.

The last combination we could take is $\vec{\mathbf{a}} \times \vec{\mathbf{b}}$. This yields

$$\Delta = \frac{1}{2} \|\vec{\mathbf{a}} \times \vec{\mathbf{b}}\| = \frac{1}{2} (\vec{\mathbf{a}} \times \vec{\mathbf{b}}) \cdot \hat{\mathbf{k}} \quad (1.39a)$$

$$= \frac{1}{2} (a_1 b_2 - a_2 b_1) \quad (1.39b)$$

$$= \frac{1}{2} [(x_3 - x_2)(y_1 - y_3) - (y_3 - y_2)(x_1 - x_3)] . \quad (1.39c)$$

Note that these three formulas are similar in the sense that one leads to the other via cyclic permutation of the indices, i.e., replacing 1 by 2, 2 by 3, and 3 by 1. \square

1.2.3 Vectors in Python

We proceed with a few examples that illustrate the application of the basic vector concepts we have presented using the Python programming language.

Example 1.5 *Vector operations in Python.*

In Python, calculations with vectors can be performed efficiently using the NumPy library. NumPy functions can operate on n -dimensional arrays, or **ndarray** objects. We present a simple Python program that calculates the length of the B -field vector introduced in Equation (1.8):

```
1 import numpy as np

   # Define a vector as a NumPy array
   B = np.array([-0.5, 1.2, 0.5])
5 print('B =', B)

   # Calculate the norm of the vector
   normB = np.linalg.norm(B)
   print('||B|| =', normB)
```

Since this is our first encounter with a Python script, we note the following:

- To run this program, we may type each line successively in a Python interpreter, which can be started by typing `python` at the Unix shell or a Windows command prompt. Alternatively, these commands can be saved into a file that we can run by typing `python <filename.py>`.
- The line numbers are only shown for convenience when referring to certain parts of the code. They are not part of a Python program, unlike BASIC or FORTRAN.
- Single-line comments are preceded by a `#` character.

- NumPy commands are not loaded by default in Python. This is why we first **import** the NumPy library in line 1. All NumPy commands are then preceded by the **np** prefix, e.g., see the definition of the B -field in line 4. This naming convention is commonly followed in NumPy programs. In later examples, even though this command may not be always explicitly shown, you may rest assured that it is always there.
- The B -vector could have been defined more simply as $B = [-0.5, 1.2, 0.5]$. In Python this creates a **list** object, which is quite different from a NumPy ndarray. The program would still run without error, as NumPy would convert internally this list to an ndarray.
- Even though this is not shown in the program, it is noteworthy that *Python lists and arrays are indexed starting at zero*. To access the first element in the vector, we would type $B[0]$, for the second element, we would type $B[1]$, and so on. We can also access the last element as $B[-1]$, the second to last as $B[-2]$, etc. Note the use of brackets (rather than parentheses).
- In line 8, the **norm** command is defined inside the linear algebra (**linalg**) package of NumPy, so it is accessed by typing **np.linalg.norm**.
- The output of the script, which is generated by the two **print** commands, is

```
B = [-0.5  1.2  0.5]
||B|| = 1.39283882772
```

Without these commands, nothing would appear on the screen. In a Matlab program, the absence of a semicolon after a variable definition leads to the variable being displayed in the command window. However, as you can see from this example, semicolons are not used in Python to terminate commands.

Example 1.6 Dot product using NumPy.

Suppose we want to calculate the inner product of two vectors, $\mathbf{a} = (1, 2, 3, 4, 5)$ and $\mathbf{b} = (6, 7, 8, 9, 10)$. This can be achieved using the NumPy **dot** command. By running the following program, the reader can verify that $c = \mathbf{a} \cdot \mathbf{b} = 130$.

```
a = np.array([1, 2, 3, 4, 5])
b = np.array([6, 7, 8, 9, 10])
c = np.dot(a, b)
```

NumPy also provides a **cross** command, which can be invoked to evaluate cross products of vectors in \mathbb{R}^2 or \mathbb{R}^3 .

Example 1.7 Element-wise operations on vectors in Python.

When programming the FEM, it is often required to operate element-wise on high-dimensional vectors. To this end, we may use various built-in NumPy functions, such as **sin** and **sqrt** (i.e., square root). For instance, suppose we have stored the x and y -components of a 2-D B -field at some points in our device in two vectors. Here is a Python program that calculates the magnitude of the B -field at each point. Arbitrary values have been set for the B -field at three points.

```
Bx = np.array([ 1, 0, 3])
By = np.array([-1, 2, 4])
normB = np.sqrt(np.square(Bx) + np.square(By))
print('||B|| = ', normB)
```

The program returns

```
||B|| = [ 1.41421356  2.          5.          ]
```

The reader may readily verify that the mathematical operations were performed element-wise.

1.2.4 Triple Products

The following triple-product identities are useful for manipulating vector fields.

The **scalar triple product** involves the dot and cross products of three vectors, in the sense $\vec{a} \cdot (\vec{b} \times \vec{c})$. Using elementary geometry, we can show that

$$\vec{a} \cdot (\vec{b} \times \vec{c}) = \pm(\text{volume of parallelepiped}), \quad (1.40)$$

where the parallelepiped is the one defined by the three vectors as 3 (of its 12) edges, as shown in Figure 1.6. The parentheses may be dropped since there can be no ambiguity in the order of these operations, and we could write instead $\vec{a} \cdot \vec{b} \times \vec{c}$. (We cannot dot-multiply \vec{a} and \vec{b} first because this yields a scalar that cannot be subsequently cross-multiplied with \vec{c} .) It is easy to show that the scalar triple product is unaffected by circular permutation of its arguments:

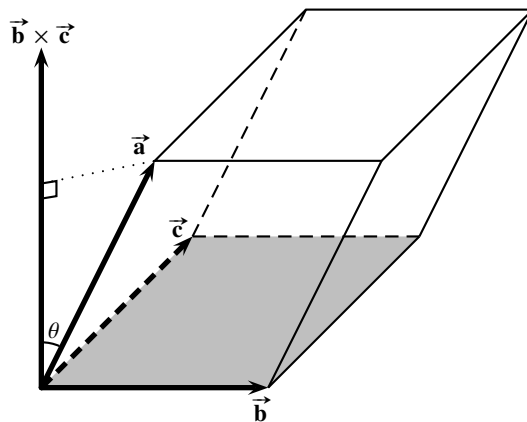
$$\vec{a} \cdot (\vec{b} \times \vec{c}) = \vec{b} \cdot (\vec{c} \times \vec{a}) = \vec{c} \cdot (\vec{a} \times \vec{b}). \quad (1.41)$$

This is left as an exercise for the reader.

The **vector triple product** is given by $\vec{a} \times (\vec{b} \times \vec{c})$. It can be shown that

$$\vec{a} \times (\vec{b} \times \vec{c}) = (\vec{a} \cdot \vec{c})\vec{b} - (\vec{a} \cdot \vec{b})\vec{c}. \quad (1.42)$$

First, we evaluate $\vec{b} \times \vec{c}$, which is a vector perpendicular to both \vec{b} and \vec{c} . Let us



The parallelepiped volume is base times height. The gray-shaded base area is $\|\vec{b} \times \vec{c}\|$. The height is $\|\vec{a}\| \cdot |\cos \theta|$, where θ is the angle between \vec{a} and $(\vec{b} \times \vec{c})$, which could be greater than 90° (for instance, this would happen if \vec{b} and \vec{c} were reversed).

Figure 1.6 The scalar triple product as the volume of a parallelepiped.

denote this normal direction by $\hat{\mathbf{n}}$. Then, we cross multiply by $\vec{\mathbf{a}}$, which returns a vector that is perpendicular to $\hat{\mathbf{n}}$, i.e., it is on the plane defined by $\vec{\mathbf{b}}$ and $\vec{\mathbf{c}}$. Hence, the formula makes sense: it expresses the triple product as a linear combination of $\vec{\mathbf{b}}$ and $\vec{\mathbf{c}}$. A proof is provided in Example 1.10.

Example 1.8 *Scalar triple product as a determinant.*

Let $\vec{\mathbf{a}} = (a_1, a_2, a_3)$, $\vec{\mathbf{b}} = (b_1, b_2, b_3)$, $\vec{\mathbf{c}} = (c_1, c_2, c_3)$. Show that

$$\vec{\mathbf{a}} \cdot (\vec{\mathbf{b}} \times \vec{\mathbf{c}}) = \begin{vmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{vmatrix}. \quad (1.43)$$

Proof Using the expansion of Equation (1.32b), we have

$$\begin{aligned} \vec{\mathbf{a}} \cdot (\vec{\mathbf{b}} \times \vec{\mathbf{c}}) &= (a_1 \hat{\mathbf{i}} + a_2 \hat{\mathbf{j}} + a_3 \hat{\mathbf{k}}) \\ &\quad \cdot [(b_2 c_3 - b_3 c_2) \hat{\mathbf{i}} + (b_3 c_1 - b_1 c_3) \hat{\mathbf{j}} + (b_1 c_2 - b_2 c_1) \hat{\mathbf{k}}] \end{aligned} \quad (1.44a)$$

$$= a_1(b_2 c_3 - b_3 c_2) + a_2(b_3 c_1 - b_1 c_3) + a_3(b_1 c_2 - b_2 c_1). \quad (1.44b)$$

By inspection, we can confirm that this is the cofactor expansion of the determinant along its first row. \square

Example 1.9 *Mechanical power from rotational motion.*

Suppose a force $\vec{\mathbf{F}}$ acts on a mass that is displaced by $\vec{\mathbf{r}}$ from the origin. The mass moves with velocity $\vec{\mathbf{v}} = d\vec{\mathbf{r}}/dt$. Find an expression for the mechanical power under rotational motion.

Solution

From physics, we know that the mechanical power exerted by the force is the dot product between force and velocity:

$$P_m = \frac{dW_m}{dt} = \vec{\mathbf{F}} \cdot \vec{\mathbf{v}}. \quad (1.45)$$

In other words, within a small time interval δt , where the body moves by $\delta \vec{\mathbf{r}} = \vec{\mathbf{v}} \delta t$, the force has done work equal to

$$\delta W_m = \vec{\mathbf{F}} \cdot \delta \vec{\mathbf{r}}. \quad (1.46)$$

Now suppose that the mass is rotating around an axis that passes through the origin so that its speed is given by

$$\vec{\mathbf{v}} = \vec{\omega} \times \vec{\mathbf{r}}, \quad (1.47)$$

where $\vec{\omega} = \omega \hat{\mathbf{a}}$ (rad/s) is an angular velocity vector. Here, $\hat{\mathbf{a}}$ is a unit vector that defines the axis of rotation. Hence, the mechanical power can be expressed as the scalar triple product

$$P_m = \vec{\mathbf{F}} \cdot (\vec{\omega} \times \vec{\mathbf{r}}). \quad (1.48)$$

Invoking the identity (1.41), we have

$$P_m = \vec{\omega} \cdot (\vec{\mathbf{r}} \times \vec{\mathbf{F}}), \quad (1.49)$$

or

$$P_m = \vec{\omega} \cdot \vec{\boldsymbol{\tau}}, \quad (1.50)$$

where $\vec{\boldsymbol{\tau}} = \vec{\mathbf{r}} \times \vec{\mathbf{F}}$ is the torque.

Example 1.10 *Vector triple-product expansion.*

Prove the identity (1.42).

Proof We have

$$\vec{\mathbf{a}} \times (\vec{\mathbf{b}} \times \vec{\mathbf{c}}) = \begin{vmatrix} \hat{\mathbf{i}} & \hat{\mathbf{j}} & \hat{\mathbf{k}} \\ a_1 & a_2 & a_3 \\ (b_2c_3 - b_3c_2) & (b_3c_1 - b_1c_3) & (b_1c_2 - b_2c_1) \end{vmatrix}. \quad (1.51)$$

Let us work with the $\hat{\mathbf{i}}$ -component of the product:

$$a_2(b_1c_2 - b_2c_1) - a_3(b_3c_1 - b_1c_3) = (a_2c_2 + a_3c_3)b_1 - (a_2b_2 + a_3b_3)c_1. \quad (1.52)$$

To bring it to the required form, we add and subtract $a_1b_1c_1$, which is incorporated into the existing b_1 and c_1 terms:

$$\begin{aligned} a_2(b_1c_2 - b_2c_1) - a_3(b_3c_1 - b_1c_3) &= (a_1c_1 + a_2c_2 + a_3c_3)b_1 \\ &\quad - (a_1b_1 + a_2b_2 + a_3b_3)c_1 \end{aligned} \quad (1.53a)$$

$$= (\vec{\mathbf{a}} \cdot \vec{\mathbf{c}})b_1 - (\vec{\mathbf{a}} \cdot \vec{\mathbf{b}})c_1. \quad (1.53b)$$

Manipulating in a similar manner the other two components, and then combining all three terms to form a vector, yields the desired result. \square

1.3 Scalar and Vector Fields

We use the concept of a field to describe quantities throughout space, e.g., the effects of electric or magnetic sources. A **scalar field** is a function that assigns a real number value at each point of a domain Ω , where this could be the whole space or only a subset. On the other hand, a **vector field** is a function that assigns a vector at each point in Ω . Primarily, we are interested in fields in 3-D space that describe physical quantities of interest, such as magnetic fields.

In a more formal manner, we denote a scalar field as $F : \Omega \subseteq \mathbb{R}^3 \rightarrow \mathbb{R}$, whereas a vector field is denoted as $\vec{\mathbf{F}} : \Omega \subseteq \mathbb{R}^3 \rightarrow \mathbb{R}^3$. If \mathbf{r} is a vector from the origin to an arbitrary point in Ω , we shall write $F(\mathbf{r})$ or $\vec{\mathbf{F}}(\mathbf{r})$ to obtain the scalar or vector field function value, respectively.

The vector \mathbf{r} is still a vector in 3-D space, so we could have denoted it by $\vec{\mathbf{r}}$, and the vector field could be denoted as $\vec{\mathbf{F}}(\vec{\mathbf{r}})$. Invoking our poetic license, we will often choose

to drop the arrow from $\vec{\mathbf{r}}$ so as to avoid cumbersome notation, especially when dealing with function arguments or differentials, such as the differential displacement $d\mathbf{r}$. However, arrows will always be used on top of electromagnetic vector fields or other physical quantities, such as the flux density $\vec{\mathbf{B}}$ or the force $\vec{\mathbf{F}}$.

Example 1.11 *Magnetic field of infinitely long, straight wire (part a: external field).*

Determine the magnetic field that is created by an infinitely long, straight, and thin wire in free space that carries a constant current i , measured in A (amperes). Of course, infinitely long wires do not exist in practice, but they are useful from a pedagogical perspective. (This could also be a reasonable approximation of the field that exists very close to a wire, even if the wire is curved.)

Solution

From physics, recall that the magnetic flux density in this case is given by

$$B(r) = \frac{\mu_0 i}{2\pi r}, \quad (1.54)$$

where the physical constant

$$\mu_0 = 4\pi 10^{-7} \text{ H/m (henries}^{13} \text{ per meter)} \quad (1.55)$$

is the **magnetic permeability of free space**, and r is the distance (in m) from an arbitrary point of interest to the wire. In SI units, the B -field is measured in T (teslas). Another commonly encountered unit for the B -field is the gauss: 1 T = 10,000 gauss.

The above formula yields the magnitude of the vector field $\vec{\mathbf{B}} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ at any point in space (except on the wire itself) as a function of r . Clearly, to complete the definition of the vector field, we also need to define the direction of the vectors, which we will do descriptively (rather than algebraically): Each vector is perpendicular to both the wire and to the shortest line segment between the point of interest and the wire (whose length equals the distance r); its direction is determined by the right-hand rule.

Example 1.12 *Vector field visualization using Python.*

We can visualize vector fields using the built-in plotting capabilities of Python. For instance, let us consider the electric field in free space created by two point charges positioned at $\mathbf{r}_1 = (-0.5, 0, 0)$ and $\mathbf{r}_2 = (0.5, 0, 0)$, respectively, in m. The charge on the left is $Q_1 = +2 \text{ C (coulomb}^{14})$, and the charge on the right is $Q_2 = -1 \text{ C}$.

The total electric field at an arbitrary point in space (displaced by \mathbf{r} from the origin) is obtained by superposition:

$$\vec{\mathbf{E}}(\mathbf{r}) = \frac{1}{4\pi\epsilon_0} \left(Q_1 \frac{\mathbf{r} - \mathbf{r}_1}{\|\mathbf{r} - \mathbf{r}_1\|^3} + Q_2 \frac{\mathbf{r} - \mathbf{r}_2}{\|\mathbf{r} - \mathbf{r}_2\|^3} \right), \quad (1.56)$$

where the physical constant

$$\epsilon_0 \approx 8.854 \cdot 10^{-12} \text{ F/m (farads per meter)} \quad (1.57)$$

is the **permittivity of free space**. The E -field is measured in V/m (volts¹⁵ per meter).

Below is a Python program that plots the vector field on the plane $z = 0$, where all vectors are horizontal (i.e., $E_z = 0$). Its output is shown in Figure 1.7, where two kinds of plots are illustrated, namely, a **quiver** plot and a **streamline** plot. The quiver plot consists of vectors at discrete user-specified points throughout the domain of interest. The streamline plot function uses the numerical data to produce a set of smooth lines, which have the property of being always tangent to the field. The streamlines originate from points that are selected internally by the plotting algorithm.

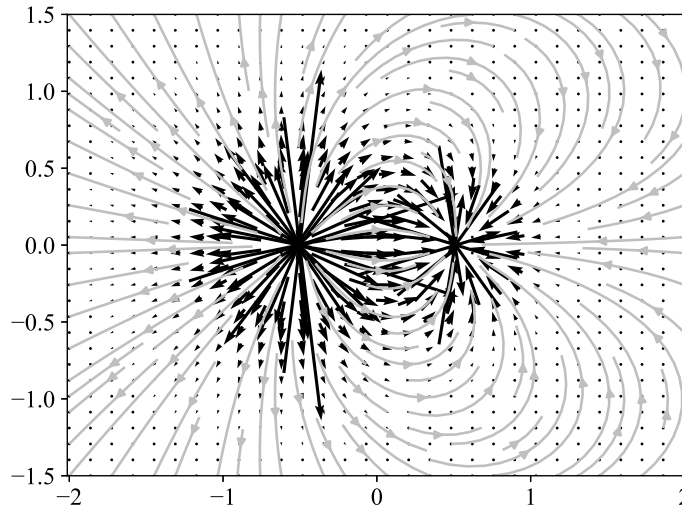


Figure 1.7 The electric field of two point charges. This figure was created by a superposition of a quiver and a streamline plot.

```

1  import numpy as np
   import matplotlib.pyplot as plt

   # Problem parameters
5  Q1 = 2
   Q2 = -1
   r1 = np.array([-0.5, 0, 0])
   r2 = np.array([ 0.5, 0, 0])
   z = 0 # we plot the projection of the E-field on this plane

10 # Calculate the electric field
   nx = ny = 30
   xv = np.linspace(-2, 2, nx)
   yv = np.linspace(-1.5, 1.5, ny)
15 Ex = np.zeros((yv.size,xv.size))
   Ey = np.zeros((yv.size,xv.size))

   for i in range(xv.size):
       x = xv[i]
20       for j in range(yv.size):
           y = yv[j]
           r = np.array([x, y, z])
           d1 = np.linalg.norm(r-r1)

```

```

    d2 = np.linalg.norm(r-r2)
25
    # We will not plot the field too close to the charges
    # because the quiver plot will be dominated by these
    # arrows. We define a minimum distance to exclude points
    # within a sphere of radius dmin around the charges.
30    dmin = 0.15
    if d1 < dmin or d2 < dmin:
        E1 = [0, 0, 0]
        E2 = [0, 0, 0]
    else:
35        E1 = Q1*(r-r1)/d1**3
        E2 = Q2*(r-r2)/d2**3
        E = E1 + E2
        Ex[j,i] = E[0]
        Ey[j,i] = E[1]
40
    # Make the grid
    xx, yy = np.meshgrid(xv, yv)

    # Plot
45    fig = plt.figure()
    ax = fig.gca()
    ax.streamplot(xx, yy, Ex, Ey, color='0.75')
    ax.quiver(xx, yy, Ex, Ey, width=.005, pivot = 'mid')
    plt.axis('equal')
50    plt.show()

```

Let us dissect the script:

- NumPy is imported in line 1. Matplotlib Pyplot, which is a Matlab-like plotting library, is imported in line 2.
- Lines 5–9 contain self-explanatory problem-specific parameter definitions.
- Lines 12–16 contain array definitions. In Python, the chained assignment of line 12 is permitted, and saves us a bit of time defining the size of the arrays `xv` and `yv`. The arrays contain evenly spaced points created by a NumPy `linspace` command. (Here, the streamline plot function requires the grid points to be evenly spaced.) Note that it would have been incorrect to define `Ey = Ex` in line 16. In Python, this would create a single object in memory (at line 15), which both parameters would refer to, and would lead to a bug in the code.
- The double **for** loop in lines 18–39 calculates and stores a normalized electric field (we are ignoring the $4\pi\epsilon_0$ term since it does not affect this plot). In Python, the **range**(`x`) command creates a list of integers starting at zero and ending at `x-1`. Note the absence of **end** keywords to delimit the **for** loops. We need, however, to ensure proper indentation (typically, each level is indented by 4 spaces), otherwise the Python interpreter will complain.
- It is important to note the inverse order of indexing the electric field arrays: they are first indexed by `y` and then by `x`. This is how they were defined in lines 15 and 16 as well. This syntax is due to how the `meshgrid` function (line 42) works by default (similar to Matlab, although NumPy allows one to change this back to “normal” with an optional argument).

- The magnitude of \vec{E} close to the charges grows very fast and would dominate the plot (due to a $1/d^2$ term, d denoting distance); hence, we decide not to plot vectors in the vicinity of charges (see lines 26–33).
- Finally, plotting takes place in lines 44–50. First, a figure is created, and then a handle to its axes is obtained. The `quiver` and `streamplot` functions have similar syntax. The `xy` axes are drawn with equal scaling in line 49. The `show` command displays the plot.

For more information on the various differences between NumPy and Matlab, the reader is advised to consult online resources on this topic.

Example 1.13 *Scalar field visualization using Python.*

Let us plot contours (also called **isolines** or **level sets**) of the electrostatic potential based on the charges in the previous example. The potential is a scalar field measured in V , and is given by

$$\varphi(\mathbf{r}) = \frac{1}{4\pi\epsilon_0} \left(Q_1 \frac{1}{\|\mathbf{r} - \mathbf{r}_1\|} + Q_2 \frac{1}{\|\mathbf{r} - \mathbf{r}_2\|} \right). \quad (1.58)$$

The following Python code snippet illustrates the use of the built-in Matplotlib Pyplot contour function. The result is shown in Figure 1.8.

```
1 # Define contour values
V = [-12, -6, -2, -1, -0.5, -0.25, 0, \
      0.25, 0.5, 0.75, 1, 1.5, 2, 6, 12]
5 # Plot
fig = plt.figure()
```

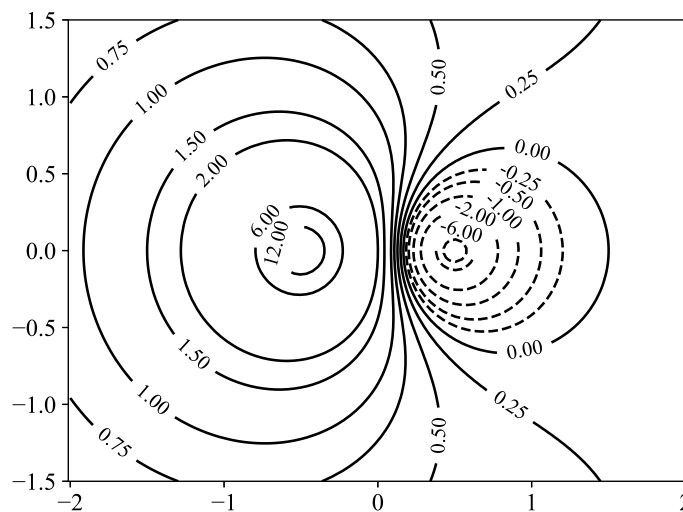


Figure 1.8 The electric potential of two point charges, generated by a contour plot. The potential values are normalized; the true potential (in V) is obtained after division by $4\pi\epsilon_0$.

```

ax = fig.gca()
CS = ax.contour(xx, yy, Phi, V, colors='k')
plt.axis('equal')
10 plt.clabel(CS, inline=1, fontsize=12, fmt = '%.2f', manual=True)
plt.show()

```

The code for computing the potential is not shown because it is similar to the previous example, and is performed over a regularly spaced grid. In line 2, we are defining which contour lines to plot. Note that the contours are not evenly distributed in this case. The contours look smooth because a relatively fine grid has been used. Also observe that negative potential values correspond to the dashed lines.

1.4 Integrals of Scalar and Vector Fields

We present several types of scalar and vector field integrals that we will encounter in this text, which are fundamental in understanding the properties of the magnetic field.

1.4.1 Line Integrals

The **line integral of a scalar field** $F : \Omega \rightarrow \mathbb{R}$ is defined over a piecewise smooth curve $C \subset \Omega$ (i.e., some curve within the space where the field exists) as

$$\int_C F(\mathbf{r}) ds, \quad (1.59)$$

where ds represents a differential (infinitesimal) arc length along the curve. Note that the result does *not* depend on the direction of travel. For example, if $\rho(\mathbf{r})$ represents the density per unit length (kg/m) of a thin wire (which is not necessarily constant), its mass can be found as $\int_C \rho(\mathbf{r}) ds$.

On the other hand, the **line integral of a vector field** $\vec{\mathbf{F}} : \Omega \rightarrow \mathbb{R}^3$ is defined over a piecewise smooth curve $C \subset \Omega$ as

$$\int_C \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{r} = \int_C \vec{\mathbf{F}}(\mathbf{r}) \cdot \hat{\mathbf{t}} dr, \quad (1.60)$$

where $d\mathbf{r} = dr \hat{\mathbf{t}}$ is a differential displacement vector that points in the direction that we traverse the path, $\hat{\mathbf{t}}$ being a unit tangent vector. Note that the line integral is based on the dot product, so it accounts only for the component of $\vec{\mathbf{F}}$ that is tangent to the integration path. In a sense, we are measuring the degree to which $\vec{\mathbf{F}}$ is aligned with the integration path. As a result, the answer is a scalar. Also, the answer depends on the direction that we travel on the curve C ; the result changes sign if we move in the opposite direction (since $d\mathbf{r}$ will become $-d\mathbf{r}$). Examples from physics are the work of a force, $W = \int_C \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{r}$, and the electric field potential, $\varphi = - \int_C \vec{\mathbf{E}}(\mathbf{r}) \cdot d\mathbf{r}$.

A special case is when the path C is *closed*, that is, when it starts and ends at the same point. We call this kind of line integral a **circulation** of the vector field, and we denote

it as

$$\oint_C \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{r}. \quad (1.61)$$

Furthermore, if this happens to be zero for any path, we call the field **conservative**.

In general, to evaluate a line integral, we need to define a bijective parametrization of the curve, $\mathbf{r} : [a, b] \rightarrow C$, such that $\mathbf{r}(a)$ and $\mathbf{r}(b)$ are the two endpoints of C . The line integral of a scalar field is then

$$\int_C F(\mathbf{r}) ds = \int_a^b F(\mathbf{r}(t)) \|\mathbf{r}'(t)\| dt, \quad (1.62)$$

whereas the line integral of a vector field is

$$\int_C \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{r} = \int_a^b \vec{\mathbf{F}}(\mathbf{r}(t)) \cdot \mathbf{r}'(t) dt. \quad (1.63)$$

Without getting too technical, we remind the reader that the line integral can be interpreted as a Riemann¹⁶ sum, i.e., by splitting the curve into a finite number of segments, and then letting their width approach zero. The displacement between two adjacent points on the curve is

$$\delta\mathbf{r}(t) = \mathbf{r}(t + \delta t) - \mathbf{r}(t) \approx \mathbf{r}'(t) \delta t, \quad (1.64)$$

where we used a first-order approximation based on a Taylor¹⁷ expansion. This also implies that the derivative $\mathbf{r}'(t)$ should be a vector that is tangent to the curve. The incremental arc length is

$$\delta s = \|\mathbf{r}'(t)\| \delta t \quad (\text{for } \delta t > 0). \quad (1.65)$$

At the limit, the deltas become differentials (e.g., δs becomes ds), and the Riemann sums become the integrals (1.62) and (1.63).

An alternative notation for the line integral of a vector field, which is obtained by taking the dot product between $\vec{\mathbf{F}} = (F_x, F_y, F_z)$ and $d\mathbf{r} = (dx, dy, dz)$, is

$$\int_C \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{r} = \int_C F_x dx + F_y dy + F_z dz. \quad (1.66)$$

The reader is cautioned that this should not be evaluated as three separate integrals. Rather, we should express the differential lengths based on the parametrization of the curve; for instance, we would substitute $dx = x'(t) dt$, etc.

If the curve is not smooth, that is, if there is a point, say $\mathbf{r}(c)$, where the curve changes direction abruptly, then the tangent directions right before and right after $\mathbf{r}(c)$ will be different. In this case, the line integral should be evaluated in two parts, as $\int_a^b = \int_a^c + \int_c^b$. This process may have to be repeated if the curve has a finite number of such discontinuities.

Example 1.14 *Line integral calculation.*

Consider the vector field

$$\vec{\mathbf{F}}(x, y) = -\operatorname{sgn}(y)F\hat{\mathbf{i}}, \quad (1.67)$$

where $F > 0$ is a constant. The field reverses direction across the x -axis. This situation is illustrated in Figure 1.9. Calculate the line integral of $\vec{\mathbf{F}}$ along the semicircular path C starting at -90° and ending at 90° in the counterclockwise direction.

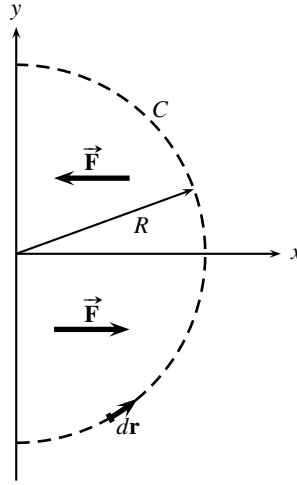


Figure 1.9 The geometry for the line integral of Example 1.14.

Solution

Note that $\vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{r} \geq 0$ throughout the path C , so we expect the answer to be positive. Let R be the radius of the semicircle. The integration path parametrization is

$$\mathbf{r}(t) = (x(t), y(t)) = (R \cos t, R \sin t), \quad (1.68)$$

with parameter $t \in [-\pi/2, \pi/2]$ representing the angle with respect to the x -axis. To integrate, we need the derivative

$$\mathbf{r}'(t) = (-R \sin t, R \cos t). \quad (1.69)$$

Hence

$$\int_C \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{r} = \int_{-\pi/2}^{\pi/2} [-\operatorname{sgn}(y(t))F\hat{\mathbf{i}}] \cdot (-R \sin t \hat{\mathbf{i}} + R \cos t \hat{\mathbf{j}}) dt \quad (1.70a)$$

$$= -FR \int_{-\pi/2}^0 \sin t dt + FR \int_0^{\pi/2} \sin t dt \quad (1.70b)$$

$$= 2FR. \quad (1.70c)$$

1.4.2 Surface Integrals

The **surface integral of a scalar field** $F : \Omega \rightarrow \mathbb{R}$ is defined on a surface $S \subset \Omega$ as the double integral

$$\iint_S F(\mathbf{r}) \, da. \quad (1.71)$$

To evaluate such an integral for a given surface, we need to parametrize the surface similarly to what we did for the line integral. A surface in \mathbb{R}^3 is parametrized with two variables; let us call them u and v . This means that we define a function $\mathbf{r} : U \times V \rightarrow S$ (so that $u \in U$ and $v \in V$), and we write $\mathbf{r}(u, v)$ for the function value at each point on the surface. The partial derivatives $\partial \mathbf{r} / \partial u$ and $\partial \mathbf{r} / \partial v$ yield vectors that are tangent to the surface at each point, pointing in the direction of increasing u and v , respectively.

We can approximate the surface integral as a Riemann sum of contributions from small surface elements:

$$\sum_k F(\mathbf{r}_k) \delta a_k, \quad (1.72)$$

where k is an element index, $F(\mathbf{r}_k)$ is the vector field value inside the element (since the elements are small, it can be assumed that F has approximately the same value within each element), and δa_k is the area of element k . Such a surface parametrization is depicted in Figure 1.10. Therefore, each surface element k resembles a parallelogram with sides

$$\mathbf{s}_k = \left. \frac{\partial \mathbf{r}}{\partial u} \right|_{\mathbf{r}=\mathbf{r}_k} \delta u_k \text{ and } \mathbf{t}_k = \left. \frac{\partial \mathbf{r}}{\partial v} \right|_{\mathbf{r}=\mathbf{r}_k} \delta v_k, \quad (1.73)$$

where the partial derivatives are evaluated inside the element k (e.g., \mathbf{r}_k could be the parallelogram center). In general, elements do not have to be equal, so we kept the subscript k in δu_k and δv_k . The element area is found by taking a cross product:

$$\delta a_k = \|\mathbf{s}_k \times \mathbf{t}_k\| = \left\| \frac{\partial \mathbf{r}}{\partial u} \times \frac{\partial \mathbf{r}}{\partial v} \right\| \delta u_k \delta v_k, \quad (1.74)$$

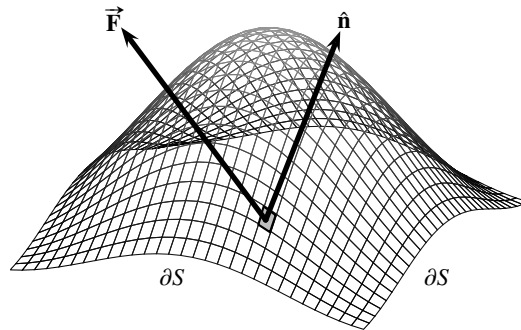


Figure 1.10 The integral of a vector field over the surface S is approximated by a Riemann sum of dot products.

assuming $\delta u_k > 0$ and $\delta v_k > 0$. Hence, the surface integral can be evaluated by

$$\iint_S F(\mathbf{r}) da = \iint_{U \times V} F(\mathbf{r}(u, v)) \left\| \frac{\partial \mathbf{r}}{\partial u} \times \frac{\partial \mathbf{r}}{\partial v} \right\| du dv. \quad (1.75)$$

If the surface is piecewise smooth, e.g., a cube, we break the integral into pieces, one for each smooth part.

The **surface integral of a vector field** $\vec{\mathbf{F}} : \Omega \rightarrow \mathbb{R}^3$ is defined on a surface $S \subset \Omega$ as the double integral

$$\iint_S \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{a} = \iint_S \vec{\mathbf{F}}(\mathbf{r}) \cdot \hat{\mathbf{n}} da, \quad (1.76)$$

where $d\mathbf{a} = da \hat{\mathbf{n}}$ is a differential area vector whose direction is determined by $\hat{\mathbf{n}}$, that is, a normal vector of unit length at each point \mathbf{r} on the surface. (We could have denoted this as a function $\hat{\mathbf{n}}(\mathbf{r})$, but this would quickly become cumbersome.) The direction of $\hat{\mathbf{n}}$ is obtained from the assumed orientation of the surface.

This kind of surface integral can also be visualized as a Riemann sum of dot product-based contributions from small elements on the surface, as shown in Figure 1.10. Taking the dot product ensures that we account only for the component of $\vec{\mathbf{F}}$ that is normal to the surface. Therefore, this kind of vector field integral is essentially a surface integral of a scalar field! If the vector field has a physical meaning of flow density, which is often the case, the surface integral may be interpreted as the aggregate vector field flow or **flux** through the surface S . Examples from physics are the magnetic flux, $\Phi = \iint_S \vec{\mathbf{B}}(\mathbf{r}) \cdot d\mathbf{a}$, and the current, $i = \iint_S \vec{\mathbf{J}}(\mathbf{r}) \cdot d\mathbf{a}$.

The surface S shown in Figure 1.10 has a boundary that is denoted by ∂S . However, a surface can be *closed*, that is, it may not have a boundary itself, but it could be the boundary of a region in space (e.g., the surface that bounds a sphere). Then we use a special integral sign:

$$\oiint_S \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{a}. \quad (1.77)$$

Using a parametrization of the surface, the surface integral can be evaluated using

$$\iint_S \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{a} = \iint_{U \times V} \vec{\mathbf{F}}(\mathbf{r}(u, v)) \cdot \left(\frac{\partial \mathbf{r}}{\partial u} \times \frac{\partial \mathbf{r}}{\partial v} \right) du dv. \quad (1.78)$$

The partial derivatives yield vectors that are tangent to the surface at each point. Therefore, their cross product will be normal to the surface. Hence, to avoid the appearance of a minus sign, the parametrization needs to be defined so that this normal vector points in the desired direction.

Although the previous case is typically what one refers to as a “surface integral of a vector field,” it is also possible to define a different kind of such an integral. In particular, we can take the surface integral

$$\vec{\mathbf{G}} = \iint_S \vec{\mathbf{F}}(\mathbf{r}) da. \quad (1.79)$$

This should be interpreted as the component-wise integration of three scalar fields,

which results in the vector $\vec{\mathbf{G}}$. For instance, the x -component of $\vec{\mathbf{G}}$ is

$$G_x = \iint_S F_x(\mathbf{r}) \, da. \quad (1.80)$$

In any case, the type of integral that we are dealing with should be apparent from the context.

Example 1.15 *Surface integral calculation.*

Calculate a surface integral on a half-sphere of radius R (i.e., without integrating over its base), as shown in Figure 1.11. The surface is oriented such that the normal vector points outwards.

Solution

This surface may be described by the parametrization

$$\mathbf{r}(\theta, \phi) = R \sin \theta \cos \phi \hat{\mathbf{i}} + R \sin \theta \sin \phi \hat{\mathbf{j}} + R \cos \theta \hat{\mathbf{k}} = R \hat{\mathbf{r}}, \quad (1.81)$$

where $\theta \in U = [0, \pi/2]$ and $\phi \in V = (-\pi, \pi]$ are the polar and azimuthal angles of a spherical coordinate system, respectively, and $\hat{\mathbf{r}}$ is the unit radial vector. In this example, the unit normal vector of the integration surface is $\hat{\mathbf{n}} = \hat{\mathbf{n}}(\theta, \phi) = \hat{\mathbf{r}}$. Hence

$$\frac{\partial \mathbf{r}}{\partial \theta} = R \cos \theta \cos \phi \hat{\mathbf{i}} + R \cos \theta \sin \phi \hat{\mathbf{j}} - R \sin \theta \hat{\mathbf{k}} = R \hat{\boldsymbol{\theta}} \quad (1.82)$$

and

$$\frac{\partial \mathbf{r}}{\partial \phi} = -R \sin \theta \sin \phi \hat{\mathbf{i}} + R \sin \theta \cos \phi \hat{\mathbf{j}} = R \sin \theta \hat{\boldsymbol{\phi}}. \quad (1.83)$$

The tangential property may be readily verified since $\partial \mathbf{r} / \partial \theta \cdot \hat{\mathbf{n}} = \partial \mathbf{r} / \partial \phi \cdot \hat{\mathbf{n}} = 0$. Using

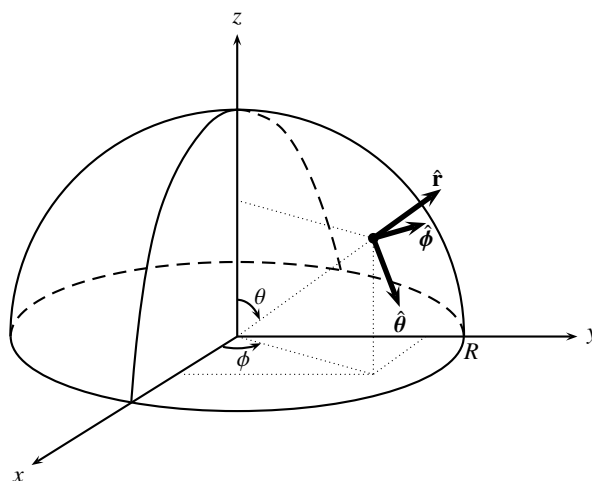


Figure 1.11 The geometry of a half-sphere displaying a spherical coordinate system.

a bit of elementary trigonometry, we obtain

$$\frac{\partial \mathbf{r}}{\partial \theta} \times \frac{\partial \mathbf{r}}{\partial \phi} = R^2(\sin^2 \theta \cos \phi \hat{\mathbf{i}} + \sin^2 \theta \sin \phi \hat{\mathbf{j}} + \sin \theta \cos \theta \hat{\mathbf{k}}) = R^2 \sin \theta \hat{\mathbf{n}}. \quad (1.84)$$

Once the surface parametrization is complete, we can proceed with the evaluation of the integral. Suppose that $\vec{\mathbf{F}}(\mathbf{r}) = \hat{\mathbf{n}}$. We would then expect the integral to yield the surface area of the half-sphere (i.e., $2\pi R^2$). Let us verify this:

$$\begin{aligned} \iint_{U \times V} \vec{\mathbf{F}}(\mathbf{r}(\theta, \phi)) \cdot \left(\frac{\partial \mathbf{r}}{\partial \theta} \times \frac{\partial \mathbf{r}}{\partial \phi} \right) d\theta d\phi &= \iint_{U \times V} \hat{\mathbf{n}} \cdot (R^2 \sin \theta \hat{\mathbf{n}}) d\theta d\phi \\ &= \int_{\phi=-\pi}^{\pi} \int_{\theta=0}^{\pi/2} R^2 \sin \theta d\theta d\phi = 2\pi R^2. \end{aligned} \quad (1.85)$$

As a second example, consider the case where $\vec{\mathbf{F}} = F \hat{\mathbf{k}}$, $F > 0$. In this case, it is more convenient to work with the expansion in Cartesian coordinates:

$$\iint_{U \times V} \vec{\mathbf{F}}(\mathbf{r}(\theta, \phi)) \cdot \left(\frac{\partial \mathbf{r}}{\partial \theta} \times \frac{\partial \mathbf{r}}{\partial \phi} \right) d\theta d\phi \quad (1.86a)$$

$$= \iint_{U \times V} F \hat{\mathbf{k}} \cdot R^2(\sin^2 \theta \cos \phi \hat{\mathbf{i}} + \sin^2 \theta \sin \phi \hat{\mathbf{j}} + \sin \theta \cos \theta \hat{\mathbf{k}}) d\theta d\phi \quad (1.86b)$$

$$= \int_{\phi=-\pi}^{\pi} \int_{\theta=0}^{\pi/2} FR^2 \sin \theta \cos \theta d\theta d\phi \quad (1.86c)$$

$$= \pi FR^2 \int_0^{\pi/2} \sin 2\theta d\theta \quad (1.86d)$$

$$= \pi FR^2. \quad (1.86e)$$

As an interesting observation, consider what would have happened if we had extended the integration to the base of the half-sphere, thus forming a closed surface. In this case, we would have obtained an additional term equal to

$$(\text{area of base})(-F) = -\pi R^2 F, \quad (1.87)$$

where the minus sign appears since the vector field $F \hat{\mathbf{k}}$ is entering the half-sphere from below, thereby opposing the outwards-pointing normal vector. Hence, the integral over the closed surface would have been $\oint_S \vec{\mathbf{F}} \cdot d\mathbf{a} = 0$. This is not unexpected since the considered vector field has zero divergence, and we could have applied Gauss' law. More on this later in §1.6.

1.4.3 Volume Integrals

The **volume integral of a scalar field** $F : \Omega \rightarrow \mathbb{R}$ is defined on a volume $V \subset \Omega$ as the triple integral

$$\iiint_V F(\mathbf{r}) dv, \quad (1.88)$$

where dv represents the differential volume. For instance, in Cartesian coordinates, $dv = dx dy dz$. In physics, this type of integral is often used when F represents some kind of density function, so integrating over a volume yields the total value of the quantity of interest, such as mass or energy. Again, the volume integral may be approximated by a Riemann sum over small volumes that cover the entire space V .

The **volume integral of a vector field** $\vec{F} : \Omega \rightarrow \mathbb{R}^3$ is obtained by performing component-wise volume integration on \vec{F} . It is denoted by

$$\vec{G} = \iiint_V \vec{F}(\mathbf{r}) dv. \quad (1.89)$$

For instance, the x -component of \vec{G} is

$$G_x = \iiint_V F_x(\mathbf{r}) dv, \quad (1.90)$$

so it is a volume integral of the scalar field F_x .

1.5 Differential Operators on Scalar and Vector Fields

In this section, we recall the various differential operators that we will encounter in this text while manipulating Maxwell's equations. Our approach here will be conceptual rather than numerical.

These operators will be associated with formulas that involve partial derivatives. For these to have meaning, it is required that the fields are differentiable (smooth) with respect to position. This will typically be the case within the volume of a single material whose properties change smoothly with position. However, discontinuities will be present across material boundaries. Since electric machines are constructed using materials of different properties, we need to be mindful of this fact.

1.5.1 Gradient

The **gradient** of a scalar field $F : \Omega \subset \mathbb{R}^3 \rightarrow \mathbb{R}$ is defined as

$$\nabla F(\mathbf{r}) = \frac{\partial F}{\partial x} \hat{\mathbf{i}} + \frac{\partial F}{\partial y} \hat{\mathbf{j}} + \frac{\partial F}{\partial z} \hat{\mathbf{k}}, \quad (1.91)$$

where the partial derivatives (assuming they exist) are evaluated at $\mathbf{r} = (x, y, z)$. If the scalar field is only defined in two dimensions, then we drop the third term in the above equation. For example, the electric field is defined as the (negative) gradient of the potential, $\vec{E} = -\nabla\phi$; see Examples 1.12 and 1.13.

The symbol ∇ is called the **nabla** or **del**. It plays the role of an *operator*, and can be defined as

$$\nabla = \frac{\partial}{\partial x} \hat{\mathbf{i}} + \frac{\partial}{\partial y} \hat{\mathbf{j}} + \frac{\partial}{\partial z} \hat{\mathbf{k}}. \quad (1.92)$$

The nabla operator is also used in the divergence and the curl, which will be introduced subsequently. Note that the gradient operates on a scalar function and returns a vector.

The expression (1.91) is valid only in Cartesian coordinates. In other coordinate systems (e.g., spherical or cylindrical) the expressions are different; these are not listed here, but they can be found in calculus texts or online.

For an interpretation of the gradient, consider two points, $P_1, P_2 \in \Omega$, displaced by \mathbf{r}_1 and \mathbf{r}_2 from the origin O , respectively, as shown in Figure 1.12. Let us fix P_1 , allowing P_2 to move arbitrarily close to P_1 , and denote by $\delta\mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1$ their relative displacement. Suppose that $\delta\mathbf{r} = \delta x \hat{\mathbf{i}} + \delta y \hat{\mathbf{j}} + \delta z \hat{\mathbf{k}}$. Then, using a Taylor expansion of F at \mathbf{r}_1 , the field changes by

$$\delta F = F(\mathbf{r}_2) - F(\mathbf{r}_1) \approx \frac{\partial F}{\partial x} \delta x + \frac{\partial F}{\partial y} \delta y + \frac{\partial F}{\partial z} \delta z, \quad (1.93)$$

as a first-order approximation, with partial derivatives evaluated at \mathbf{r}_1 ; hence,

$$\delta F \approx \nabla F(\mathbf{r}_1) \cdot \delta\mathbf{r}. \quad (1.94)$$

In the special case where P_1 is a local maximum or minimum point of F , the gradient is zero (i.e., all partial derivatives are zero). Otherwise, a nontrivial-level curve through P_1 exists (as shown in Figure 1.12). Suppose that both points are on this curve, $F(\mathbf{r}_1) = F(\mathbf{r}_2) = C$, so that $\delta F = 0$. As P_2 comes arbitrarily close to P_1 , the displacement $\delta\mathbf{r}$ becomes tangent to the curve C . If $\hat{\mathbf{t}}$ is a unit tangent vector at \mathbf{r}_1 , then $\mathbf{r}_2 = \mathbf{r}_1 + h \hat{\mathbf{t}}$. So, we have $\delta F = 0 = \nabla F(\mathbf{r}_1) \cdot h \hat{\mathbf{t}} + O(h^2)$, as $h \rightarrow 0$. Therefore, we can argue that *the gradient has to be normal to the level set at P_1 since a small (but nonzero) step of length h in the tangent direction would lead to $\nabla F(\mathbf{r}_1) \cdot \hat{\mathbf{t}} \approx 0$* . Figures 1.7 and 1.8 on pages 17 and 19, respectively, provide an illustration of this property using the electrostatic field as an example: the E -field is normal to the equipotential surfaces.

Furthermore, this line of thinking can lead us to the conclusion that *the gradient points at the direction where the field increases the most*. To see this, note that the dot product in Equation (1.94), $\delta F = \|\nabla F\| \|\delta\mathbf{r}\| \cos \theta$, becomes maximum when the angle θ between the displacement and the gradient is zero (maintaining a displacement of fixed length). In Figure 1.7, the electric field points towards the direction of maximum potential *decrease* due to the presence of the minus sign in its definition.

In general, the gradient is related to the **directional derivative** along the direction \mathbf{v} ,

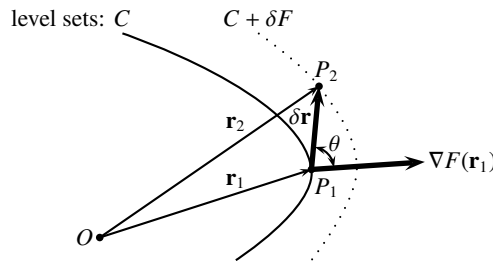


Figure 1.12 Geometrical interpretation of the gradient in two dimensions.

where it is customary to use a unit vector \mathbf{v} (i.e., $\|\mathbf{v}\| = 1$). This is defined as

$$\nabla F_{\mathbf{v}}(\mathbf{r}) = \left. \frac{d}{dh} F(\mathbf{r} + h\mathbf{v}) \right|_{h=0} \quad (1.95a)$$

$$= \lim_{h \rightarrow 0} \frac{F(\mathbf{r} + h\mathbf{v}) - F(\mathbf{r})}{h} \quad (1.95b)$$

$$= \lim_{h \rightarrow 0} \frac{\nabla F(\mathbf{r}) \cdot (h\mathbf{v}) + (\text{high-order terms } O(h^2))}{h} \quad (1.95c)$$

$$= \nabla F(\mathbf{r}) \cdot \mathbf{v}. \quad (1.95d)$$

Example 1.16 *Gradient calculation.*

Calculate the gradient of the scalar field $F : \mathbb{R}^2 \rightarrow \mathbb{R}$ given by

$$F(x, y) = x^2 + y^2. \quad (1.96)$$

Solution

The level sets of F are circles centered at the origin. The gradient of F is

$$\nabla F(x, y) = \frac{\partial F}{\partial x} \hat{\mathbf{i}} + \frac{\partial F}{\partial y} \hat{\mathbf{j}} = 2x \hat{\mathbf{i}} + 2y \hat{\mathbf{j}}. \quad (1.97)$$

This vector points radially away from the origin, so it is perpendicular to the level sets, as expected. Note that $\nabla F = 0 \Rightarrow x = y = 0$. The gradient is zero at the origin, where F obtains its minimum value.

Example 1.17 *Conservative vector fields.*

Suppose that a vector field $\vec{\mathbf{F}} : \Omega \rightarrow \mathbb{R}^3$ is the gradient of a scalar field $\varphi : \Omega \rightarrow \mathbb{R}$, that is, $\vec{\mathbf{F}}(\mathbf{r}) = \nabla \varphi(\mathbf{r})$. For any (piecewise) smooth path C within Ω , which is parametrized by $t \in [a, b]$ so that $\mathbf{r}(t) = (x(t), y(t), z(t))$, we have

$$\int_C \vec{\mathbf{F}}(\mathbf{r}) \cdot d\mathbf{r} = \int_a^b \nabla \varphi(\mathbf{r}(t)) \cdot \mathbf{r}'(t) dt \quad (1.98a)$$

$$= \int_a^b \left(\frac{\partial \varphi}{\partial x} \frac{dx}{dt} + \frac{\partial \varphi}{\partial y} \frac{dy}{dt} + \frac{\partial \varphi}{\partial z} \frac{dz}{dt} \right) dt \quad (1.98b)$$

$$= \int_a^b \frac{d\varphi(\mathbf{r}(t))}{dt} dt \quad (1.98c)$$

$$= \varphi(\mathbf{r}(b)) - \varphi(\mathbf{r}(a)), \quad (1.98d)$$

where we used the chain rule for the derivative $d\varphi/dt$, and the fundamental theorem of calculus in the last step. This means that the line integral does not depend on the path but only on its two endpoints. The integral, in other words, is path independent. Also, if the two endpoints coincide so that the path is closed, the integral is zero. It can be shown that the converse is also true, i.e., if $\vec{\mathbf{F}}$ is path independent, then it is the gradient of some scalar field. This type of vector field is called **conservative**. For example, the electrostatic field is conservative since $\vec{\mathbf{E}} = -\nabla \varphi$ (a minus sign is added by convention).

1.5.2 Divergence

Often a vector field $\vec{\mathbf{F}} : \Omega \rightarrow \mathbb{R}^3$ has a physical meaning of flow density (i.e., flow per m^2), in which case we may be interested in calculating the total flow out of a given volume $V \in \Omega$ surrounding a point of interest P . This would be the case if there exist sources or sinks of the flow; for instance, this happens around point charges as shown in Figure 1.7 on page 17.

Hence, we define a scalar quantity called the **divergence** as the ratio

$$\nabla \cdot \vec{\mathbf{F}}(\mathbf{r}) = \lim_{|V| \rightarrow 0} \frac{\text{total outwards flow}}{|V|}, \quad (1.99)$$

where $|V| = \iiint_V dv$ denotes the volume size. Therefore, the divergence has units of flow per m^3 . Examples from physics are the divergence of the electric displacement field, $\nabla \cdot \vec{\mathbf{D}} = \rho$, and the divergence of the magnetic field, $\nabla \cdot \vec{\mathbf{B}} = 0$. A vector field that has zero divergence everywhere, like the magnetic field, is called **incompressible** or **solenoidal**.

In Cartesian coordinates, when the vector field $\vec{\mathbf{F}} = (F_x, F_y, F_z)$ is differentiable, the divergence is given by the formula

$$\nabla \cdot \vec{\mathbf{F}}(\mathbf{r}) = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z}, \quad (1.100)$$

where the partial derivatives are evaluated at $\mathbf{r} = (x, y, z)$, that is, the displacement of P from the origin. For a 2-D field, we drop the last term. (In spherical or cylindrical coordinates, different formulas apply.) Note how the divergence can be obtained formally by taking the dot product between the nabla operator in Equation (1.92) and the vector field.

Proof Let us illustrate this concept using a small cube surrounding a center point $P = (x', y', z') = (a, a, a)/2$, as shown in Figure 1.13. For computational convenience, the

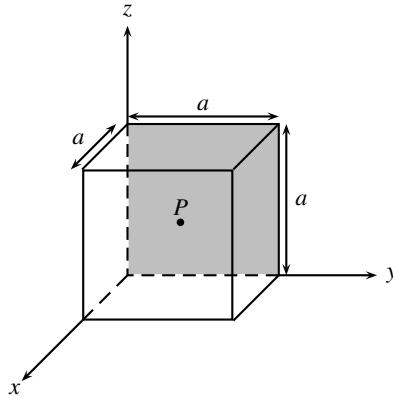


Figure 1.13 Illustration of the divergence as an outwards flow from a cube surrounding a point P at its center.