Theory of Dimensioning

An Introduction to Parameterizing Geometric Models

Vijay Srinivasan
IBM Corporation
White Plains
and Columbia University
New York, New York, U.S.A.

MARCEL DEKKER, INC.   NEW YORK • BASEL

Theory of Dimensioning

An Introduction to Parameterizing Geometric Models

Vijay Srinivasan
IBM Corporation
White Plains
and Columbia University
New York, New York, U.S.A.

MARCEL DEKKER, INC.
NEW YORK • BASEL

Library of Congress Cataloging-in-Publication Data

A catalog record for this book is available from the Library of Congress.

ISBN 0-203-91303-5 Master e-book ISBN

ISBN: 0-8247-4624-4 (Print Edition)

Headquarters

Marcel Dekker, Inc. 270 Madison Avenue, New York, NY 10016, U.S.A.
tel: 212–696–9000; fax: 212–685–4540

This edition published in the Taylor & Francis e-Library, 2005.

"To purchase your own copy of this or any of Taylor & Francis or Routledge's collection of thousands of eBooks please go to www.eBookstore.tandf.co.uk."

Distribution and Customer Service

Marcel Dekker, Inc. Cimarron Road, Monticello, New York 12701, U.S.A.
tel: 800–228–1160; fax: 845–796–1772

Eastern Hemisphere Distribution

Marcel Dekker AG, Hutgasse 4, Postfach 812, CH-4001 Basel, Switzerland
tel: 41–61–260–6300; fax: 41–61–260–6333

World Wide Web

http://www.dekker.com

The publisher offers discounts on this book when ordered in bulk quantities. For more information, write to Special Sales/Professional Marketing at the headquarters address above.

MECHANICAL ENGINEERING

A Series of Textbooks and Reference Books

Founding Editor

L.L.Faulkner

Columbus Division, Battelle Memorial Institute
and Department of Mechanical Engineering
The Ohio State University
Columbus, Ohio

1. *Spring Designer's Handbook,* Harold Carlson
2. *Computer-Aided Graphics and Design,* Daniel L.Ryan
3. *Lubrication Fundamentals,* J.George Wills
4. *Solar Engineering for Domestic Buildings,* William A.Himmelman
5. *Applied Engineering Mechanics: Statics and Dynamics,* G.Boothroyd and C.Poli
6. *Centrifugal Pump Clinic,* Igor J.Karassik
7. *Computer-Aided Kinetics for Machine Design,* Daniel L.Ryan
8. *Plastics Products Design Handbook, Part A: Materials and Components; Part B: Processes and Design for Processes,* edited by Edward Miller
9. *Turbomachinery: Basic Theory and Applications,* Earl Logan, Jr.
10. *Vibrations of Shells and Plates,* Werner Soedel
11. *Flat and Corrugated Diaphragm Design Handbook,* Mario Di Giovanni
12. *Practical Stress Analysis in Engineering Design,* Alexander Blake
13. *An Introduction to the Design and Behavior of Bolted Joints,* John H. Bickford
14. *Optimal Engineering Design: Principles and Applications,* James N.Siddall
15. *Spring Manufacturing Handbook,* Harold Carlson

16. *Industrial Noise Control: Fundamentals and Applications,* edited by Lewis H. Bell

17. *Gears and Their Vibration: A Basic Approach to Understanding Gear Noise,* J.Derek Smith

18. *Chains for Power Transmission and Material Handling: Design and Applications Handbook,* American Chain Association

19. *Corrosion and Corrosion Protection Handbook,* edited by Philip A. Schweitzer

20. *Gear Drive Systems: Design and Application,* Peter Lynwander

21. *Controlling In-Plant Airborne Contaminants: Systems Design and Calculations,* John D.Constance

22. *CAD/CAM Systems Planning and Implementation,* Charles S.Knox

23. *Probabilistic Engineering Design: Principles and Applications,* James N. Siddall

24. *Traction Drives: Selection and Application,* Frederick W.Heilich III and Eugene E.Shube

25. *Finite Element Methods: An Introduction,* Ronald L.Huston and Chris E. Passerello

26. *Mechanical Fastening of Plastics: An Engineering Handbook,* Brayton Lincoln, Kenneth J.Gomes, and James F.Braden

27. *Lubrication in Practice: Second Edition,* edited by W.S.Robertson

28. *Principles of Automated Drafting,* Daniel L.Ryan

29. *Practical Seal Design,* edited by Leonard J.Martini

30. *Engineering Documentation for CAD/CAM Applications,* Charles S.Knox

31. *Design Dimensioning with Computer Graphics Applications,* Jerome C. Lange

32. *Mechanism Analysis: Simplified Graphical and Analytical Techniques,* Lyndon O.Barton

33. *CAD/CAM Systems: Justification, Implementation, Productivity Measurement,* Edward J.Preston, George W.Crawford, and Mark E.Coticchia

34. *Steam Plant Calculations Manual,* V.Ganapathy

35. *Design Assurance for Engineers and Managers,* John A.Burgess

36. *Heat Transfer Fluids and Systems for Process and Energy Applications,* Jasbir Singh

37. *Potential Flows: Computer Graphic Solutions,* Robert H.Kirchhoff

38. *Computer-Aided Graphics and Design: Second Edition,* Daniel L.Ryan

39. *Electronically Controlled Proportional Valves: Selection and Application,* Michael J.Tonyan, edited by Tobi Goldoftas

40. *Pressure Gauge Handbook,* AMETEK, U.S.Gauge Division, edited by Philip W.Harland

41. *Fabric Filtration for Combustion Sources: Fundamentals and Basic Technology,* R.P.Donovan

42. *Design of Mechanical Joints,* Alexander Blake

43. *CAD/CAM Dictionary,* Edward J.Preston, George W.Crawford, and Mark E. Coticchia

44. *Machinery Adhesives for Locking, Retaining, and Sealing,* Girard S.Haviland

45. *Couplings and Joints: Design, Selection, and Application,* Jon R.Mancuso

46. *Shaft Alignment Handbook,* John Piotrowski

47. *BASIC Programs for Steam Plant Engineers: Boilers, Combustion, Fluid Flow, and Heat Transfer,* V.Ganapathy

48. *Solving Mechanical Design Problems with Computer Graphics,* Jerome C. Lange

49. *Plastics Gearing: Selection and Application,* Clifford E.Adams

50. *Clutches and Brakes: Design and Selection,* William C.Orthwein

51. *Transducers in Mechanical and Electronic Design,* Harry L.Trietley

52. *Metallurgical Applications of Shock-Wave and High-Strain-Rate Phenomena,* edited by Lawrence E.Murr, Karl P.Staudhammer, and Marc A. Meyers

53. *Magnesium Products Design,* Robert S.Busk

54. *How to Integrate CAD/CAM Systems: Management and Technology,* William D.Engelke

55. *Cam Design and Manufacture: Second Edition;* with cam design software for the IBM PC and compatibles, disk included, Preben W.Jensen

56. *Solid-State AC Motor Controls: Selection and Application,* Sylvester Campbell

57. *Fundamentals of Robotics,* David D.Ardayfio

58. *Belt Selection and Application for Engineers,* edited by Wallace D.Erickson

59. *Developing Three-Dimensional CAD Software with the IBM PC,* C.Stan Wei

60. *Organizing Data for CIM Applications,* Charles S.Knox, with contributions by Thomas C.Boos, Ross S.Culverhouse, and Paul F.Muchnicki

61. *Computer-Aided Simulation in Railway Dynamics,* by Rao V.Dukkipati and Joseph R.Amyot

62. *Fiber-Reinforced Composites: Materials, Manufacturing, and Design,* P.K. Mallick

63. *Photoelectric Sensors and Controls: Selection and Application,* Scott M. Juds

64. *Finite Element Analysis with Personal Computers,* Edward R.Champion, Jr, and J.Michael Ensminger

65. *Ultrasonics: Fundamentals, Technology, Applications: Second Edition, Revised and Expanded,* Dale Ensminger

66. *Applied Finite Element Modeling: Practical Problem Solving for Engineers,* Jeffrey M.Steele

67. *Measurement and Instrumentation in Engineering: Principles and Basic Laboratory Experiments,* Francis S.Tse and Ivan E.Morse

68. *Centrifugal Pump Clinic: Second Edition, Revised and Expanded,* Igor J. Karassik

69. *Practical Stress Analysis in Engineering Design: Second Edition, Revised and Expanded,* Alexander Blake

70. *An Introduction to the Design and Behavior of Bolted Joints: Second Edition, Revised and Expanded,* John H.Bickford

71. *High Vacuum Technology: A Practical Guide,* Marsbed H.Hablanian

72. *Pressure Sensors: Selection and Application,* Duane Tandeske

73. *Zinc Handbook: Properties, Processing, and Use in Design,* Frank Porter

74. *Thermal Fatigue of Metals,* Andrzej Weronski and Tadeusz Hejwowski

75. *Classical and Modern Mechanisms for Engineers and Inventors,* Preben W. Jensen

76. *Handbook of Electronic Package Design,* edited by Michael Pecht

77. *Shock-Wave and High-Strain-Rate Phenomena in Materials,* edited by Marc A.Meyers, Lawrence E.Murr, and Karl P.Staudhammer

78. *Industrial Refrigeration: Principles, Design and Applications,* P.C.Koelet

79. *Applied Combustion,* Eugene L.Keating

80. *Engine Oils and Automotive Lubrication,* edited by Wilfried J.Bartz

81. *Mechanism Analysis: Simplified and Graphical Techniques, Second Edition, Revised and Expanded,* Lyndon O.Barton

82. *Fundamental Fluid Mechanics for the Practicing Engineer,* James W. Murdock

83. *Fiber-Reinforced Composites: Materials, Manufacturing, and Design, Second Edition, Revised and Expanded,* P.K.Mallick

84. *Numerical Methods for Engineering Applications,* Edward R.Champion, Jr.

85. *Turbomachinery: Basic Theory and Applications, Second Edition, Revised and Expanded,* Earl Logan, Jr.

86. *Vibrations of Shells and Plates: Second Edition, Revised and Expanded,* Werner Soedel

87. *Steam Plant Calculations Manual: Second Edition, Revised and Expanded,* V.Ganapathy

88. *Industrial Noise Control: Fundamentals and Applications, Second Edition, Revised and Expanded,* Lewis H.Bell and Douglas H.Bell

89. *Finite Elements: Their Design and Performance,* Richard H.MacNeal

90. *Mechanical Properties of Polymers and Composites: Second Edition, Revised and Expanded,* Lawrence E.Nielsen and Robert F.Landel

91. *Mechanical Wear Prediction and Prevention,* Raymond G.Bayer

92. *Mechanical Power Transmission Components,* edited by David W.South and Jon R.Mancuso

93. *Handbook of Turbomachinery,* edited by Earl Logan, Jr.

94. *Engineering Documentation Control Practices and Procedures,* Ray E. Monahan

95. *Refractory Linings Thermomechanical Design and Applications,* Charles A. Schacht

96. *Geometric Dimensioning and Tolerancing: Applications and Techniques for Use in Design, Manufacturing, and Inspection,* James D.Meadows

97. *An Introduction to the Design and Behavior of Bolted Joints: Third Edition, Revised and Expanded,* John H.Bickford

98. *Shaft Alignment Handbook: Second Edition, Revised and Expanded,* John Piotrowski

99. *Computer-Aided Design of Polymer-Matrix Composite Structures,* edited by Suong Van Hoa

100. *Friction Science and Technology,* Peter J.Blau

101. *Introduction to Plastics and Composites: Mechanical Properties and Engineering Applications,* Edward Miller

102. *Practical Fracture Mechanics in Design,* Alexander Blake

103. *Pump Characteristics and Applications,* Michael W.Volk

104. *Optical Principles and Technology for Engineers,* James E.Stewart

105. *Optimizing the Shape of Mechanical Elements and Structures,* A.A.Seireg and Jorge Rodriguez

106. *Kinematics and Dynamics of Machinery,* Vladimír Stejskal and Michael Valášek

107. *Shaft Seals for Dynamic Applications,* Les Horve

108. *Reliability-Based Mechanical Design,* edited by Thomas A.Cruse

109. *Mechanical Fastening, Joining, and Assembly,* James A.Speck

110. *Turbomachinery Fluid Dynamics and Heat Transfer,* edited by Chunill Hah

111. *High-Vacuum Technology: A Practical Guide, Second Edition, Revised and Expanded,* Marsbed H.Hablanian

112. *Geometric Dimensioning and Tolerancing: Workbook and Answerbook,* James D.Meadows

113. *Handbook of Materials Selection for Engineering Applications,* edited by G. T.Murray

114. *Handbook of Thermoplastic Piping System Design,* Thomas Sixsmith and Reinhard Hanselka

115. *Practical Guide to Finite Elements: A Solid Mechanics Approach,* Steven M. Lepi

116. *Applied Computational Fluid Dynamics,* edited by Vijay K.Garg

117. *Fluid Sealing Technology,* Heinz K.Muller and Bernard S.Nau

118. *Friction and Lubrication in Mechanical Design,* A.A.Seireg

119. *Influence Functions and Matrices,* Yuri A.Melnikov

120. *Mechanical Analysis of Electronic Packaging Systems,* Stephen A. McKeown

121. *Couplings and Joints: Design, Selection, and Application, Second Edition, Revised and Expanded,* Jon R.Mancuso

122. *Thermodynamics: Processes and Applications,* Earl Logan, Jr.

123. *Gear Noise and Vibration,* J.Derek Smith

124. *Practical Fluid Mechanics for Engineering Applications,* John J.Bloomer

125. *Handbook of Hydraulic Fluid Technology,* edited by George E.Totten

126. *Heat Exchanger Design Handbook,* T.Kuppan

127. *Designing for Product Sound Quality,* Richard H.Lyon

128. *Probability Applications in Mechanical Design,* Franklin E.Fisher and Joy R. Fisher

129. *Nickel Alloys,* edited by Ulrich Heubner

130. *Rotating Machinery Vibration: Problem Analysis and Troubleshooting,* Maurice L.Adams, Jr.

131. *Formulas for Dynamic Analysis,* Ronald L.Huston and C.Q.Liu

132. *Handbook of Machinery Dynamics,* Lynn L.Faulkner and Earl Logan, Jr.

133. *Rapid Prototyping Technology: Selection and Application,* Kenneth G. Cooper

134. *Reciprocating Machinery Dynamics: Design and Analysis,* Abdulla S. Rangwala

135. *Maintenance Excellence: Optimizing Equipment Life-Cycle Decisions,* edited by John D.Campbell and Andrew K.S.Jardine

136. *Practical Guide to Industrial Boiler Systems,* Ralph L.Vandagriff

137. *Lubrication Fundamentals: Second Edition, Revised and Expanded,* D.M. Pirro and A.A.Wessol

138. *Mechanical Life Cycle Handbook: Good Environmental Design and Manufacturing,* edited by Mahendra S.Hundal

139. *Micromachining of Engineering Materials,* edited by Joseph McGeough

140. *Control Strategies for Dynamic Systems: Design and Implementation,* John H.Lumkes, Jr.

141. *Practical Guide to Pressure Vessel Manufacturing,* Sunil Pullarcot

142. *Nondestructive Evaluation: Theory, Techniques, and Applications,* edited by Peter J.Shull

143. *Diesel Engine Engineering: Thermodynamics, Dynamics, Design, and Control,* Andrei Makartchouk

144. *Handbook of Machine Tool Analysis,* Ioan D.Marinescu, Constantin Ispas, and Dan Boboc

145. *Implementing Concurrent Engineering in Small Companies,* Susan Carlson Skalak

146. *Practical Guide to the Packaging of Electronics: Thermal and Mechanical Design and Analysis,* Ali Jamnia

147. *Bearing Design in Machinery: Engineering Tribology and Lubrication,* Avraham Harnoy

148. *Mechanical Reliability Improvement: Probability and Statistics for Experimental Testing,* R.E.Little

149. *Industrial Boilers and Heat Recovery Steam Generators: Design, Applications, and Calculations,* V.Ganapathy

150. *The CAD Guidebook: A Basic Manual for Understanding and Improving Computer-Aided Design,* Stephen J.Schoonmaker

151. *Industrial Noise Control and Acoustics,* Randall F.Barron

152. *Mechanical Properties of Engineered Materials,* Wolé Soboyejo

153. *Reliability Verification, Testing, and Analysis in Engineering Design,* Gary S. Wasserman

154. *Fundamental Mechanics of Fluids: Third Edition,* I.G.Currie

155. *Intermediate Heat Transfer,* Kau-Fui Vincent Wong

156. *HVAC Water Chillers and Cooling Towers: Fundamentals, Application, and Operation,* Herbert W.Stanford III

157. *Gear Noise and Vibration: Second Edition, Revised and Expanded,* J. Derek Smith

158. *Handbook of Turbomachinery: Second Edition, Revised and Expanded,* Earl Logan, Jr., and Ramendra Roy

159. *Piping and Pipeline Engineering: Design, Construction, Maintenance, Integrity, and Repair,* George A.Antaki

160. *Turbomachinery: Design and Theory,* Rama S.R.Gorla and Aijaz Ahmed Khan

161. *Target Costing: Market-Driven Product Design,* M.Bradford Clifton, Henry M.B.Bird, Robert E.Albano, and Wesley P.Townsend

162. *Fluidized Bed Combustion,* Simeon N.Oka

163. *Theory of Dimensioning: An Introduction to Parameterizing Geometric Models,* Vijay Srinivasan

164. *Structural Analysis of Polymeric Composite Materials,* Mark E.Tuttle

Additional Volumes in Preparation

Handbook of Pneumatic Conveying Engineering, David Mills, Mark G. Jones, and Vijay K.Agarwal

Handbook of Mechanical Design Based on Material Composition, George E. Totten, Lin Xie, and Kiyoshi Funatani

Mechanical Wear Fundamentals and Testing: Second Edition, Revised and Expanded, Raymond G.Bayer

Engineering Design for Wear Second Edition, Revised and Expanded, Raymond G.Bayer

Clutches and Brakes: Design and Selection, Second Edition, William C. Orthwein

Progressing Cavity Pumps, Downhole Pumps, and Mudmotors, Lev Nelik

Mechanical Engineering Software

Spring Design with an IBM PC, Al Dietrich

Mechanical Design Failure Analysis: With Failure Analysis System Software for the IBM PC, David G.Ullman

To Pavani, Pavitra, and Gopal

Preface

Dimensioning is one of the most common engineering activities in industry. All engineers, regardless of their field of specialization, have an intuitive idea of how to dimension a sketch of an object of their interest. In engineering schools, most engineers are trained in drafting or (more recently) in engineering graphics to create stylized representations of sketches and to dimension them. Modern computer-aided drafting and design systems provide functions that make sketching and dimensioning easy to generate and modify using a computer. National and international standards bodies have been standardizing projected views and symbols for dimensioning so that anyone trained in these standards can generate or interpret drawings that conform to these standards.

However, a theory of dimensioning that is worthy of our information age is absent. Yes, there have been some ad hoc theories to explain various steps involved in dimensioning. These theories are, at best, codifications of useful industrial practices in dimensioning that have evolved over a long period of time. However, it has been a matter of some embarrassment to those of us who work in this field that we have not been able to give a scientific explanation of dimensioning to our students and colleagues. This book remedies that unfortunate predicament. It presents a theory of dimensioning that is synthesized from several areas of geometry, starting from the works of Euclid and culminating in recent advances in classification of continuous symmetry groups. It is worth reflecting on how this synthesis has been achieved.

In the late 1980s, it became clear that existing standardized practices of dimensioning, tolerancing, and associated metrological means of verification had hit a hard limit in industry, which was undergoing profound changes brought about by the information revolution. This concern was openly expressed in 1989 at an international meeting organized by the American Society of Mechanical Engineers (ASME) and funded by the U.S. National Science Foundation. The meeting resulted in two initiatives. One was to quickly produce a mathematical companion to the ASME Y14.5 standard on dimensioning and tolerancing; this goal was achieved in 1994 by a newly formed ASME Y14.5.1 subcommittee. The other initiative was a more ambitious international effort to produce an integrated, mathematically rigorous chain of ISO standards for dimensioning, tolerancing, and associated metrological verification methods. In the mid-1990s, relevant ISO groups addressed this task by organizing as one body, ISO Technical Committee (ISO/TC) 213. The committee is currently issuing a chain of standards as envisioned earlier. As a result of these two initiatives, greater attention has been focused on developing a mathematically rigorous approach to dimensioning and tolerancing.

Several researchers in industry and in universities around the world had anticipated the need for a mathematical theory of dimensioning and tolerancing even before these recent developments. A particularly noteworthy theory was introduced in France under the rubric of *technologically and topologically related surfaces* (TTRS); it was slowly disseminated to the English-speaking world in the 1990s. This theory was based on some new results in classification of continuous symmetry groups. The mathematical correctness of the work was investigated and verified by a U.S. group (consisting of researchers from Boeing and IBM) in the mid-1990s. This work seemed to provide a useful theory of relative positioning, which was an important, and previously unavailable, requirement for a theory of dimensioning.

While such efforts to find mathematically firm foundations for dimensioning and tolerancing were progressing, a parallel effort was brewing in another ISO group responsible for producing standards for exchange of geometric information in product models. Such standards—known loosely as STEP (STandard for Exchange of Product model data) standards—are issued by the ISO/TC 184/SC 4 subcommittee. In the late 1990s this group initiated an effort to standardize the exchange of parameters and constraints specified in geometric models. During my participation in this effort on behalf of IBM, it became clear that the theory being developed for dimensioning could also serve as a theory of geometric parameterization, thereby enhancing the value of such a theory significantly. In addition, it became evident that a theory of dimensioning is of value independent of any consideration of tolerancing. This realization imposed a heightened sense of urgency to bring the theoretical work on dimensioning to a conclusion and publish it.

It was in the deliberations in these different, but related, standards committees during the years of 2000 and 2001 that a satisfactory synthesis was achieved. It resulted in a unification of several seemingly different theoretical ideas on size dimensioning and position dimensioning using the simple notion of congruence. It turns out that it is the congruence of point-sets and tuples (that is, rigid collections) of point-sets that matter. This gave the theory of dimensioning a conceptual clarity that was easy to explain and to understand.

You may ask, *What is the use of a theory of dimensioning? After all, haven't engineers been dimensioning drawings for centuries?* In fact, the power of a theory becomes evident only when this task of dimensioning (and parameterizing) is computerized and the resulting information is interpreted by other computers. Data models have to be developed, and their completeness has to be ensured; likewise, algorithms need to be designed and their correctness proved. A theory helps us to accomplish these aims. Accordingly, CAD/CAM software developers and users, and the related standards committees, have an interest in the theory of dimensioning. I hope that this book serves this interest, and that researchers, engineers, and software developers find it helpful to their work. It is important to point out that the theory presented here puts the current practice of dimensioning on a strong theoretical footing. It is not disruptive, in the sense that it proposes to reinforce and improve rather than replace the current practice. This is a critical factor for industrial acceptance of this theory and the associated standardization.

Since the mid-1990s, I have been introducing bits and pieces of published work in dimensioning and tolerancing in my geometric modeling course at Columbia University. It is offered as a graduate-level course, but nearly half of the students are undergraduates who choose it as an elective. Most of the students are from the mechanical engineering department, but several have been from biomedical engineering, computer science, civil engineering, and industrial engineering departments. The lecture notes for this course were finally compiled into a "Theory of Dimensioning" text during the Spring 2002 semester; this book is a direct result of this effort. So I am confident that other teachers will find it a useful textbook for their students.

I start the course by emphasizing that a theory of dimensioning is a prerequisite to a good understanding of tolerancing. I also point out that the course is an introduction to parameterizing geometric models. The link to tolerancing particularly motivates mechanical engineering students, while the larger scope of parametric geometric modeling attracts a wider audience. Teachers may find it advantageous to emphasize both benefits in explaining the scope of the course.

The only prerequisite for the course is a good knowledge of basic geometry and related mathematics, which most students acquire in high school and in any decent undergraduate program. It helps a great deal if the student has had a course in engineering graphics and CAD. Brief introductions to matrices, group theory, graph theory, and solids have been included in the appendixes, which supplement the main body of the book. Most students, at least in my class at Columbia, have no prior knowledge of group theory It helps to teach or review these topics in the class as they arise. I have deliberately omitted parametric curves and surfaces from the appendixes because there are several excellent textbooks that cover this topic—it is sufficient to spend a few hours of class time on that material.

This book is full of examples and figures, which is quite unusual for a book on theory. I find this to be the best way to explain this theory to my students and colleagues. The subject matter is so intuitive that illustrations become part of the thought process, and so it is a good practice to encourage this type of thinking. However, I have taken great care to ensure that this feature in no way detracts from the rigor of the theory. Proofs are given for a few important theorems. If a theorem is well known or obvious, as is the case in many instances, the reader is directed to a readily available reference where more details can be found.

This book owes much to nearly 15 years of close interaction and collaboration with several colleagues in industry, universities, and standards committees, as well as my students at Columbia. Andre Clement introduced me to the group theory work in France and Michael O'Connor showed me how to formalize this work using Lie groups. Herb Voelcker and Mike Pratt constantly and strongly encouraged the pursuit of mathematical theories on which industrial practices can be standardized. Alan Jones and Michael Leyton provided valuable feedback in different phases of preparation of this book. Colleagues at PDES Inc., ISO, and ASME kindly contributed their knowledge, time, and support in numerous meetings on CAD, dimensioning, and tolerancing. Finally, I want to thank the staff of Marcel Dekker, Inc., for their help in bringing this book to the market.

Vijay Srinivasan

Contents

1
Introduction

Dimensions are numerical values assigned to certain geometric parameters. They are expressed in units of distance or angle. In engineering drawings some dimensions are indicated explicitly on two-dimensional projected views using standardized notations. Three-dimensional geometric models in computer-aided design (CAD) systems carry dimensions internally as values assigned to certain variables. A theory of dimensioning is also a theory of parameterizing geometric models.

Standards for indicating dimensions on engineering drawings are issued in the United States by the American Society of Mechanical Engineers (ASME) and internationally by the International Organization for Standardization (ISO). However, they do not provide a theory of dimensioning. The designer is expected to learn the art of dimensioning from his peers and other sources, such as textbooks on drafting.

1.1 AN EARLIER THEORY OF DIMENSIONING

In the mid-1930s, popular engineering drafting textbooks in the United States offered a theory of dimensioning that has survived to date. It is sufficiently compact that it can be reproduced (from Carl Svensen, *Drafting for Engineers*) in a few pages, as follows. Dimensions are indicated by arrows in the accompanying figures. In the following narrative Svensen refers to himself as the author when discussing his earlier book *Essentials of Drafting*.

> **Elements of the Theory of Dimensioning.** The theory of dimensioning as developed by the author was originally published in *Essentials of Drafting* and is now finding its way into various courses and textbooks, where its importance is recognized by devoting a separate chapter to it instead of a set of "general rules" as was formerly done.
> The following statement is quoted from *Essentials of Drafting:* "Constructions can be separated into parts and these parts can then be divided into geometrical solids. Each of the solids can then be dimensioned and their relations to each other fixed." Thus, there are two kinds of dimensions:
>
> 1. Size dimensions
> 2. Location dimensions
>
> **The elementary cases of size dimensioning** include the common geometrical solids shown in Fig. 1.1, which may be termed *positive* or *negative*. The cases are conveniently classified as follows:

1. *The prism and modifications.* The rectangular prism requires three dimensions, two of which are given on one view and the third on one of the other views.
2. *The cylinder* requires two dimensions, diameter and length, both of which are given on one view.
3. *The cone* requires two dimensions, both of which are given on one view. The frustum of a cone requires three dimensions.
4. *The pyramid* may have dimensions on one or both views, depending on the shape of the base.
5. *The sphere* requires only one dimension, the diameter.
6. *Other solids* require dimensions as determined by their geometrical properties and the purposes for which they are used. Examples of the application of size dimensioning are illustrated in Fig. 1.2.

Location dimensions are used to fix the positions of elementary parts in relation to each other or the location of groups of parts in relation to axes, contact surfaces, or other references. Prisms are generally located with reference to surfaces, but axes may be used, or both axes and surfaces, according to the requirements of position, kind of prism, and the purpose which they serve. In Figure 1.3-I there are two basic or locating surfaces, which meet at *A,* from which locating dimensions are given to the surfaces of the prism, which meet at *B*. Cylinders are located by axes and bases. In Figure 1.3-II the machined surface of the cylinder and the axis meet at point *B,* which fixes the position of the cylinder by two dimensions from

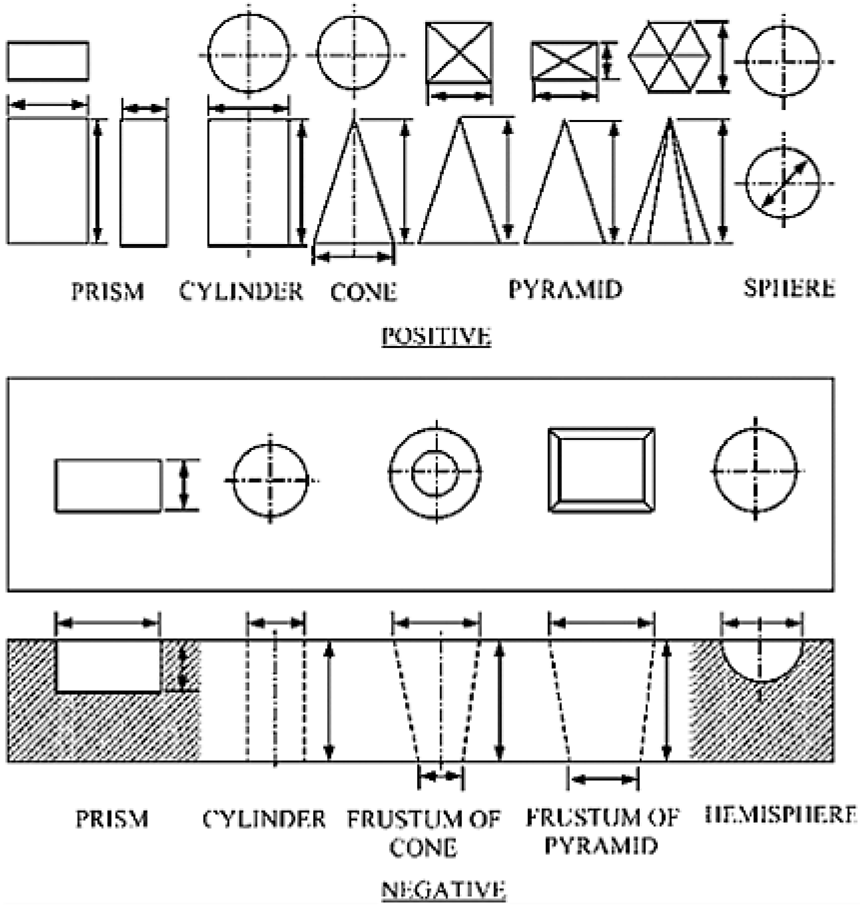


FIGURE 1.1 Elementary cases of size dimensioning. (From Svensen, 1935.)

the locating surfaces, which meet at *A*. Spheres are located by their centers. The location dimensions for other shapes and combination of shapes are dependent on geometrical properties and their relation to the whole object or one of its parts.

T**he elementary cases of location dimensioning** comprise *center-to-center* dimensions, *surface-to-center* (or *center-to-surface*) dimensions, and *surface-to-surface* dimensions (Fig. 1.4).

Procedures in Dimensioning. The four steps to be considered in applying the theory of dimensioning are:

1. Divide the object into elementary parts (type solids positive and negative).
2. Dimension each elementary part (size dimension).

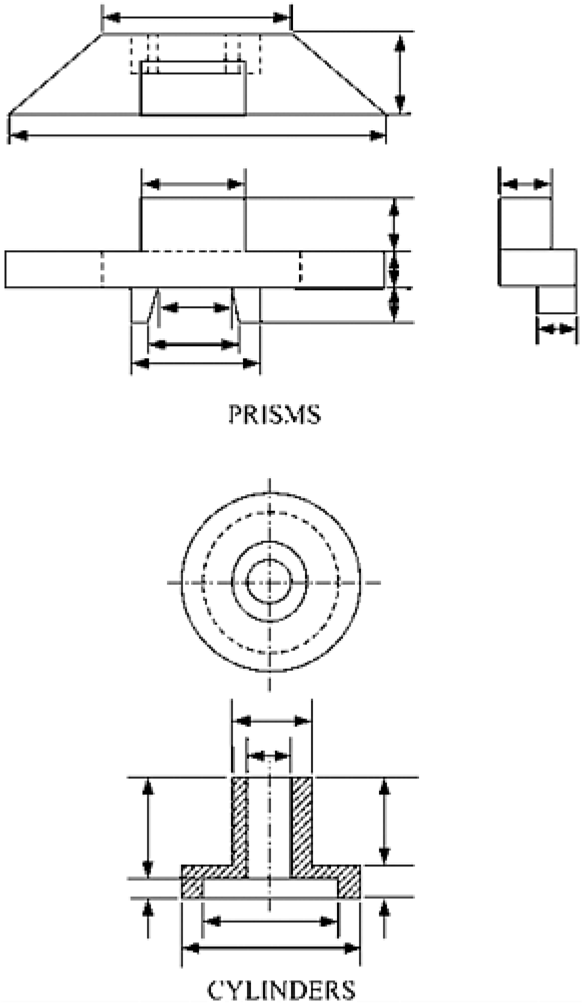


FIGURE 1.2 Size dimensions. (From Svensen, 1935.)

3. Determine locating axes and surfaces.
4. Locate the parts (location dimensions).

A few general remarks on the foregoing theory of dimensioning are worth making here. First, what Svensen calls “elementary parts” or “solids” will be called *features* in modern terminology. Note that these are not just surfaces; they carry additional information as to which side of the surface the material lies using the “positive” or “negative” attribute. His classification of these elementary parts for the purpose of size dimensioning into six cases is empirical—but he gets very close to a rigorous classification of geometrical objects presented in Chapter 6 on the basis of their symmetry.

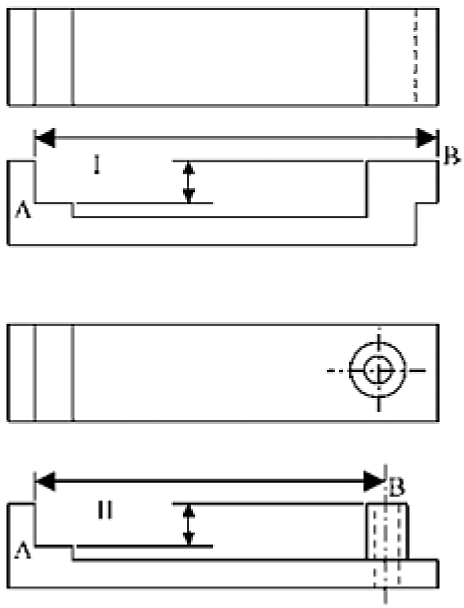


FIGURE 1.3 Location dimensions. (From Svensen, 1935.)

Second, his "location dimensions" fix the relative positions of geometrical objects. In other words, they are numerical values assigned to certain relative position parameters. Though he doesn't say it explicitly, his "size dimensions" are also numerical values, assigned to intrinsic shape parameters of his elementary parts or solids.

Third, he fumbles along in describing how to do location dimensioning. But he captures the essence when he says that centers, axes, and surfaces play lead roles in relative positioning. Also note that he talks about positioning not

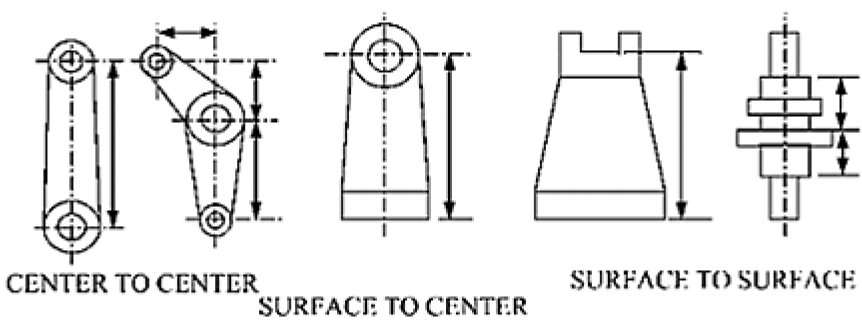


FIGURE 1.4 Cases of location dimensioning. (From Svensen, 1935.)

only features ("elementary parts") but also groups of features ("groups of parts") relative to each other. These notions will be formalized in Chapter 7.

These remarks, some of which are critical, should in no way diminish our appreciation for Svensen's contribution. Svensen should be given full credit for his remarkable engineering foresight. His work anticipated much of the later mathematical developments in dimensioning. In fact, this book formalizes and generalizes Svensen's work and presents a theory that supports all standardized practices of dimensioning in industry.

1.2 OUTLINE OF A MODERN THEORY OF DIMENSIONING

Size dimensions define the exact shape of a geometric object. They are the intrinsic characteristics of the shape, in the sense that these dimensions don't change when the geometric object is moved around. On the other hand, location dimensions define how two geometric objects are positioned relative to each other; these dimensions change when one object is moved relative to the other.

In the modern theory of dimensioning described in this book we abandon this simple, flat classification of dimensions into size and location dimensions. Instead, we impose a hierarchy on dimensions, as illustrated in Figure 1.5, because, as we will see in the rest of book, it is more powerful in building a dimensioning theory for any complex object. The dimensional taxonomy has two types of dimensions: *intrinsic* and *relational*. At the simplest level, elementary curves and surfaces have intrinsic dimensions—one may call them elementary size dimensions a la Svensen—that define the exact shapes of these curves and surfaces. For example, an unbounded (that is, of infinite length) cylinder has its radius (or diameter, as is normally done in engineering practice) as the intrinsic dimension because it doesn't change when the cylinder is moved around in space. It defines the exact shape of the cylinder no matter where it is positioned.

When two or more such elementary objects are considered, we can dimension the relative positioning of them—these are the relational dimensions. For example, if we consider two cylinders whose axes are parallel, then their relative positioning is determined solely by the distance between the axes. This distance is the relational dimension between the two cylinders, and changing it will change their relative positioning. However, when we consider a collection of two or more geometric objects, the relational dimensions among the objects become part of the intrinsic dimensions of the collection. This happens, for example, when we consider the two cylinders as holes drilled in the same block. This collection of the two parallel cylinders has three intrinsic dimensions: two

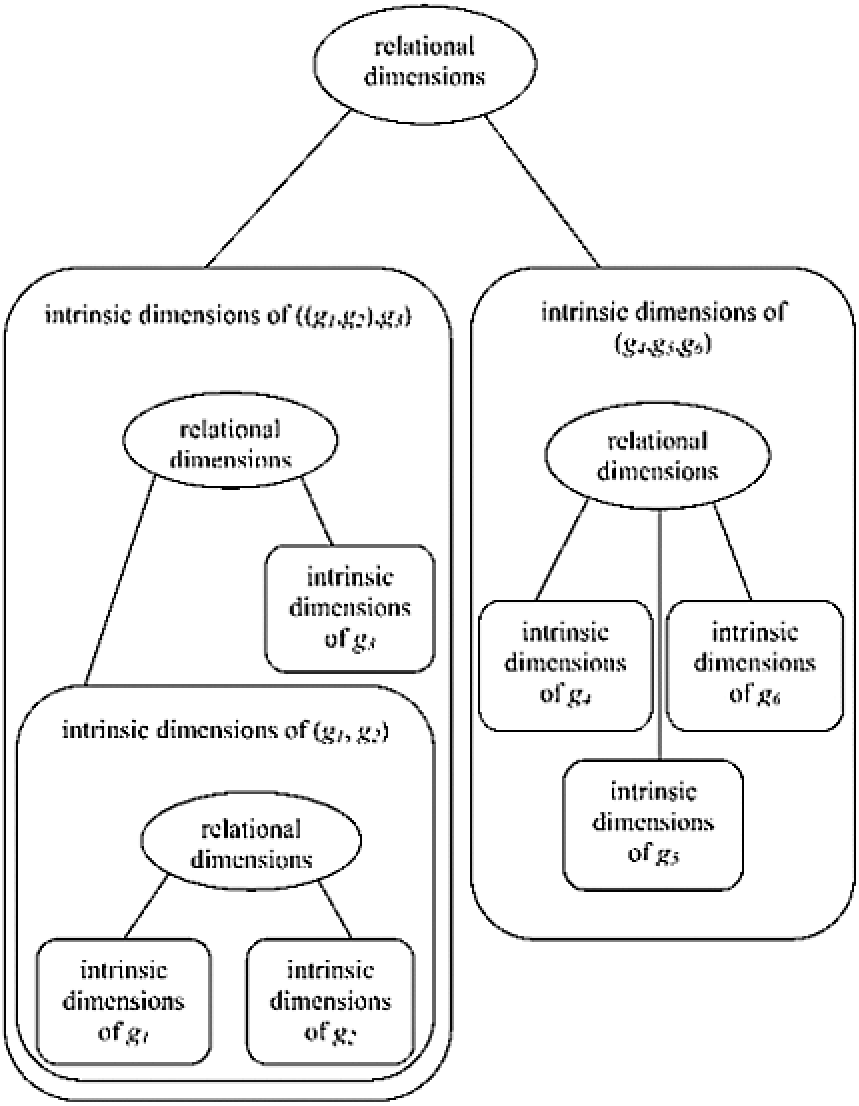


FIGURE 1.5 A dimenisonal taxonomy.

cylinder radii and one distance between their axes. As we move up the hierarchy, intrinsic dimensions subsume relational dimensions that appeared below.

Intrinsic dimensions are closely related to the concept of congruence under rigid motion. Loosely stated, two geometric objects are congruent if each can be moved by rigid motion so as to cover the other completely. Congruent geometric objects are identical copies that happen to have been positioned in different places in space. We say that geometric objects that have the same intrinsic dimensions must be congruent. Note that the converse need not be true, because the same geometric object can be dimensioned in multiple ways. (Just imagine how many ways a triangle can be dimensioned.)

The distinction between intrinsic and relational dimensions may seem artificial because it is purely a matter of the level of hierarchy under consideration. A cleaner, but theoretically equivalent, approach is one where dimensioning is defined recursively. Consider a geometric object *g* that is divided into two subobjects g_1 and g_2. If g_1 and g_2 have been dimensioned, then dimensioning the relative position of g_1 and g_2 completes the dimensioning of *g*. The recursion ends when each subobject is deemed to have been completely dimensioned (by some authority not yet defined) or is an elementary curve or surface that can be easily dimensioned using geometric theories (such as, for example, those presented in Chapters 3 and 4).

In either case, in general we define dimensions as those intrinsic characteristics of a geometric object that remain invariant under rigid motion of the geometric object. Because of this, it turns out that *the central question we pose in the modern theory of dimensioning is whether two given geometric objects are congruent under rigid motion*. We will devise procedures to answer this question and, in that process, come up with certain geometric parameters. Dimensioning is then just a task of assigning numerical values to these parameters. We start with a detailed treatment of the concept of congruence in Chapter 2. This is followed by a discussion of dimensioning elementary curves and surfaces in Chapters 3 and 4, respectively Chapter 5 is devoted to the seemingly simple task of dimensioning the relative positions of elementary objects such as points, lines, planes, and helices; it presents a special theory of relative positioning. Chapter 6 describes the notion of symmetry using group theoretic ideas, and it sets the stage for a general theory of dimensioning relative positions of arbitrary geometric objects in Chapter 7. So Chapter 7 can be considered as providing a general theory of relative positioning. Dimensional constraints are the topic of discussion in Chapter 8. Finally, the important topic of dimensioning solids is covered in Chapter 9.

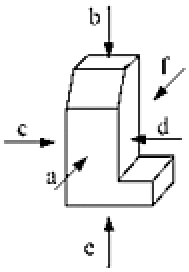
1.3 STANDARDIZED INDICATION OF DIMENSIONS

Since we will be dealing with dimensioning quite extensively, it is worth noting some stylized indications of dimensions standardized by the ASME and ISO. As we have already seen in Figures 1.1 through 1.4, dimensioned distances and angles are indicated by arrows. These are called *dimension lines,* which, with their arrowheads, show the direction and extent of a dimension. Dimension lines are often broken in the middle to show the numerical value (as, for example, in Figure 1.7). These are often, but not always, indicated on projected views of a part.

The ISO defines two alternative orthographic projection methods, as summarized in Table 1.1. Third angle projection is the preferred method in the

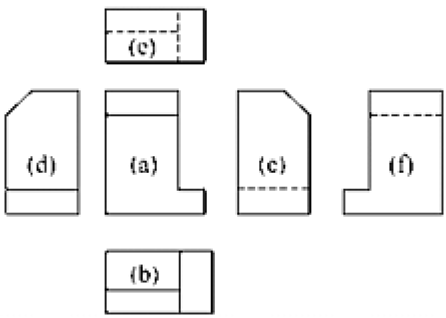
TABLE 1.1 ISO's definition of first and third angle projection methods. The hidden lines are shown dashed

Designation of Views



View in direction a=View from the front
View in direction b View from above
View in direction c—View from the left
View in direction d—View from the right
View in direction e—View from below
View in direction f—View from the rear

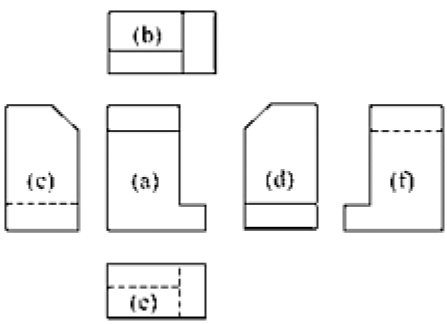
First Angle Projection Method



With reference to the front view (a), the other views are:

- The view from above (b), is placed underneath
- The view from below (e), is placed above
- The view from left (c), is place on the right
- The view from the right (d), is placed on the left
- The view from the rear (f) may be placed on the left, or on the right, as convenient

Third Angle Projection Method



With reference to the front view (a), the other views are:

- The view from above (b), is placed above
- The view from below (e), is placed underneath
- The view from left (c), is place on the left
- The view from the right (d), is placed on the right
- The view from the rear (f) may be placed on the left, or on the right, as convenient

FIGURE 1.6 Title block icons for first and third angle projections in engineering drawings.

United States, while the first angle projection method is used in many countries in the rest of the world. Because of this, it is customary to indicate an icon (in the form of a frustum of a cone, as shown in Figure 1.6) in the title block of every drawing so that we know which method of projection has been used in that print. It is also permissible to use projections along a direction of viewing different from these. In such cases, the direction of view will be indicated in the drawing.

Dimension lines are usually placed outside the outline of a part in a view. For this reason, *extension lines* (also known as *projection lines*) are used to indicate the extension of a surface or a point. Extension lines start with a short visible gap from the outline of the part and extend beyond the outermost related dimension line. Sometimes, a *leader line* is used to direct a dimension. Normally, a leader line will terminate in an arrowhead. If the leader line is used just to point to a feature, then it can terminate in a dot placed on the feature of interest. Figure 1.7 illustrates dimension lines, extension lines, and leader lines.

Standards also permit coordinate dimensioning. Examples of rectangular (Cartesian) coordinate dimensioning and polar coordinate dimensioning on two-dimensional projected views are shown in Figure 1.8. This methodology

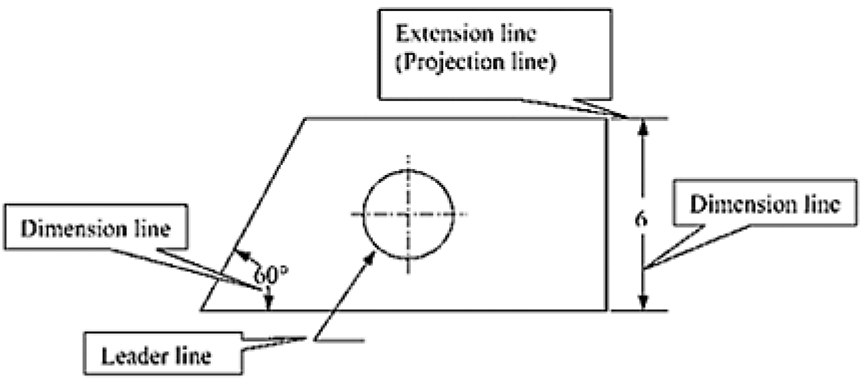


FIGURE 1.7 Illustration of dimension, extension, and leader lines.

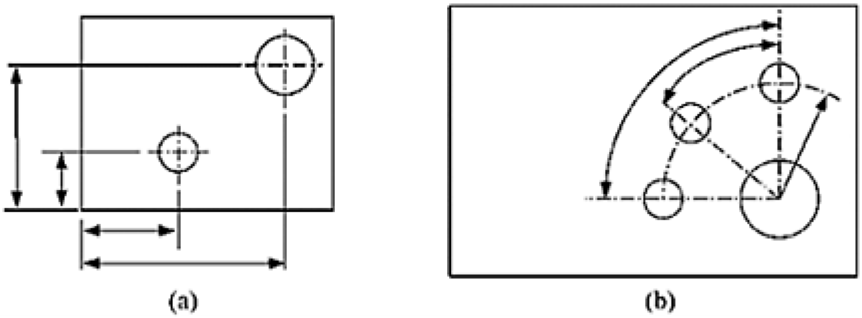


FIGURE 1.8 (a) Rectangular (Cartesian) coordinate dimensioning. (b) Polar coordinate dimensioning.

can be extended to three dimensions in the form of three-dimensional Cartesian coordinate dimensions, cylindrical polar coordinate dimensions, and spherical coordinate dimensions.

The last observation in the preceding paragraph anticipates the fact that dimensions can also be indicated on an isometric view of a part. Here, as before, a dimension shown between two extension lines is usually the distance not between the two parallel lines but between two parallel planes from which the extension lines emanate. This can often be a cause of confusion. The problem is completely avoided in a three-dimensional CAD model because it is view independent and the dimensions are carried as values for certain variables in the software.

1.4 EXERCISES

1. Sketch isometric views of the two solids shown in Figure 1.2. (*Hint:* These are third angle projections, following the American custom.) Try to dimension each solid completely in the isometric views. Record the assumptions made along the way. Classify some of these assumptions as constraints, such as incidence (that is, overlapping or coincidence), parallelism, and perpendicularity.
2. Dimension the same two parts in Figure 1.2 differently.
3. Dimensions (that is, the numerical values) are often changed during the course of a design. Figure 1.9 shows a part that has been (partially) dimensioned in two different ways. In Figure 1.9(a) the two dimensions are chained one after the other. (Hence the term *chain dimensioning*.)

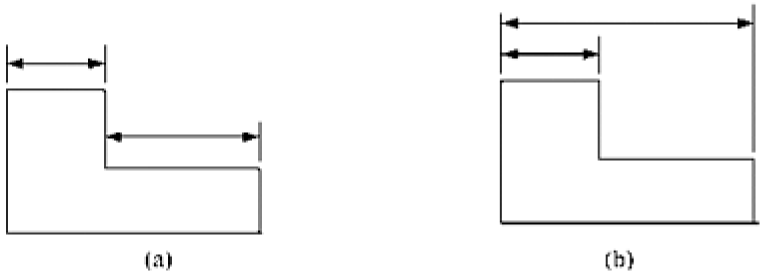


FIGURE 1.9 A part (partially) dimensioned in two different ways.

In Figure 1.9(b) these are defined in parallel, both originating from a common baseline, which is, actually, a base plane. (Hence the term *baseline dimensioning*.) Examine the effects of changing dimensions in each case. Give an example in each case to defend that way of dimensioning.

4. In industrial practice one often hears the terms *functional dimensions* and *nonfunctional dimensions*. What do you think is meant by these? Can you defend the dimensioning in Figures 1.9(a) and 1.9(b) as functional? (For the record: ASME merely stipulates that "dimensions shall be selected and arranged to suit the function and mating relationship of a part". The ISO formally defines that "a functional dimension is a dimension which is essential to the function of a part.")
5. What do you think is meant by *overdimensioning* and *underdimensioning*? Give simple examples of each. How would one determine that a part has been over- or underdimensioned? Can you think of some systematic (that is, algorithmic) way to accomplish this?

1.5 NOTES AND REFERENCES

An excellent history of engineering drawing can be found in Booker (1963). In the first half of the 20th century, manual drafting textbooks by French (1918) and Svensen (1935) were popular in the United States. A combined work of French and Svensen (1966) appeared in the second half of the century. Much of their manual methods were rendered obsolete by the arrival of computer-aided drafting systems in the late 1970s. What remain of interest to us are their philosophies and theories of dimensioning.

British Standard No. 308–1927 (1927) was one of the earliest national drafting standards. The first American Standard on drafting appeared in 1935 under the chairmanship of the aforementioned Thomas E.French. It was initiated by the ASME, which has since then been the driving force behind its further development. Revisions of this standard appeared at roughly 10-year intervals in 1946, 1957, 1966, 1973, 1982, and 1995. Over the years, the focus shifted gradually from purely drafting and dimensioning to geometric dimensioning and tolerancing (GD&T). The latest ASME national standard is called the ASME Y14.5M-1994 (1995) standard on "Dimensioning and Tolerancing" and is largely focused on tolerancing. Also in 1995, ASME published, for the first time, a mathematical companion called ASME Y14.5.1M-1994 that provides mathematical definition of tolerancing principles. Relevant ISO standards on dimensioning are ISO 128–1982 (1982), dealing with drawing layout, and ISO/R 129–1959 (1959), dealing with dimensioning rules.

2
Congruence

In high school plane geometry one learns how to determine if two triangles are congruent. Here are some theorems learned at school.

Theorem 2.1: Side-Angle-Side *If two sides and the included angle of one triangle equal, respectively, two sides and the included angle of another triangle, the two triangles are congruent.*

Theorem 2.2: Side-Side-Side *If the three sides of one triangle equal, respectively, the three sides of another triangle, the two triangles are congruent.*

Theorem 2.3: Angle-Side-Angle *If two angles and the included side of one triangle equal, respectively, two angles and the included side of another triangle, the two triangles are congruent.*

While learning these theorems the student goes through the mental process of taking one triangle and placing it on the other, vertex-on-vertex and side-on-side, so that one completely overlaps the other. This is called the *method of superposition*. In some cases this mental exercise could be carried out by moving the triangle without leaving the plane, as shown in Figure 2.1(a). But in

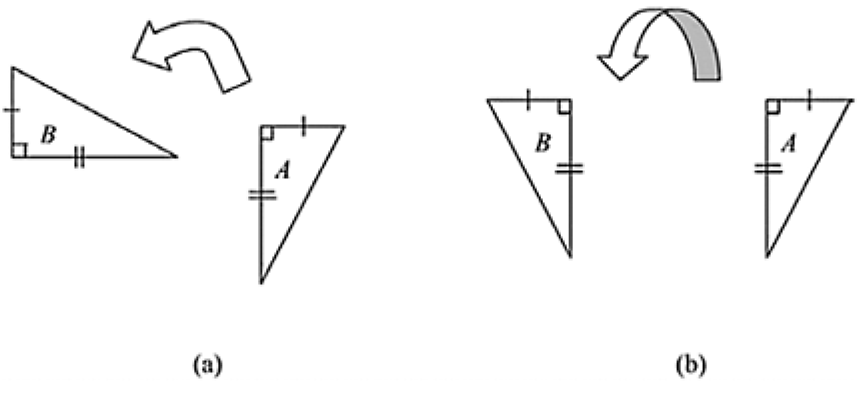


FIGURE 2.1 Congruent triangles *A* and *B* in the plane. (a) *A* can be moved to overlap *B* without leaving the plane. (b) *A* has to be lifted off the plane by flipping to overlap *B*.

most cases the student has to lift a triangle out of the plane by flipping it so that it can be placed on the other triangle; see Figure 2.1(b) for an example. In any case, Theorems 2.1, 2.2, and 2.3 teach us at least three ways to dimension a triangle: Assign numerical values (1) for two sides and the included angle, (2) for the three sides, or (3) for one side and the two adjacent angles. All these are intrinsic dimensions of the triangle.

This short encounter with triangle congruence portends a general theory of dimensioning. If congruence theorems such as these can be established for some class of geometric objects, then we can test whether two geometric objects from this class are congruent by comparing only a few distances and angles, which can be treated as parameters and given symbolic names. When numerical values are assigned to these parameters, they become the dimensions.

Since congruence plays such a prominent role in our theory of dimensioning, we will look at it in some detail in this chapter.

2.1 POINT-SETS AND TUPLES

Terms such as *shapes* and *geometric objects* are not precise enough for mathematical treatment. It is better to turn to sets of points, or *point-sets* for short, in Euclidean space. A point-set is an unordered collection of points that are symbolically included within curly brackets. For example, $S=\{p_1, p_2, \ldots, p_n\}$ is a point-set that is an unordered collection of n points. All results from classical set theory are applicable to point-sets. (Section A4.1 in Appendix 4 provides a brief primer on set theory, which the reader may want to review before proceeding further.) Two point-sets S_1 and S_2 are equal (denoted $S_1=S_2$) if every point in S_1 is also in S_2 and every point in S_2 is also in S_1.

> **Example 2.1** Let p_1, p_2, p_3, p_4, p_5 be five distinct points and let $S_1=\{p_1, p_2, p_3, p_4\}$, $S_2=\{p_2, p_1, p_4, p_3\}$, $S_3=\{p_1, p_3, p_5\}$ be point-sets. Then $S_1=S_2$ and $S_1 \neq S_3$. Note that the order of elements is not important.
>
> **Example 2.2** Point-sets can also be defined using formulas as in $S_4=\{(x, y): x^2+y^2=1\}$, $S_5=\{(x, y): x=\cos\theta, y=\sin\theta, 0 \leq \theta < 2\pi\}$, and $S_6=\{(x, y, z): x+y+z-1=0\}$. We interpret the indication for S_4 as a set of points in the plane with x-and y-coordinates that satisfy the equation $x^2+y^2=1$. Similar interpretations hold for S_5 and S_6. Note that $S_4=S_5$ because they are but different representations of the same unit circle in the plane centered at the origin. S_6 is a plane in space.

A set containing a finite number of elements is called a *finite set,* and a set containing an infinite number of elements is an *infinite set*.

> **Example 2.3** S_1, S_2, and S_3 in Example 2.1 are finite point-sets. S_4, S_5, and S_6 in Example 2.2 are infinite point-sets.

A point-set is *bounded* if there is sphere of finite radius that can contain it completely; otherwise, it is *unbounded*. Note that an infinite point-set can be bounded.

> **Example 2.4** In Examples 2.1 and 2.2, S_1, S_2, S_3, S_4, and S_5 are bounded point-sets. S_6 is an unbounded point-set.

A *tuple* is an ordered collection whose members are symbolically enclosed by parentheses. For example, $T=(p_1, p_2, \ldots, p_n)$ is a tuple of n points. A tuple can contain other tuples as members. Familiar examples of a tuple are the ordered pair (in two dimensions) and ordered triplets (in three dimensions) of coordinates to indicate a point. A tuple with n members is called an *n-tuple*. Two tuples are equal if and only if they have the same number of members and the members are equal when taken pairwise in order. More formally, we have the following definition.

Definition 2.1: Tuple Equality *$(S_1, S_2, \ldots, S_n)=(P_1, P_2, \ldots, P_n)$ if and only if $S_i=P_i$ for all i.*

Example 2.5 Let p_1, p_2, p_3, p_4 be four distinct points and let $T_1=(p_1, p_2, p_3, p_4)$ and $T_2=(p_2, p_1, p_3, p_4)$ be two 4-tuples. $T_1 \neq T_2$ because the members are not equal when taken in order.

Example 2.6 With the point-sets defined in Examples 2.1 and 2.2, we have $(S_1, S_4)=(S_2, S_5)$.

2.2 RIGID MOTION

Rigid motion is a particular type of transformation of points in Euclidean space. The concept of rigid motion is central to our study of congruence. It is best represented using matrices. Let's start with a right-handed, orthogonal coordinate system, where the coordinate axes are labeled *x, y,* and *z*. Consider a transformation in which a point p with coordinates (x, y, z) is transformed to a point p' with coordinates (x', y', z') by the matrix operation

$$\begin{Bmatrix} x' \\ y' \\ z' \end{Bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \end{Bmatrix} + \begin{Bmatrix} x_0 \\ y_0 \\ z_0 \end{Bmatrix} \tag{2.1}$$

where $[x_0, y_0, z_0]^T$ is a translation vector and the coefficients a_{ij} in the 3×3 matrix A are real. Appendix 1 gives a brief review of matrices, which should be read along with this chapter. Matrix A is called the *rotation matrix,* and transformation (2.1) is called *rigid motion* (also known as *solid displacement*) when A is *orthogonal* and its determinant is+1. Let's look at this statement in some detail. Several properties of orthogonal matrices are well known and are quoted here.

When A is orthogonal, $A^TA=I$. That is,

$$\begin{bmatrix} a_{11} & a_{21} & a_{31} \\ a_{12} & a_{22} & a_{32} \\ a_{13} & a_{23} & a_{33} \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

This means that the nine coefficients a_{ij} cannot be chosen arbitrarily. There are nine equations involving the coefficients that result from Eq. (2.2). These are:

$$a_{11}^2 + a_{21}^2 + a_{31}^2 = 1, \quad a_{12}^2 + a_{22}^2 + a_{32}^2 = 1, \quad a_{13}^2 + a_{23}^2 + a_{33}^2 = 1$$
$$a_{11}a_{12} + a_{21}a_{22} + a_{31}a_{32} = 0, \quad a_{11}a_{13} + a_{21}a_{23} + a_{31}a_{33} = 0$$
$$a_{12}a_{13} + a_{22}a_{23} + a_{32}a_{33} = 0, \quad a_{12}a_{11} + a_{22}a_{21} + a_{32}a_{31} = 0$$
$$a_{13}a_{11} + a_{23}a_{21} + a_{33}a_{31} = 0, \quad a_{13}a_{12} + a_{23}a_{22} + a_{33}a_{32} = 0 \qquad (2.3)$$

Of these nine equations, the last three are duplicates of the previous three. So only six independent equations remain for nine coefficients a_{ij}. Therefore, an orthogonal matrix *A* can have only three independent parameters.

An important consequence of the orthogonality of *A* is that distances are preserved under transformation (2.1). Recall that in classical mechanics a rigid body is defined as a point-set in which the distance between any two points remains invariant when the body is subjected to motion. If p_1 and p_2 are two points with coordinates (x_1, y_1, z_1) and (x_2, y_2, z_2), respectively, then the Euclidean distance between them is denoted and given by

$$d(p_1, p_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \qquad (2.4)$$

If matrix *A* is orthogonal and transformation (2.1) is applied to both p_1 and p_2, transforming them to p'_1 and p'_2, respectively, then it can be shown that $d(p_1, p_2) = d(p'_1, p'_2)$. In addition, it can be shown that angles are also preserved under this transformation. Generally, when we talk about applying transformation (2.1) to a geometric object we mean that every point in the point-set that describes the geometric object is subjected to the same transformation and that the entire point-set is transformed to another point-set. It is now easy to see the connection between the rigid body of classical mechanics and the rigid motion applied to a point-set.

The determinant of an orthogonal matrix *A* can only be +1 or −1. In either case, distances are preserved under transformation (2.1); for this reason, such a transformation is called *isometry*. When the determinant of orthogonal matrix *A* is +1, transformation (2.1) is a rigid motion consisting only of translation and rotation; *A* then becomes a rotation matrix, and it requires only three independent parameters for its definition. The translation vector in Eq. (2.1) requires three independent parameters as well. Hence a rigid motion—we say, in fact, a rigid body—has six degrees of freedom: three for translation and three for rotation.

Example 2.7 In Eq. (2.1), let the translation vector be zero and

$$A = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (2.5)$$

A is orthogonal, as can be easily verified, and its determinant is −1. The transformation is a reflection about the *yz*-plane. That is, the transformation merely reverses the sign of the *x*-coordinate of every point. This is an isometric transformation because it preserves distances. But it is not a rigid motion. Figure 2.2 shows how this transformation acts on a tetrahedron.

E**xample 2.8** In Eq. (2.1), let the translation vector be zero and

$$A = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{2.6}$$

A is orthogonal and its determinant is +1. This transformation is a rigid motion. It applies a rotation of 180° about the *y*-axis to every point on which it acts. Figure 2.3 shows its action on a tetrahedron.

E**xample 2.9** In two-dimensional cases, Eq. (2.1) reduces to

$$\begin{Bmatrix} x' \\ y' \end{Bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{Bmatrix} x \\ y \end{Bmatrix} + \begin{Bmatrix} x_0 \\ y_0 \end{Bmatrix} \tag{2.7}$$

In the *xy*-plane if we let the translation vector be zero and

$$A = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \tag{2.8}$$

then we have a reflection about the *y*-axis. *A* has a determinant of −1. Again, this is an isometry but not a rigid motion. The transformation of triangles in Figure 2.1(b) can be achieved using

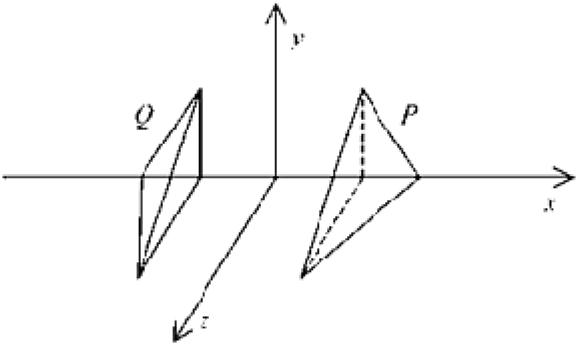


FIGURE 2.2 Reflection about the *yz*-plane transforms tetrahedron *P* to tetrahedron *Q*. This is an isometry but not a rigid motion. Unlike the triangles in Figure 2.1, *P* cannot be moved in three-dimensional Euclidean space to overlap *Q* by rigid motion. Therefore, *P* and *Q* are congruent under isometry but not under rigid motion.

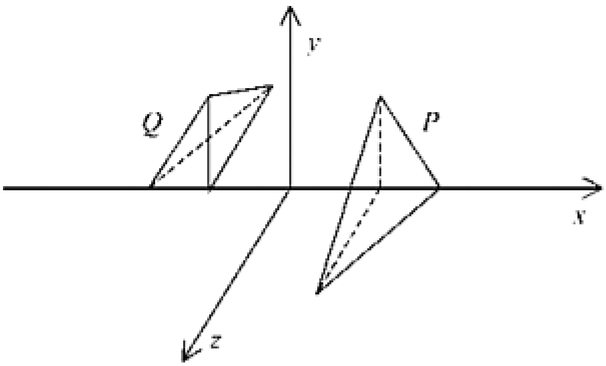


FIGURE 2.3 Rotation of 180° about the *y*-axis transforms tetrahedron *P* to tetrahedron *Q*. This is a rigid motion. *P* and *Q* are congruent under rigid motion.

Eq. (2.8). But if the *xy*-plane is then embedded in a three-dimensional space, then the same transformation of triangles shown in Figure 2.1(b) can be achieved via matrix *A* given by Eq. (2.6), which is a rigid motion. This example illustrates that a reflection in a two-dimensional plane can be achieved by a rigid motion if the plane is embedded in a three-dimensional space.

Equation (2.1) can be rewritten more compactly as

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & x_0 \\ a_{21} & a_{22} & a_{23} & y_0 \\ a_{31} & a_{32} & a_{33} & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.9}$$

or simply as

$$X' = RX \tag{2.10}$$

where X and X' are points represented in homogeneous coordinates and R is a rigid motion represented by a special 4×4 matrix whose elements come from rotation matrix A and translation vector $[x_0, y_0, z_0]^T$. For computational purposes, it is useful to remember that matrix R representing the rigid motion is composed of submatrices of the following form:

$$R = \begin{bmatrix} A & X_0 \\ 0_3 & 1 \end{bmatrix} \tag{2.11}$$

where A is the rotation matrix, X_0 is the translation vector, and 0_3 is the null row vector

[0, 0, 0].

Representing an entire rigid motion by a single matrix has some conceptual advantages as well. If two rigid motions are applied in sequence, then the result can be obtained by simply multiplying the two matrices that represent them, as in

$$RR' = \begin{bmatrix} A & X_0 \\ 0_3 & 1 \end{bmatrix} \begin{bmatrix} A' & X_0' \\ 0_3 & 1 \end{bmatrix} = \begin{bmatrix} AA' & AX_0' + X_0 \\ 0_3 & 1 \end{bmatrix} \tag{2.12}$$

firmly keeping in mind that the rigid motion represented by R' is applied before the rigid motion represented by R. If we want to undo a rigid motion—that is, reverse the result of a rigid motion—we just take the inverse of the matrix and premultiply it. The inverse of R is easily computed as

$$R^{-1} = \begin{bmatrix} A & X_0 \\ 0_3 & 1 \end{bmatrix}^{-1} = \begin{bmatrix} A^T & -A^T X_0 \\ 0_3 & 1 \end{bmatrix} \tag{2.13}$$

where we have exploited the orthogonality (that is, its inverse is its transpose) of the rotation matrix A. If R is chosen as the 4×4 identity matrix, it doesn't move the point(s) at all. It is called the *identity rigid motion*. We will exploit this strong connection between rigid motion and its matrix representation throughout our study.

As an operation on point-sets, rigid motion can be given a symbolic representation that is motivated by the matrix form of Eq. (2.10). If we denote a rigid motion by r, then the result of applying it on a point-set S is indicated by $r(S)$, or rS for short, which is also a point-set.

Example 2.10 Let S_1 be a unit sphere centered at the origin. It can be represented by the point-set $S_1=\{(x, y, z):x^2+y^2+z^2=1\}$. Now, represent by r a rigid motion consisting only of unit translation along the x-axis. Then rS_1 is the point-set that results from the application of the rigid motion r on S_1, and it can be represented by $\{(x, y, z):x^2-2x+y^2+z^2=0\}$. To establish this, first observe that the rigid motion transformation is given by $x'=x+1$, $y'=y$, and $z'=z$. The corresponding rigid motion matrix is

$$R = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Because of the simple nature of this transformation, it is easy to see that the inverse transformation is given by $x=x'-1$, $y=y'$, and $z=z'$. We can also see this by inverting the matrix R using Eq. (2.13), which yields

$$R^{-1} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Substituting these in the defining equation of S_1 we get

$$x^2 + y^2 + z^2 = 1 \Rightarrow (x' - 1)^2 + (y')^2 + (z')^2 = 1$$
$$\Rightarrow (x')^2 - 2x' + (y')^2 + (z')^2 = 0$$

Therefore rS_1 can be represented by the point-set $\{(x', y', z'): (x')^2-2x'+(y')^2+(z')^2=0\}$. Since the variables involved within the curly brackets are dummy variables, it is valid to say that $rS_1=\{(x, y, z):x^2-2x+y^2+z^2=0\}$.

Extending the notion of rigid motion to tuples, we have the following definition.

Definition 2.2: Tuple Rigid Motion $r(S_1, S_2, \ldots, S_n)=(rS_1, rS_2, \ldots, rS_n)$.

That is, a rigid motion applied to a tuple is the tuple of rigid motions applied to its members.

With this definition a mechanical model for a tuple of geometric objects can be offered. Imagine that the members of a tuple are rigidly welded together by an invisible welding material. When one member is subjected to a rigid motion, all of them move by the same rigid motion. Since a tuple can contain other tuples as members, Definitions 2.1 and 2.2 are recursive.

Example 2.11 In the plane consider the following points defined by their *x*- and *y*-coordinates:

$p_1=(2, 1), p_2=(-2, 1), p_3=(-2, -1), p_4=(2, -1)$

Let P be the 4-tuple (p_1, p_2, p_3, p_4) and apply a rigid motion r consisting of a unit shift along the *x*-axis and a unit shift along the *y*-axis. Then rP is the 4-tuple (q_1, q_2, q_3, q_4), where

$q_1=(3, 2), q_2=(-1, 2), q_3=(-1, 0), q_4=(3, 0)$

Example 2.12 In space let p_1 be the point (0, 0, 0) and S_1 be the surface $\{(x, y, z):x^2+y^2+z^2=1\}$. Now consider the 2-tuple (p_1, S_1) and apply to it a rigid motion r consisting of just a unit shift along the *x*-axis. The result is the 2-tuple $r(p_1, S_1)=(p_2, S_2)$, where $p_2=(1, 0, 0)$ and $S_2=\{(x, y, z):x^2-2x+y^2+z^2=0\}$. See Example 2.10 for the justification of the representation of S_2.

2.3 REFLEXIVE SYMMETRY AND CHIRALITY

The concept of symmetry is very important and will be developed in detail in Chapter 6. Here we briefly discuss a particular type of symmetry related to reflection encountered in Example 2.7.

Two points p_1 and p_2 are said to be positioned symmetrically with respect to a plane or line if and only if this plane or line bisects the line segment between p_1 and p_2 perpendicularly. These points are positioned symmetrically with respect to a point if and only if this point is the midpoint of the line segment between p_1 and p_2. In all three cases, each of the points p_1 and p_2 is said to be symmetrical to the other with respect to the plane, line, or point.

The notion of symmetry can be extended beyond two points to a point-set. A point-set S is symmetrical with respect to a plane (or line or point) if and only if for every point p in S we can find a point q in S such that p and q are symmetrically positioned with respect to the plane (or line or point). We can then say that the point-set S has a plane of symmetry, a line of symmetry (or axis), or a point of symmetry (or center).

Example 2.13

1. A sphere is symmetrical with respect to its center. It is also symmetrical with respect to any plane or line through its center.
2. A baseball bat has an axis of symmetry, and any plane through this axis is a plane of symmetry. It doesn't have a point of symmetry.
3. The frame of a tennis racket has two perpendicular planes of symmetry, intersecting at an axis of symmetry. It has no center of symmetry.

Example 2.14 In the plane the curve $C=\{(x, y):y^2=2x\}$ defines a parabola. The x-axis is the only line of reflexive symmetry for the parabola, so it is the unique axis of the parabola. The parabola has no point of symmetry, that is, no center. (Sketch the parabola to verify these results.)

Example 2.15 In the plane the curve $C=\{(x, y):x^2+2y^2=1\}$ defines an ellipse. Both the x- and y-axes are lines of reflexive symmetry for the ellipse. Hence these are the axes of the ellipse. The origin (0, 0) is the unique point of symmetry, that is, the center. (Sketch the ellipse to verify these results.)

Example 2.16 In space the surface $S=\{(x, y, z):x^2+ 2y^2+3z^2=1\}$ defines an ellipsoid. It is symmetrical with respect the xy-, yz-, and zx-planes. The x-, y-, and z-axes are the axes of the ellipsoid. The origin (0, 0, 0) is the unique point of symmetry, that is, the center. (Sketch the ellipsoid to verify these results.)

Closely related to reflexive symmetry is the concept of *chirality*. It is best defined by the original words of Lord Kelvin: "I call any geometrical figure, or group of points, chiral, and say it has chirality, if its image in a plane mirror, ideally realized, cannot be brought to coincide with itself." By this definition, tetrahedron P in Figure 2.2 is chiral and has chirality; the same can be said of tetrahedron Q. If the definition is specialized to the plane (meaning that we are not allowed to move out of the plane), then we see that the triangle A is chiral in the plane of Figure 2.1. Other important examples follow.

Example 2.17 A helix is a space curve. One such helix can be represented by the point-set $H_1=\{(x, y, z): x=2\cos\theta, y=2 \sin\theta, z=0.5\theta/(2\pi)\}$. It has a base-cylinder radius of 2 units and a *pitch,* which is defined as axial advance per revolution, of 0.5 units. A mirror reflection of H_1 is given by the point-set $H_2=\{(x, y, z): x=2 \cos\theta, y=2 \sin \theta, z=-0.5\theta/(2\pi)\}$, where the xy-plane acts as the reflection plane. That is, H_2 is obtained by simply reversing the sign of the z-coordinate in the definition of H_1. We call H_1 a right-handed helix and H_2 a left-handed helix. It can be shown that H_1 and H_2 are not congruent under rigid motion. In general, left-and right-handed helices are not congruent. Hence a helix is chiral.

Example 2.18 A three-dimensional right-handed, orthogonal Cartesian coordinate system is chiral. This is due to the facts that its mirror image is a left-handed, orthogonal coordinate system and that these two are not congruent under rigid motion. This is an example where a tuple is given chirality.

A point-set that is not chiral is called *achiral*. The parabola and ellipse in Examples 2.14 and 2.15, respectively, are achiral in the plane. They are achiral even when the plane that contains them is embedded in space. The ellipsoid of Example 2.16 is also achiral.

Often we will use the term *chirality* synonymously with *handedness*. It is important to make sure that this use causes no confusion. It is proper to say that the right-handed and left-handed orthogonal Cartesian coordinate systems do not have the same chirality, even though both are chiral. Similarly, tetrahedra P and Q in Figure 2.2 do not have the same chirality, and helices H_1 and H_2 in Example 2.17 do not have the same chirality, though all these objects are chiral.

2.4 VECTORS AND ORIENTED POINT-SETS

Sometimes it is useful to associate an orientation to a point-set. We may want to take a straight line and give it an orientation, as we did with each of the x-, y-, z-coordinates axes. Orientation can be represented by vectors of the form

$$\mathrm{v}^T=[v_x, v_y, v_z] \tag{2.14}$$

These vectors have magnitude and direction. The magnitude of the vector is given by

$$|v| = \sqrt{v_x^2 + v_y^2 + v_z^2} \tag{2.15}$$

and a unit vector is defined as

$$\hat{v} = \frac{v}{|v|} \tag{2.16}$$

Like points, vectors can also be transformed by rigid motion. Unlike points, vectors are affected only by rotation and not by translation, because we care only about the orientation of these vectors. So we can transform a vector using the rotation matrix as

$$v' = Av \tag{2.17}$$

Two types of products can be defined for vectors. The first is a *dot product,* defined as

$$v_1 \cdot v_2 = v_{1x}v_{2x} + v_{1y}v_{2y} + v_{1z}v_{2z} \tag{2.18}$$

The dot product, also known as the *inner product* of the vectors, is a scalar. If θ is the angle between the two vectors, then the dot product is equal to $|v_1||v_2|\cos\theta$. When the two vectors are perpendicular to each other, the dot product vanishes. The second product defined between two vectors is a *cross product*. It is a vector given by

$$v_1 \times v_2 = [v_{1y}v_{2z} - v_{1z}v_{2y},\ v_{1z}v_{2x} - v_{1x}v_{2z},\ v_{1x}v_{2y} - v_{1y}v_{2x}]^T \tag{2.19}$$

Its magnitude is equal to $|v_1||v_2|\sin\theta$, which means that it vanishes if the two vectors are oriented in the same or the opposite direction. When this is not the case, the two vectors can be brought to lie in a plane, and the direction of the cross product is perpendicular to this plane. The orientation of the cross product is determined by the right-hand rule: When we curl our right-hand fingers from vector v_1 to v_2, the extended thumb is pointing toward the orientation of $v_1 \times v_2$.

If we denote the unit vectors along the *x*-, *y*-, and *z*-axes in a right-handed, orthogonal Cartesian coordinate system by *i, j,* and *k,* respectively, then $k=i\times j$ and the 3-tuple *(i, j, k)* is called a right-handed *triad* or, simply, a *trihedron*. If *k* is chosen such that $k=-i\times j$, then we have a left-handed triad.

An important property of the rigid motion is that it preserves both the dot product and the cross product of vectors. This means that under the action of rotation matrix *A,* with its determinant equaling+1, we have

$$Av_1 \cdot Av_2 = v_1 \cdot v_2 \text{ and } Av_1 \times Av_2 = A(v_1 \times v_2) \tag{2.20}$$

If *A* is a reflection, as in Example 2.7, then it need not preserve the cross product; that is, the orientation can be reversed under reflection. So under rigid motion a right-handed triad is moved to another right-handed triad, whereas a reflection with respect to a plane transforms a right-handed triad to a left-handed triad. As we saw earlier in Example 2.18, this is the cause of chirality of the coordinate systems.

Straight lines and planes can be oriented by assigning a vector parallel to the straight line and one perpendicular to the plane, respectively. Although individual oriented lines in plane and space and oriented planes in space are achiral, we can construct tuples of them that have chirality.

Example 2.19 Consider a 3-tuple of three mutually orthogonal, oriented lines in space. These oriented lines can be the *x*-, *y*-, and *z*-axes of an orthogonal coordinate system, and the tuple has chirality as we have already seen.
Example 2.20 A 3-tuple consisting of an oriented plane in space and two nonparallel, oriented lines in that plane is chiral and has chirality.

2.5 CONGRUENCE, DIMENSIONS, AND PARAMETERS

Two geometric objects are congruent under rigid motion if one can be transformed by rigid motion to the other. There is also a notion of congruence under isometry, as illustrated in Figure 2.2 for a simple example involving tetrahedra, but for engineering applications involving dimensioning we will stick to congruence under rigid motion. The reason for this is the necessity to distinguish between parts that are interchangeable and those that are not. For example, the left- and right-hand gloves are mirror images of each other and they are congruent under isometry. But they are not interchangeable parts and, in industrial parlance, have different part numbers. We could, however, distinguish them by observing that these gloves are not congruent under rigid motion. From now on, when we mention congruence we mean congruence under rigid motion.

Generically, we have the following definition.

Definition 2.3 *Two point-sets S_1 and S_2 are congruent if and only if there exists a rigid motion r such that $rS_1=S_2$. Two tuples T_1 and T_2 are congruent if and only if there exists a rigid motion r such that $rT_1=T_2$.*

Let’s try to apply this definition directly to prove some congruence based on some of the examples examined before.

Example 2.21 The two surfaces defined by point-sets

$S_1=\{(x, y, z):x^2+y^2+z^2=1\}$ and
$S_2=\{(x, y, z):x^2-2x+y^2+z^2=0\}$

are congruent because, as we saw in Example 2.10, there exists a rigid motion *r* consisting of just a unit translation along the *x*-axis such that $rS_1=S_2$.
Example 2.22 In the plane consider the following points defined by their *x*- and *y*-coordinates:

$p_1=(2, 1), p_2=(-2, 1), p_3=(-2, -1), p_4=(2, -1)$
$q_1=(3, 2), q_2=(-1, 2), q_3=(-1, 0), q_4=(3, 0)$

1. The 2-tuple (p_1, p_2) is congruent to the 2-tuple (p_3, p_4). It is also congruent to the 2-tuple (q_1, q_2). (Why?)
2. The 2-tuple (p_1, p_2) is not congruent to the 2-tuple (q_1, q_3). (Why?)
3. The 4-tuple (p_1, p_2, p_3, p_4) is congruent to the 4-tuple (q_1, q_2, q_3, q_4) because there is a simple translation—a unit shift along the x-axis and a unit shift along the y-axis—that moves one of the tuples to coincide with the other, as we saw in Example 2.11.
4. However, the 4-tuple (p_1, p_2, p_4, p_3) is not congruent to the 4-tuple (q_1, q_2, q_3, q_4) because, however hard we try, we will not be able to come up with a rigid motion r such that $r(p_1, p_2, p_4, p_3)=(q_1, q_2, q_3, q_4)$.

Example 2.23 In space consider the following points defined by their coordinates

$p_1=(0, 0, 0)$ and $p_2=(1, 1, 1)$

and two surfaces defined by

$S_1=\{(x, y, z):x^2+y^2+z^2=1\}$ and
$S_2=\{(x, y, z):x^2-2x+y^2+z^2=0\}$

As we saw in Example 2.21, S_1 and S_2 are congruent. But the 2-tuple (p_1, S_1) is not congruent to the 2-tuple (p_2, S_2) because, however hard we try, we will not be able to find a rigid motion r such that $r(p_1, S_1)=(p_2, S_2)$.

As the preceding examples illustrate, Definition 2.3 is an existential definition, in the sense that it requires the existence of a rigid motion r for congruence without describing how such a rigid motion can or cannot be found. Proving the existence or otherwise of the rigid motion r is left as a tricky problem to be solved by the person asking the question about the congruence. Instead of searching for the elusive rigid motion, we will seek specific congruence theorems of the following form.

Congruence Theorem Template *If two* ⟨*geometric objects that belong to a class*⟩ *have the same* ⟨*chirality and distance and angle measures*⟩, *then they are congruent.*

The triangle congruence theorems 2.1, 2.2, and 2.3 are examples of such specific congruence theorems. Strictly speaking, in a two-dimensional plane these triangle theorems are congruence theorems under isometry. But we will embed the plane in three-dimensional space in which these theorems are congruence theorems under rigid motion, as demonstrated in Example 2.9. In specific congruence theorems, the distance and angle measures appear as variables—such as side lengths and included angles—which we call parameters. If there are n parameters, the geometric object under consideration is said to belong to an n-parameter family. For example, a triangle belongs to a 3-parameter family. Congruence is established if these parameters assume the same values in the two geometric objects. Dimensions are then just the numerical values assigned to these parameters.

So the congruence theorems state that two point-sets that belong to a particular class are congruent if they have the same dimensions. The theorems also tell us what these dimensions are. Our interest in congruence theorems should now be obvious. As the triangle congruence theorems demonstrate, the same point-set can be parameterized and, hence, dimensioned in multiple ways.

Finally, a word of caution about the meaning of the term *dimension* is in order. Sometimes we use the word *dimension* to refer to the dimensionality of space we are dealing with. A two-dimensional plane and three-dimensional space are examples of this usage. Later, we will also refer to higher-dimensional spaces, such as a six-dimensional space, consisting of translations and rotations. It is also common to talk about the dimensionality of a parametric space when referring to a parameterization of a geometric object. However, as we have observed earlier, we also use the term *dimension* to refer to a numerical value for a geometric parameter. The context should make it very clear as to what we mean by dimension. Fortunately, there is no ambiguity about the meaning of *dimensioning* in engineering; it is the act of specifying dimensions, that is, numerical values to certain geometric parameters.

2.6 EXERCISES

1. Assume that you live in a two-dimensional planar world and are not allowed to move out of this plane. Restate the three triangle congruence theorems (2.1, 2.2 and 2.3) for this world. How would you define chirality in this world? How would you dimension a triangle in this world?
2. Specialize Eqs. (2.9) through (2.13) for the case of rigid motions in a two-dimensional plane. Homogeneous coordinates for a point in the plane are given by the vector

$$\begin{Bmatrix} x \\ y \\ 1 \end{Bmatrix}$$

3. In each of the following cases, give an example of a point-set that

- Has a plane of symmetry, but no axis or center.
- Has an axis, but no plane of symmetry or center.
- Has a center, but no plane of symmetry or axis.

4. Prove the following assertions:

- If a point-set *S* has two planes of symmetry, then their intersection is an axis of *S*.
- If a point-set *S* has two perpendicular, intersecting axes, then a line perpendicular to both these axes and passing through their intersection is also an axis of *S*.
- If a point-set *S* has a center that lies on an axis of *S*, then the plane perpendicular to the axis and passing through the center is a plane of symmetry of *S*.

5. A triangle is dimensioned as shown in Figure 2.4(a). Is the dimensioning valid, that is, does this define a unique triangle up to rigid motion? If so, what is the associated congruence theorem?

- Another triangle is dimensioned as shown in Figure 2.4(b). Try drawing it with a ruler, a compass, and a protractor. Is the dimensioning valid? What is the associated congruence theorem?

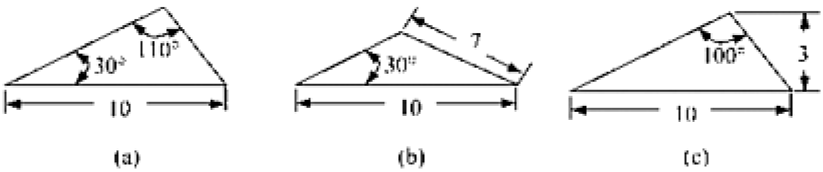


FIGURE 2.4 Three examples of dimensioned triangles. Are they valid?

- Yet another triangle is dimensioned as shown in Figure 2.4(c). Try drawing it with a ruler, a compass, and a protractor. Is the dimensioning valid? What is the associated congruence theorem? Is it chiral in the plane of the triangle?

6. When are two planar quadrilaterals congruent? Dimension the simple planar quadrilateral shown in Figure 2.5(a) using your result. What happens when the quadrilateral is not simple, that is, there is an edge intersecting a nonadjacent edge, as in Figure 2.5(b).
7. When are two planar polygons congruent? Dimension a planar polygon using your result. Consider both simple and nonsimple polygons.
8. "All planar objects are achiral when the plane that contains them is embedded in space." What do you think is meant by this statement? Is it true, or are there any qualifications/exceptions to it?
9. Let $S=\{(x, y, z):x^2+2y^2+3z^2=1\}$ be the point-set that represents an ellipsoid. What is the point-set that results from applying a rigid motion of 90° rotation about the z-axis and a translation of unit shift along the x-axis to S?
10. When are two tetrahedra congruent? Note that chirality is important here. Dimension a tetrahedron using your result.
11. When are two polyhedra congruent? Dimension a polyhedron using your result.

2.7 NOTES AND REFERENCES

The three triangle congruence theorems, 2.1, 2.2 and 2.3, appear in Euclid's *Elements*. Joyce (http://aleph0.clarku.edu/~djoyce/java/elements/elements.html) maintains an excellent online version of the *Elements* with interactive graphics. The Side-Angle-Side theorem (Book I, Proposition 4) appears very early in the *Elements,* where Euclid first uses the method of "superposition" to prove it. This may require moving one triangle outside of the plane. But the triangles don't have to be in the same plane to begin with, and they often are not in the same plane when this proposition is invoked in solid geometry. Book I, Proposition 8 is the Side-Side-Side theorem, and in the same Book I, Proposition 26 is an elaboration of the Angle-Side-Angle theorem [in fact, in Proposition 26, he also discusses the case that supports examples like Figure 2. 4(a)]. Euclid doesn't use the word *congruence;* instead he just proves that if certain sides and angles are equal, then the other sides and angles are also equal.

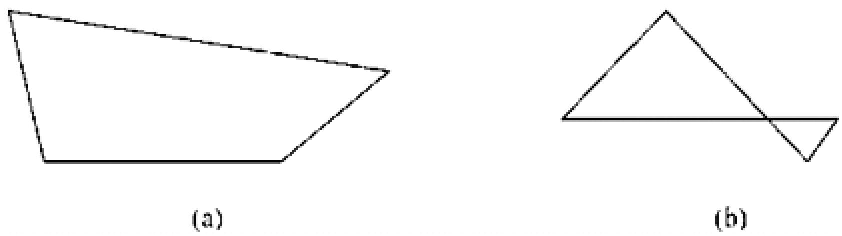


FIGURE 2.5 (a) A simple quadrilateral, because each edge intersects only its adjacent edges at its endpoints. (b) A nonsimple quadrilateral, in which an edge also intersects a nonadjacent edge.

The one-to-one correspondence of points on an oriented line and real numbers is not quite obvious. Struik (1953) gives a simple explanation of this connection. After Rene Descartes formally introduced coordinates in the 17th century to study geometry, the link between algebra and geometry grew stronger. Use of matrices to represent linear transformations such as rigid motion started only in the late 19th century.

Lord Kelvin's quote on chirality is from his Baltimore lectures on molecular dynamics and the wave theory of light, which were finally published in 1904. The word *chiral* is derived from the Greek word *kheir,* which means "hand." The concept of chirality is quite important in chemistry because many important molecules are chiral and both image forms appear in nature, each with different properties. The 2001 Nobel Prize in chemistry was awarded to those who developed molecules that can catalyze important reactions so that only one of the two mirror image forms is produced.

3
Dimensioning Elementary Curves

Curves are simple geometric objects to deal with, and they can be used to generate (for example, by sweep operations) some commonly known surfaces. Therefore it is only natural to start with curves. We begin by asking whether two given curves are congruent. For circles we have an easily provable result.

Theorem 3.1: Circle Congruence Theorem *If two circles have the same radius, then they are congruent.*

So for circles we can treat the radius as the sole intrinsic parameter. The circle belongs to a 1-parameter family of curves. Assigning a numerical value to this parameter completes the task of dimensioning a circle. Actually, one may want to dimension the diameter rather than the radius in some cases, but that is merely a matter of engineering convenience.

In fact, it is possible to generalize Theorem 3.1 for all plane curves. A general plane curve, of course, does not have just one radius. If the curvature (inverse of the radius of curvature) of a curve expressed as a function of its arc length computed from a suitable point and in a given direction, say, counterclockwise, equals that of another curve expressed similarly, then they can be shown to be congruent. We will encounter a general version of this theorem at the end of this chapter. That is why curvature is treated as an intrinsic characteristic of a planar curve. This powerful result, alas, is not of much use for dimensioning because specifying curvature at every point on a curve is not very practical. Luckily, if we restrict ourselves to some special classes of curves we can find congruence theorems that result in only a small number of dimensions.

The simplest curve is the unbounded straight line. It doesn't have an intrinsic dimension, in the sense that all straight lines are congruent to one another. The next elementary curve is a planar curve of the second degree.

3.1 CONICS

Conics are planar algebraic curves of the second degree. They can be represented implicitly as the set of points satisfying a general second-degree equation in coordinates x and y as

$$\{(x, y): Ax^2+By^2+Cxy+Dx+Ey+F=0\} \tag{3.1}$$

for real coefficients $A, B, C, D, E,$ and $F,$ where at least one of A, B, C is nonzero. A soft analysis of this second-degree equation gives us some useful insight. The six coefficients

A, B, C, D, E, and *F* can take arbitrary real values, but the equation remains unaltered if the coefficients are multiplied by the same factor. Hence only the ratios of these six coefficients are significant. This means that a conic can have, in general, five independent parameters (or degrees of freedom), out of which three—two translational and one rotational—are accounted for in rigid motion in the plane. So the intrinsic shape of a conic depends at most on two independent parameters.

This soft analysis is borne out by a more rigorous analysis in classical analytic geometry, which gives us the following classification theorem.

Theorem 3.2: Conics Classification Theorem *Any planar curve of second degree governed by an equation of the form of Eq. (3.1) can be moved by purely rigid motion in the plane so that its transformed equation can assume one and only one of the nine canonical forms given in Table 3.1.*

Of these nine canonical equations, only six correspond to curves in the real *x-y* plane, and so we will ignore the imaginary ones as being of no relevance in engineering design. Out of these six real curves, three (ellipse, hyperbola, and parabola) have nonzero curvature, while the remaining three (intersecting lines, parallel lines, and coincident lines) are special collections of a pair of straight lines. *Conics* is the short name for conic sections, and this name is derived from the well-known fact that the ellipse, the hyperbola, and

TABLE 3.1 Classification of Conics

Conic type	Canonical equation	Intrinsic parameters
1 Real ellipse	$\frac{x^2}{a^2}+\frac{y^2}{b^2}=1, \quad a \geq b$	*a, b*
2 Imaginary ellipse	$\frac{x^2}{a^2}+\frac{y^2}{b^2}=-1$	Not relevant
3 Hyperbola	$\frac{x^2}{a^2}-\frac{y^2}{b^2}=1$	*a, b*
4 Parabola	$y^2-2lx=0$	l
5 Real intersecting lines	$\frac{x^2}{a^2}-\frac{y^2}{b^2}=0, \quad a \geq b$	b/a or $\tan^{-1}(b/a)$
6 Imaginary intersecting lines [intersecting at a real point (0, 0)]	$\frac{x^2}{a^2}+\frac{y^2}{b^2}=0$	Not relevant
7 Real parallel lines	$x^2=a^2$	a
8 Imaginary parallel lines	$x^2=-a^2$	Not relevant
9 Coincident lines	$x^2=0$	None

the parabola can be obtained by intersecting a right circular cone with planes, as shown in Figure 3.1.

An immediate consequence of the conics classification theorem is the following congruence theorem.

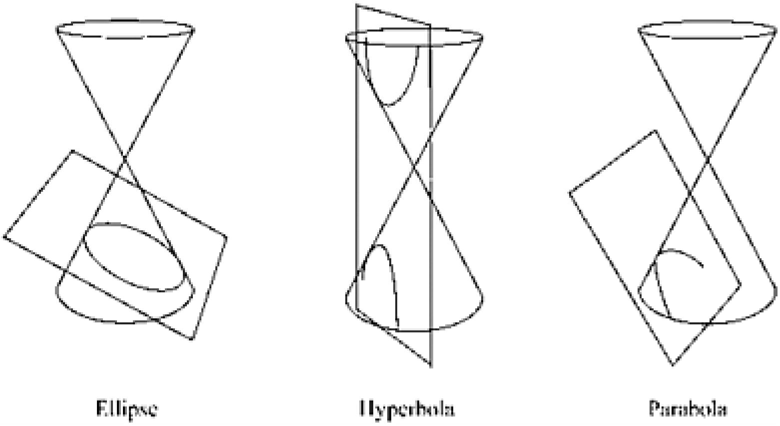


FIGURE 3.1 Sectioning a right circular cone by a plane, producing an ellipse (plane cuts only one sheet of the cone completely), a hyperbola (plane is parallel to the axis of the cone), and a parabola (plane is parallel to a generator of the cone).

Theorem 3.3: Conics Congruence Theorem *Two conics are congruent if and only if they have the same canonical equation.*

So if two conics have the same classification and the intrinsic parameters (listed in the last column of Table 3.1) in their canonical equations assume the same values, then they are congruent. This fact provides a simple way to dimension conics; we just have to declare the type (from Table 3.1) of the conic and assign numerical values to its intrinsic parameters. Thanks to the conics classification theorem, we need to consider only four major types: the ellipse, the hyperbola, the parabola, and a pair of straight lines. Of these, the ellipse, the hyperbola, and the parabola are called *nondegenerate* conics; pairs of lines are the *degenerate* conics.

3.1.1 Ellipse

The ellipse belongs to a 2-parameter family of curves. It is the only bounded curve among the conics. It has two axes of symmetry, as shown in Figure 3.2. These two axes intersect at the center (of symmetry) of the ellipse. A geometrical interpretation of the intrinsic parameters a and b is shown in Figure 3.2(a), where, when positive numerical values are assigned to them, the larger of the two is called the semimajor axis and the smaller the semiminor axis. Calling some parameters and their numerical values axes may sound strange, but the names have stuck through history. If $a=b$, we have the important special case of a circle with $a=b=$the radius. If two ellipses have the same

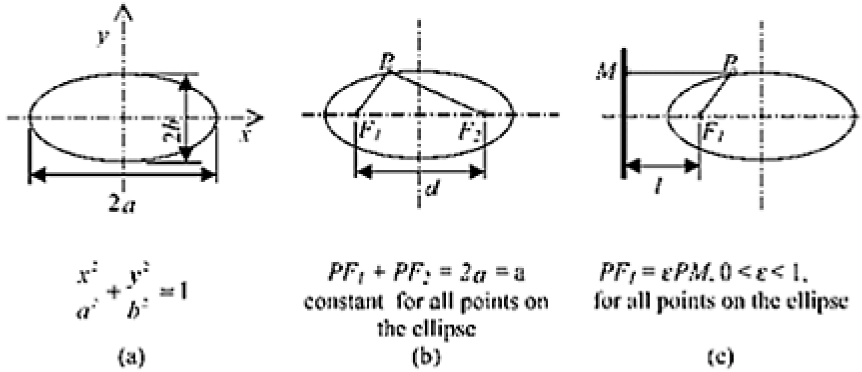


FIGURE 3.2 Ellipse defined by (a) the canonical algebraic equation, (b) two foci, F_1 and F_2, and the sum of the distances of any point on the ellipse from the foci, and (c) a focus F_1, a directrix, shown as the thick vertical line, and an eccentricity ε.

major axis and the same minor axis, then they have the same canonical equation and hence are congruent. Theorem 3.1 is just a special case of this property. Figure 3.2(a) also shows how an ellipse can be dimensioned. The ellipse is the only bounded conic.

The ellipse can be defined by other means as well. There are coordinate-free definitions of conics dating back to the ancient Greeks. Figure 3.2(b) illustrates one such definition of the ellipse, as the locus of a point the sum of whose distances from two fixed points called *foci* is a constant. This constant is also the major axis *2a*. The distance *d* between the foci is equal to $2\sqrt{a^2-b^2}$ So the distance d between the foci and the sum *2a* of the distances of any point on the ellipse from the foci can be considered as another pair of intrinsic parameters. Figure 3.2(b) shows one of the intrinsic dimensions by indicating the distance between the foci. Though it is difficult to indicate the sum of the distances of any point on the ellipse from the foci as the other intrinsic dimension in a drawing, it can be captured easily within a CAD system.

Another classical definition of the ellipse invokes a *directrix* and a *focus,* which are an arbitrarily fixed line and an arbitrarily fixed point not on the line, respectively. See Figure 3.2(c). The ellipse is the locus of a point *P* whose distance from the focus F_1 is ε (the *eccentricity*) times its distance from the directrix *M,* where $0<\varepsilon<1$. It is therefore possible to consider the distance *l* between the focus and the directrix and the eccentricity ε as a pair of intrinsic parameters for ellipses. Again, it is easy to indicate the intrinsic dimension for *l,* as shown in Figure 3.2(c), in a drawing, but it is not so easy to indicate the eccentricity as a dimension; CAD systems have better means of capturing these dimensions.

3.1.2 Hyperbola

The hyperbola also belongs to a 2-parameter family of curves. It has two disjoint branches, and each branch is unbounded. Its two axes of symmetry are shown in Figure

3.3. These two axes intersect at the center (of symmetry) of the hyperbola. A geometrical interpretation of its intrinsic parameter *a* is easily shown in Figure 3.3(a). The value *2a* is called the *transverse axis*. Interpretation of the intrinsic parameter *b* requires some additional considerations. The hyperbola has two *asymptotes,* which are intersecting straight lines given by the equation

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = 0 \tag{3.2}$$

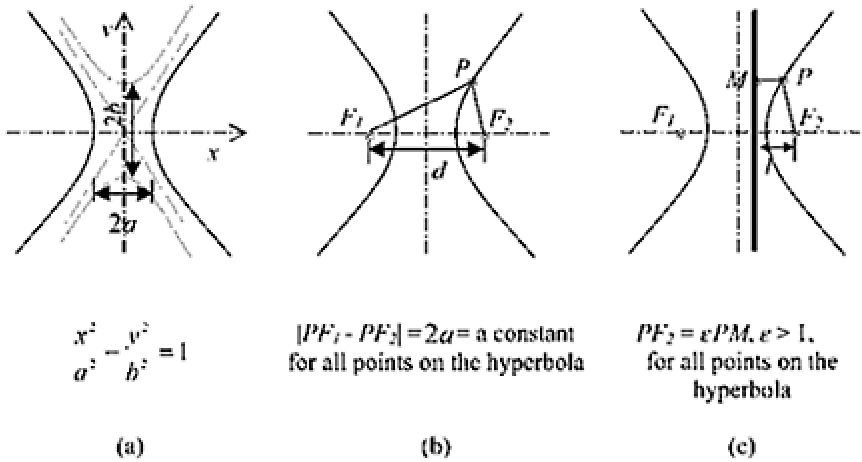


FIGURE 3.3 Hyperbola defined by (a) the canonical algebraic equation, (b) two foci, F_1 and F_2, and the difference between the distances of any point on the hyperbola from the foci, and (c) a focus F_2, a directrix, shown as the thick vertical line, and an eccentricity *ε*.

An asymptote may be regarded as the limiting case of a tangent when the point of contact goes to infinity. The lines of Eq. (3.2) are also asymptotes for a *conjugate hyperbola,* defined by

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = -1 \tag{3.3}$$

The asymptotes and the conjugate hyperbola are shown dotted in Figure 3.3(a). The value *2b* is called the *conjugate axis*. *2a* and *2b* form the sides of a rectangle whose vertices lie on the asymptotes, thus permitting a dimensioning scheme shown in Figure 3.3(a). We are justified in treating them as dimensions because two hyperbolas that have the same transverse and conjugate axes are congruent.

Other classical definitions that predate analytic geometry are available for hyperbolas. In one of them, illustrated in Figure 3.3(b), the hyperbola is the locus of a point *P* the difference of whose distances from two fixed focus points F_1 and F_2 is a constant, which is equal to *2a*. It means that, in Figure 3.3(b), the difference between PF_1 and PF_2 is kept a constant equal to *2a*. Here again the distance *d* between the foci and the constant difference *2a* can be considered as intrinsic parameters of the hyperbola, permitting a partial dimensioning, as in Figure 3.3(b).

It is also possible to define hyperbolas using the focus, the directrix, and the eccentricity as was done with ellipse. Allowing the eccentricity ε to assume positive values exceeding unity yields hyperbolas. Similar to the ellipse, the distance *l* between the focus and the directrix and ε can be treated as intrinsic parameters and dimensioned. See Figure 3.3(c).

3.1.3 Parabola

The parabola has only one intrinsic parameter and only one axis of symmetry. It has no center of symmetry. Hence it is the only noncentral conic. It is made up of one connected piece and is unbounded. Unlike the ellipse and the hyperbola, a geometric interpretation of its intrinsic parameter *l* requires a consideration of its focus. To understand how the focus of a parabola is defined, we need to consider the locus definition of the parabola. The parabola is the locus of a point that is equidistant from a fixed line *(directrix)* and a fixed point *(focus)* not on the line. See Figure 3.4(b). As seen in Figure 3.4(a), the chord of the parabola through its focus and perpendicular to its axis is called the *latus rectum* and is of length 2*l*. The vertex of the parabola is at the origin *O* and is at a distance *l*/2 from the focus. Parabolas that have the same-length latus recta are congruent.

The distance between the focus and the directrix is an intrinsic parameter *l* for the parabola, and it can be dimensioned. The parabola can be obtained as a limiting case of the ellipse by fixing *l* and driving ε toward unity in Figure 3.2(c).

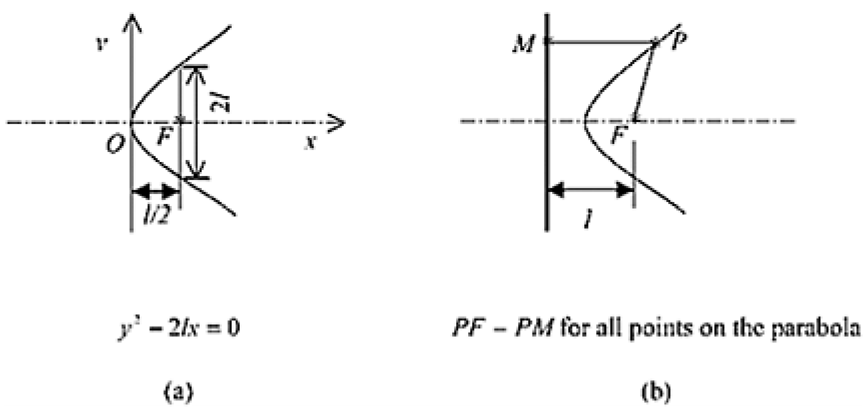


FIGURE 3.4 Parabola defined by (a) the canonical algebraic equation, and (b) a focus *F* and a directrix, shown as the thick vertical line.

3.1.4 Pairs of Straight Lines

A single straight line (or, equivalently, a pair of coincident lines) does not possess any intrinsic dimension. However, when we have two intersecting lines, as in Figure 3.5(a), they have the included angle θ as a relational parameter between them. It is also an intrinsic parameter of the pair of intersecting lines considered together, because if two pairs of intersecting lines have the same angle of intersection then the two pairs are congruent. It can be dimensioned as shown in Figure 3.5(a).

When two straight lines are distinct and parallel, as in Figure 3.5(b), the separating distance between them is a relational parameter between them. It is also the intrinsic parameter of the pair, because two pairs of parallel lines that have the same separating distance are congruent. Such lines are dimensioned in Figure 3.5(b). The case of distinct parallel lines is the only case of conics that is not obtained by sectioning a cone with a plane.

These cases involving pair of straight lines illustrate how relational dimensions between two objects become intrinsic dimensions when we consider a tuple of them.

3.1.5 Reduction to the Canonical Form

We have completed the task of dimensioning conics. However, for the sake of completeness, let's examine how a general second-degree equation can be reduced to its canonical form. The first task is to determine what type of the

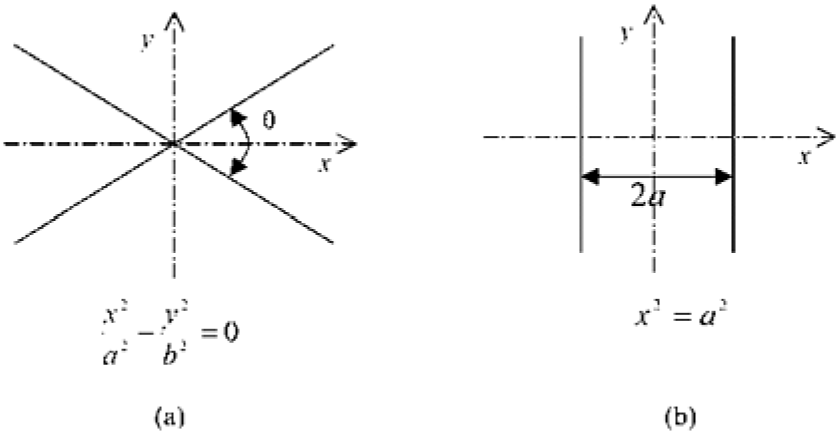


FIGURE 3.5 Pair of lines. (a) Two intersecting lines with their intrinsic parameter *b/a*. Note that $\theta=2\tan^{-1}(b/a)$. (b) Two distinct parallel lines with their intrinsic parameter *a*.

conic is given by the general equation. To simplify the notation, we will recast the Eq. (3.1) in the form

$$c_{11}x^2+2c_{12}xy+c_{22}y^2+2c_{13}x+2c_{23}y+c_{33}=0 \tag{3.4}$$

so that it can be written in a convenient matrix form as

$$\begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0 \tag{3.5}$$

where $c_{21}=c_{12}$, $c_{31}=c_{13}$, and $c_{32}=c_{23}$. So the 3×3 matrix C, called the *coefficient matrix,* in Eq. (3.5) is symmetric. Now define four determinants as

$$D_3 = \begin{vmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{vmatrix}, \quad D_2 = \begin{vmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{vmatrix}, \tag{3.6}$$

$$\alpha_{11} = \begin{vmatrix} c_{22} & c_{23} \\ c_{32} & c_{33} \end{vmatrix}, \quad \alpha_{22} = \begin{vmatrix} c_{11} & c_{13} \\ c_{31} & c_{33} \end{vmatrix}$$

(It will be assumed that, by now, the reader is familiar with the matrix computations reviewed in Appendix 1.) In these determinants, one can see that α_{11}, α_{22}, and D_2 are just the cofactors of c_{11}, c_{22}, and c_{33}, respectively, in C. It can be shown that these determinants lead us directly to the type of the conic represented by Eq. (3.4) using Table 3.2. Once the type has been determined, the curve is rotated and translated so that it is brought to the canonical form.

TABLE 3.2 Decision Table for Type Classification of Conics

$D_3 \neq 0$	$D_2 \neq 0$	$D_2<0$		Hyperbola
		$D_2>0$	$c_{11}D_3$ (or $c_{22}D_3$)<0 $c_{11}D_3$ (or $c_{22}D_3$)>0	Real ellipse Imaginary ellipse
	$D_2=0$			Parabola
$D_3=0$	$D_2 \neq 0$	$D_2<0$ $D_2>0$		Real intersecting lines Imaginary intersecting lines
	$D_2=0$	α_{11} (or α_{22})<0 α_{11}(or α_{22})>0 α_{11} (or α_{22})=0		Real parallel lines Imaginary parallel lines Coincident lines

Example 3.1 Consider the curve defined by the second-degree equation $3x^2+2xy+3y^2+14x+20y-183=0$. It can be written using a symmetric coefficient matrix C as

$$\begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} 3 & 1 & 7 \\ 1 & 3 & 10 \\ 7 & 10 & -183 \end{bmatrix} \begin{Bmatrix} x \\ y \\ 1 \end{Bmatrix} = 0$$

Using the definition of determinants in Eq. (3.6), we see that, for this curve, D_3=−1771, D_2=8, α_{11}=−649, and α_{22}=−598. Therefore, from Table 3.2 we infer that the curve is an ellipse. See Figure 3.6 for a plot of this curve. By rotation and translation, it can be brought to the canonical form

$$\frac{x^2}{100}+\frac{y^2}{50}=1$$

E**xample 3.2** Consider the curve defined by the second-degree equation $3x^2+10xy+3y^2+46x+34y+93=0$. It can be written using a symmetric coefficient matrix C as

$$\begin{bmatrix} x & y & 1 \end{bmatrix}\begin{bmatrix} 3 & 5 & 23 \\ 5 & 3 & 17 \\ 23 & 17 & 93 \end{bmatrix}\begin{Bmatrix} x \\ y \\ 1 \end{Bmatrix}=0$$

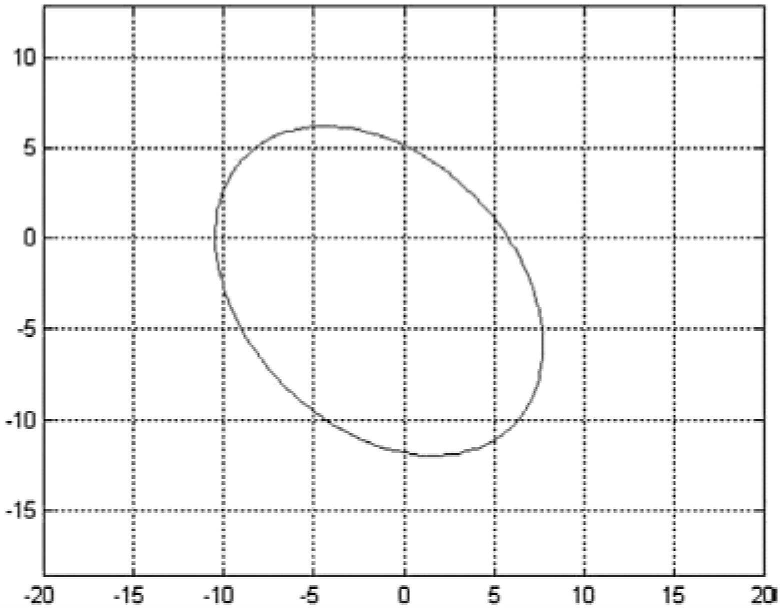


FIGURE 3.6 Plot of the curve in Example 3.1.

Using the definition of determinants in Eq. (3.6), we see that, for this curve, D_3=−32, D_2=−16, α_{11}=−10, and α_{22}=−250. Therefore, from Table 3.2 we infer that the curve is a hyperbola. See Figure 3.7 for a plot of this curve. By rotation and translation, it can be brought to the canonical form

$$x^2-4y^2=1$$

E**xample 3.3** Consider the curve defined by the second-degree equation

$4x^2 - 4xy + y^2 + 8\sqrt{5}x + 6\sqrt{5}y - 15 = 0$. It can be written using a symmetric coefficient matrix *C* as

$$\begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} 4 & -2 & 4\sqrt{5} \\ -2 & 1 & 3\sqrt{5} \\ 4\sqrt{5} & 3\sqrt{5} & -15 \end{bmatrix} \begin{Bmatrix} x \\ y \\ 1 \end{Bmatrix} = 0$$

Using the definition of determinants in Eq. (3.6), we see that, for this curve, D_3=100, D_2=0, α_{11}=−60, and α_{22} =−160.

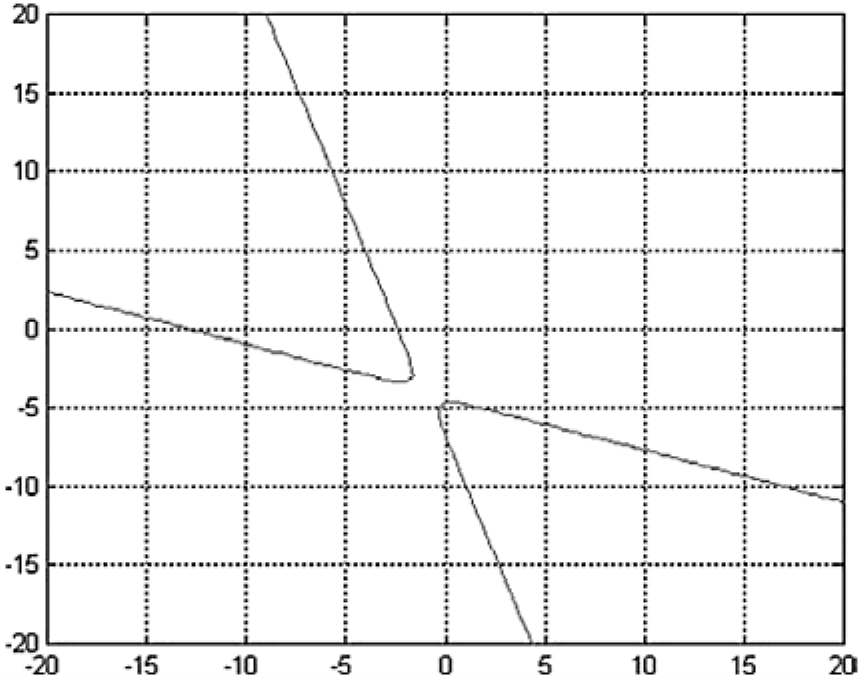


FIGURE 3.7 Plot of the curve in Example 3.2.

Therefore, from Table 3.2 we infer that the curve is a parabola. See Figure 3.8 for a plot of this curve. By rotation and translation, it can be brought to the canonical form

$$y^2 = \frac{1}{4}x$$

3.1.6 Summary of Conics Dimensioning and Extensions

Table 3.3 summarizes commonly used intrinsic dimensions for conics. Also shown in the table are the corresponding intrinsic parameters.

Any conic has at least one axis of symmetry. This is due to the fact that in the canonical form at least one coordinate variable appears only in its second power. Replacing it by its negative will then leave the equation unchanged. Therefore the other coordinate axis is an axis of reflection. This leads to the fact that conics are achiral in the plane that contains them.

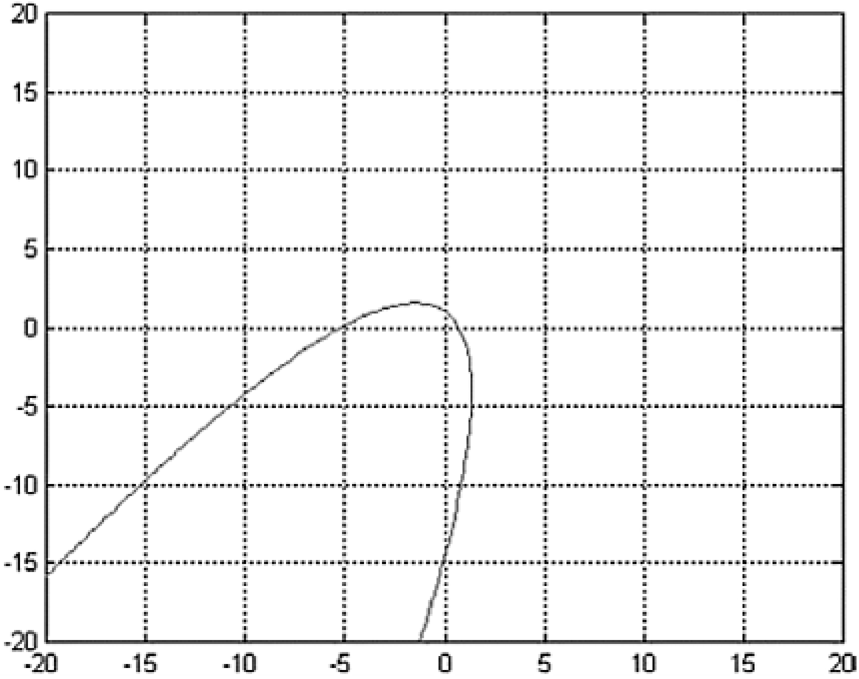


FIGURE 3.8 Plot of the curve in Example 3.3.

TABLE 3.3 Summary of Intrinsic Dimensions for Conics

Conic type	Intrinsic dimension name	Corresponding intrinsic parameter	Reference figure
Ellipse	Major axis	2*a*	Figure 3.2
	Minor axis	2*b*	
Circle	Diameter	2×radius	
Hyperbola	Transverse axis	2*a*	Figure 3.3
	Conjugate axis	2*b*	
Parabola	Latus rectum	2*l*	Figure 3.4
Intersecting lines	Included angle	$2\tan^{-1}(b/a)$	Figure 3.5(a)
Parallel lines	Separating distance	2*a*	Figure 3.5(b)

Conics dimensioning extends very simply to conic half-spaces, that is, two-dimensional regions bounded by conics. For example, an elliptic disk in the plane can be defined using the inequality

$$\frac{x^2}{100}+\frac{y^2}{50}\leq 1$$

Dimensioning it is the same as dimensioning the ellipse that bounds it because congruence theorems can be found for the elliptic half-spaces just as we did for ellipses. In general, a conic half-space can be defined using the inequality

$$c_{11}x^2+2c_{12}xy+c_{22}y^2+2c_{13}x+2c_{23}y+c_{33}\leq 0 \tag{3.7}$$

and dimensioning it is the same as dimensioning the bounding conic.

Conic curves can be swept in space to create surfaces. For example, sweeping an ellipse along a line perpendicular to the plane that contains the ellipse generates an (unbounded) elliptic surface called the *elliptic cylinder*. Dimensioning this elliptic cylinder is the same as dimensioning the ellipse that is swept. Similar arguments hold for hyperbolic and parabolic cylinders. Observe that two intersecting planes and two parallel planes can also be generated by sweeping two intersecting lines and two parallel lines, respectively, perpendicular to the planes that contains the pair of lines; so they can be dimensioned similarly. All these are examples of *translational* sweeps of conics perpendicular to the plane containing the conic.

Conic curves can also be subjected to *rotational* sweep to generate surfaces. In particular, they can be rotationally swept about an axis of symmetry (recall that every conic has at least one axis of reflexive symmetry) to generate a surface of revolution.

1. An ellipsoid of revolution can be generated by rotationally sweeping an ellipse about its major or minor axis.
2. Two disjoint surfaces of revolution can be generated by rotationally sweeping a hyperbola about its transverse axis. Just one surface of revolution can be generated by rotationally sweeping the hyperbola about its conjugate axis.
3. A paraboloid of revolution can be generated by rotationally sweeping a parabola about its axis of symmetry.
4. A right circular cone (consisting of two equal conical surfaces meeting at their common apex) can be generated by rotationally sweeping two intersecting lines about one of their two axes of symmetry.
5. A right circular cylinder can be generated by rotationally sweeping two distinct, parallel lines about their bisector line of symmetry.

In all these cases, dimensioning each of these surfaces of revolution is the same as the dimensioning of the swept conic.

These translational or rotational swept surfaces also bound three-dimensional half-spaces. Just as we saw in the case of two-dimensional half-spaces, dimensioning these three-dimensional half-spaces is the same as dimensioning the conics that were swept.

3.2 FREE-FORM CURVES

Curves of third and higher degree do not admit simple and compact classification as conics. This means that even for general cubics we do not have the luxury of simple congruence theorems. Fortunately, most of the engineering uses of cubic and higher-order curves occur as free-form curves, such as Bézier and B-spline curves.

The conic curve representation in Eq. (3.1) was referred to as *implicit* because the governing equation is of the implicit form $f(x, y)=0$. Such curves are also called *implicit curves*. A curve can also be represented *parametrically* by expressing $x=f_1(t)$, $y=f_2(t)$, and, if it is a space curve, $z=f_3(t)$. In this representation t is the parameter, which may be confined to vary within a finite interval. Such curves are also called *parametric curves*. This may cause some confusion in our treatment of parameters in geometric models, and the distinction can be clarified as follows.

Recall that a curve is treated as a point-set S, where the members of the set are individual points. In a parametric curve each value of the scalar parameter t is mapped to a unique point on the curve in a plane or in space. This is a simple way of addressing each point in a particular point-set S. But the whole curve itself, that is, the point-set S itself, can be dimensioned or parameterized from the outset. Under this scheme we can have two different curves, that is, point-sets S_1 and S_2, that correspond to different sets of dimensions. For example, a circle can be represented parametrically as $x= r \cos(t)$, $y=r \sin(t)$, where t is the angle parameter that can vary between 0 and 2π for any particular circle. However, the circle itself is parameterized by its radius r, and this holds good whether the circle is given a parametric representation as here or an implicit representation as in $x^2+y^2=r^2$. The meaning of the term *parameter* should be clear from the context.

A particular parametric representation of a curve is a linear combination of certain functions (called *basis functions*), where coordinates of certain points (called *control points*) are used as the multiplying coefficients, as in

$$\begin{Bmatrix} x(t) \\ y(t) \end{Bmatrix} = \begin{Bmatrix} x_0 \\ y_0 \end{Bmatrix} \varphi_0(t) + \begin{Bmatrix} x_1 \\ y_1 \end{Bmatrix} \varphi_1(t) + \cdots + \begin{Bmatrix} x_n \\ y_n \end{Bmatrix} \varphi_n(t) \tag{3.8}$$

for plane curves, or as in

$$\begin{Bmatrix} x(t) \\ y(t) \\ z(t) \end{Bmatrix} = \begin{Bmatrix} x_0 \\ y_0 \\ z_0 \end{Bmatrix} \varphi_0(t) + \begin{Bmatrix} x_1 \\ y_1 \\ z_1 \end{Bmatrix} \varphi_1(t) + \cdots + \begin{Bmatrix} x_n \\ y_n \\ z_n \end{Bmatrix} \varphi_n(t) \tag{3.9}$$

for space curves. Both of Eqs. (3.8) and (3.9) can be written more compactly as

$$p(t) = \sum_{i=0}^{n} p_i \varphi_i(t) \tag{3.10}$$

Here, the points p_i are called *control points* and the functions φ_i are called *basis functions*.

Each control point pi in Eq. (3.10) has the coordinates (x_i, y_i) in Eq. (3.8) in the case of planar curves and (x_i, y_i, z_i) in Eq. (3.9) in the case of space curves. When the control points are joined in sequence by line segments, the resulting object is called the *control polygon*. It might have been better to call it a "control polyline" because it doesn't close on itself, it may self-intersect, and it may not lie in a plane, but the name *control polygon* has stuck and we will continue to use it. Note that the sequence, and hence the indices, of the control points are important. We can also consider the tuple of control points $(p_0, p_1, \ldots, p_n)$ as a representation of the control polygon because, when the points in the tuple are connected in the indicated sequence, it yields the control polygon.

For free-form curves, the parameter t in the basis functions φ_i is constrained to vary within a finite real interval $[\alpha, \beta]$ so that the curves are bounded. In addition, for reasons that will become obvious soon, the basis functions φ_i are chosen such that they satisfy an important property called the *partition of unity,* given by

$$\sum_{i=0}^{n} \varphi_i(t) \equiv 1 \qquad \text{for } t \in [\alpha,\beta] \tag{3.11}$$

Now we prove an interesting theorem.

Theorem 3.4: Free-Form Curve Invariance Theorem *A free-form curve represented by Eq. (3.10) is intrinsically invariant under rigid motion of its control points if and only if its basis functions partition unity in the interval of interest.*

Proof. To prove this theorem for the more general space curve, let's denote the homogeneous coordinates of the control point p_i by X_i so that

$$X_i = \begin{Bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{Bmatrix} \tag{3.12}$$

Then, if the basis functions partition unity, we can rewrite Eq. (3.10) as

$$X(t) = \sum_{i=0}^{n} X_i \varphi_i(t) \tag{3.13}$$

Here, $X(t)$ is the homogeneous coordinate of the point $p(t)$. Now, premultiply both sides of Eq. (3.13) by a 4×4 matrix R, as in Eq. (2.9), that represents a rigid motion. This would result in

$$RX(t) = \sum_{i=0}^{n} (RX_i)\varphi_i(t) \tag{3.14}$$

where matrix multiplication has been distributed within the summation. This shows that the curve transformed by the rigid motion is the same as the curve obtained with control points transformed by the same rigid motion.

To prove "the only if" part, we just need to show that intrinsic invariance under rigid motion implies partition of unity. Assume that the curve is intrinsically invariant under rigid motion of its control points. This means that Eq. (3.14) holds. Expanding the matrix multiplication on both sides and looking at the last row, we obtain

$$1 = \sum_{i=0}^{n} \varphi_i(t) \tag{3.15}$$

which is the result we seek. The planar version of the theorem is proved similarly.

A consequence of Theorem 3.4 is the following congruence theorem.

Theorem 3.5: Free-Form Curve Congruence Theorem *Two free-form curves that share the same basis functions that partition unity are congruent if the tuples of their control points are congruent.*

This implies that dimensioning a free-form curve whose basis functions partition unity is the same as dimensioning its control polygon. It also follows that these free-form curves can be parameterized by parameterizing their control polygons. We will now look at some popular free-form curves.

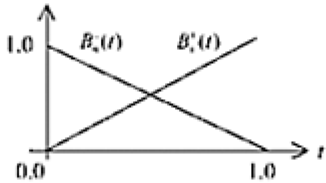
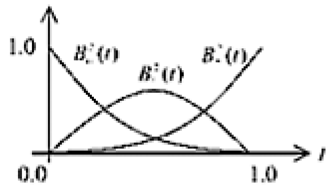
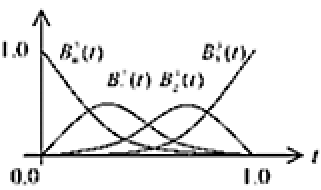
3.2.1 Bézier Curves

When the basis functions in Eq. (3.10) are chosen to be the *Bernstein basis functions* given by

$$\varphi_i(t) = B_i^n(t) = \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i}, \qquad \text{where } t \in [0,1] \tag{3.16}$$

the resulting curve is called a *Bézier curve*. Bernstein basis functions of degree n are used when there are n+1 control points. See Table 3.4 for some low-degree Bernstein basis functions and plots of their graphs. Since the Bernstein basis functions are given unambiguously by Eq. (3.16), a Bézier curve is completely defined by its control points. Figure 3.9 shows three cubic Bézier curves, all having the same set of control points. However, the order sequences of these control points are different in different curves, as the control polygons illustrate. The corresponding Bézier curves are also quite different. These curves show the importance of the tuple of control points, or, equivalently, the control polygon. Bézier curve is a bounded curve due to the finite interval over which the parameter t can vary.

TABLE 3.4 Low-Degree Bernstein Basis Functions

Degree	Bernstein basis functions	Plots of the functions
1	$B_0^1(t)=1-t$ $B_1^1(t)=t$	
2	$B_0^2(t)=(1-t)^2$ $B_1^2(t)=2t(1-t)$ $B_2^2(t)=t^2$	
3	$B_0^3(t)=(1-t)^3$ $B_1^3(t)=3t(1-t)^2$ $B_2^3(t)=3t^2(1-t)$ $B_3^3(t)=t^3$	

It can be easily verified that the Bernstein basis functions partition unity. That is,

$$\sum_{i=0}^{n} B_i^n(t) \equiv 1 \qquad \text{for } t \in [0,1] \tag{3.17}$$

The best way to see this is to obtain a binomial expansion of the left side of the simple identity

$$\{t+(1-t)\}^n \equiv 1 \tag{3.18}$$

A cursory glance at the plots in Table 3.4 should also satisfy the reader as to the veracity of Eq. (3.17). The key result for us, then, is that a Bézier curve can be dimensioned by dimensioning its control polygon. Figure 3.10 illustrates two ways to dimension a planar cubic Bézier curve by dimensioning its control

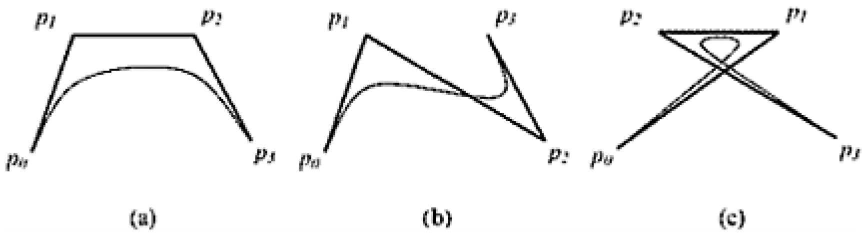


FIGURE 3.9 Three cubic Bézier curves (a–c) along with their control points and control polygons.

polygon. In Figure 3.10(a), the three segments and their included angles are dimensioned. In Figure 3.10(b), a coordinate dimensioning is employed, where, without loss of generality, the first control point is fixed at the origin of a Cartesian coordinate system and the first segment of the control polygon is aligned with the positive *x*-axis. Note that these are not the only options to dimension a cubic Bézier curve.

Following these examples, a simple analysis shows that an *n*th-degree Bézier curve needs $2n-1$ dimensions if it lies in a plane and $3n-3$ dimensions if it lies in space. These are also the numbers of independent parameters if we choose to parameterize these curves.

A quadratic Bézier curve (that is, of second degree), which has three control points, requires three dimensions. It is just an arc of a parabola. From our earlier study of the parabola, recall that an unbounded parabola needs only one dimension. But here a parabolic arc needs two additional dimensions, to

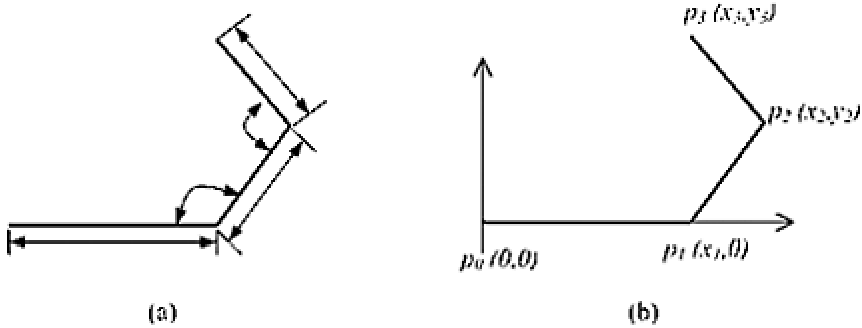


FIGURE 3.10 Dimensioning the control polygon of a planar cubic Bézier curve. (a) Segment lengths and included angles are dimensioned. (b) Coordinate dimensioning of the control points.

indicate where it starts and where it ends. It is also instructive to dimension the control polygon of this quadratic Bézier curve; it is the same as dimensioning a triangle. As we saw in Chapter 2, there are many ways to accomplish even this simple task.

Before we leave Bézier curves, we should consider some of the interesting properties exhibited by these curves.

1. *Endpoint interpolation:* The Bézier curve passes through the first and the last control points. In fact, we might say that it starts at the first control point and ends at the last control point.
2. *End tangents:* The tangent to the Bézier curve at the first control point is aligned with the first segment of the control polygon. The tangent at the last control point is aligned with the last segment of the control polygon.
3. *Convex hull containment:* Note that the Bernstein basis functions are non-negative in the [0, 1] interval. This, combined with the partition of unity property, ensures that the Bézier curve is contained within the convex hull of its control points.

3.2.2 B-Spline Curves

When a set of Bézier curves are smoothly spliced together, we obtain a B-spline curve. In practice, the smooth joining of the Bézier curves is accomplished by a set of B-spline basis functions. To define these basis functions, first we choose a *knot sequence* $[u_0, u_1, \ldots, u_M]$ of real numbers that is nondecreasing; that is, $u_{i-1} \leq u_i \leq u_{i+1}$. Then the *n*th-degree B-spline basis function of parameter u is defined recursively as

$$\varphi_i(u) = N_i^n(u) = \frac{(u - u_{i-1})}{(u_{i+n-1} - u_{i-1})} N_i^{n-1}(u) + \frac{(u_{i+n} - u)}{(u_{i+n} - u_i)} N_{i+1}^{n-1}(u) \qquad (3.19)$$

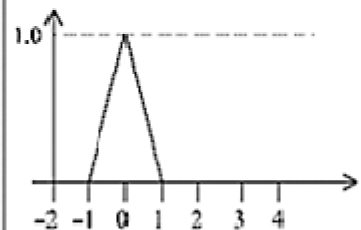
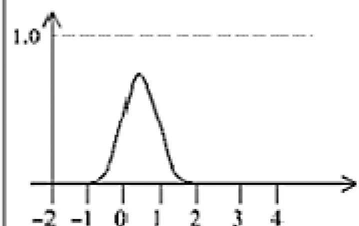
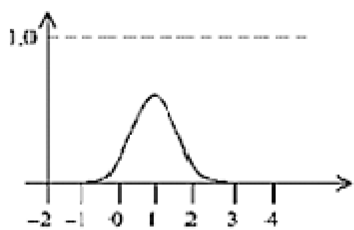
where

$$N_i^0(u) = \begin{cases} 1 & \text{if } u_{i-1} \leq u \leq u_i \\ 0 & \text{otherwise} \end{cases} \qquad (3.20)$$

Table 3.5 shows a typical B-spline basis function of first, second, and third degrees, assuming a uniform knot sequence. In general, we treat the knot sequence as part of the definition of the B-spline basis functions, and the knots need not be uniformly spaced.

Note that a B-spline basis function is nonzero only over a finite interval. We say that such a function has only a *compact support*. This means that local modifications can be made to a B-spline curve by moving a few control points

TABLE 3.5 B-Spline Basis Functions of Low Degrees

Degree n	$N_u^n(u)$	Plot of $N_u^n(u)$
1	$N_u^1(u)=\begin{cases} 1+u & \text{if } -1\le u\le 0 \\ 1-u & \text{if } 0\le u\le 1 \\ 0 & \text{otherwise} \end{cases}$	
2	$N_u^2(u)=\begin{cases} \frac{1}{2}(1+u)^2 & \text{if } -1\le u\le 0 \\ \frac{1}{2}(-2u^2+2u+1) & \text{if } 0\le u\le 1 \\ \frac{1}{2}(2-u)^2 & \text{if } 1\le u\le 2 \\ 0 & \text{otherwise} \end{cases}$	
3	$N_u^3(u)=\begin{cases} \frac{1}{6}(1+u)^3 & \text{if } -1\le u\le 0 \\ \frac{1}{6}(-3u^3+3u^2+3u+1) & \text{if } 0\le u\le 1 \\ \frac{1}{6}(3u^3-15u^2+21u-5) & \text{if } 1\le u\le 2 \\ \frac{1}{6}(3-u)^3 & \text{if } 2\le u\le 3 \\ 0 & \text{otherwise} \end{cases}$	

A uniform knot sequence [−1, 0, 1, 2, 3, 4] is used.

in that vicinity without affecting the curve in other places. Figure 3.11 shows a set of nine quadratic B-spline basis functions, defined over a uniform knot sequence [−1, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. Notice that these functions partition unity only over the interval [1, 8]. The first seven of these basis functions have been used in constructing a second-degree

B-spline curve in the same Figure 3.11. These are a set of five parabolic arcs that have been chained together with tangent continuity. This B-spline curve is defined over the parametric interval [1, 6] because the seven basis functions partition unity over that interval. We could add as many control points as we want without increasing the degree of the curve segments that have been smoothly spliced together to produce the composite curve. In a general B-spline curve, the knot sequence need not be uniform.

In summary, we observe that by choosing a proper interval for *u* we can guarantee that the B-spline basis functions partition unity. Then the problem reduces to dimensioning, or parameterizing, just the control polygon.

3.2.3 Rational Curves

Of all the nondegenerate conics, only (a piece of) the parabola is represented by the Bézier or the B-spline curve. To capture pieces of the ellipse or the

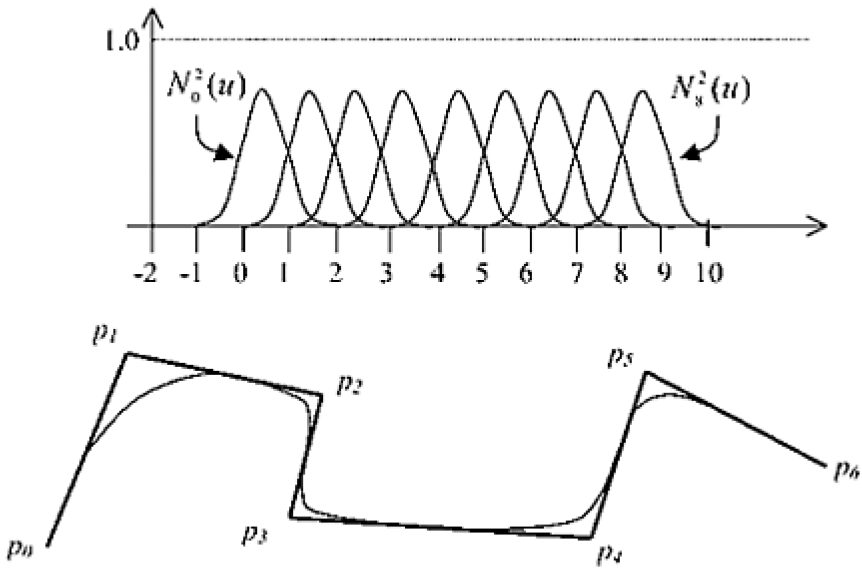


FIGURE 3.11 A set of quadratic B-spline basis functions defined over a uniform knot sequence, and a quadratic B-spline curve using these bases.

hyperbola, we need a rational parametric representation of the form

$$p(t) = \frac{\sum_{i=0}^{n} w_i p_i \varphi_i(t)}{\sum_{i=0}^{n} w_i \varphi_i(t)} \tag{3.21}$$

where wi is the scalar *weight* assigned to the control point p_i. The parameter t is confined to the interval [α, β], as before. We can have the rational Bézier curve or the rational B-spline curve, depending on the choice of the basis function $\varphi_i(t)$.

Example 3.4 A Bézier curve with three control points p_1=(1, 0), p_2=(1, 1), and p_3=(0, 1) defines a parabolic arc. But a rational Bézier curve with the same control points and weights w_1=1.0, w_2=1.0, and w_3=2.0 defines a circular arc. (Why?)

We note that the new basis functions

$$\Phi_i(t) = \frac{w_i\varphi_i(t)}{\sum_{j=0}^{n} w_j\varphi_j(t)} \tag{3.22}$$

still satisfy the partition-of-unity property. (Here the weights are treated as part of the definition of the basis functions.) Therefore, the rational curve also can be dimensioned (or parameterized) by dimensioning (or parameterizing) its control polygon. These results apply directly to the so-called nonuniform rational B-splines (NURBS) curves as well.

3.3 SPACE CURVES

Space curves are those that do not lie in a plane. Free-form space curves have already been dealt with in the last section, where the problem was reduced to dimensioning or parameterizing the control polygons that lie in space. But not all useful space curves have the Bézier or B-spline representation. The helix is one such example.

The helix is a special space curve. It has a constant, nonzero curvature and a constant, nonzero torsion. To understand what we mean by this, we need some background in tangents, normals, and binormals. Consider a parametric representation of a space curve, where the parameter s has a geometric meaning of being the arc length along the curve of any point in question from an arbitrary reference point on the curve. It is also called a *natural representation* of the curve. It gives the curve an orientation. If we regard $p(s)$ as the position vector of any point on the curve, then we have $p(s)=[x(s), y(s), z(s)]^T$ as its coordinates in terms of the arc length s. We can then differentiate it with respect to s to get the unit tangent vector

$$\hat{t} = \frac{dp}{ds} = \begin{Bmatrix} \frac{dx}{ds} \\ \frac{dy}{ds} \\ \frac{dz}{ds} \end{Bmatrix} \tag{3.23}$$

A different choice of the natural representation would give the same unit tangent vector (or its negative if the opposite orientation to the curve has been taken).

A normal vector to the curve is obtained by differentiating the unit tangent vector with respect to *s,* as in

$$\frac{d\hat{t}}{ds} = \kappa(s)\hat{n} \tag{3.24}$$

where κ*(s)* is called the *curvature* at that point and $\hat{n}$ is the unit principal normal vector. The curvature is an intrinsic quantity of the curve because it is independent of the natural parameterization. It tells us how much the curve is curving away from the tangent. Now consider the unit *binormal* vector, defined as the vector cross product

$$\hat{b} = \hat{t} \times \hat{n} \tag{3.25}$$

The tuple $(\hat{t}, \hat{n}, \hat{b})$ forms a right-handed, orthonormal triad. A differentiation of the unit binormal with respect to *s* yields

$$\frac{d\hat{b}}{ds} = -\tau(s)\hat{n} \tag{3.26}$$

where $\tau(s)$ is called the torsion of the curve at that point. Just as with curvature, torsion is an intrinsic property of the curve. The torsion tells us how much the curve is twisting away from the plane determined by the tangent and the normal vectors.

With these preliminaries, the stage is set for the following impressive theorem.

Theorem 3.6: Fundamental Existence and Uniqueness Theorem of Curves *Let κ(s) and* $\tau(s)$ *be arbitrary continuous functions on a≤s≤b. Then there exists, except for position in space, one and only one space curve C for which κ(s) is the curvature,* $\tau(s)$ *is the torsion and s is a natural parameter along C.*

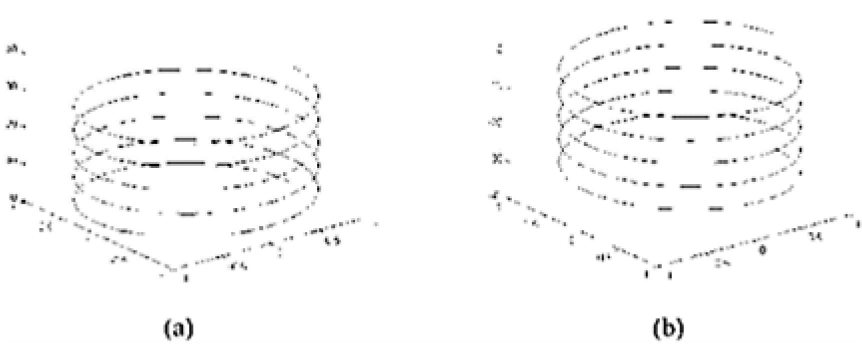


FIGURE 3.12 (a) A right-handed helix. (b) A left-handed helix.

It follows immediately that two space curves are congruent if and only if they have the same arc length parameterization of curvature and torsion. (It specializes to the case of planar curves if we set the torsion to zero.) Obviously, this is the most general congruence theorem for curves we have. Unfortunately, in the area of dimensioning, its use is limited.

Fortunately, we have a direct application of this theorem for the helix. Consider the space curve $\{(x, y, z): x=a \cos \theta, y=a \sin \theta, z=b\theta\}$, where $a>0$ and $b\neq 0$. By varying its angle parameter θ, we see that it traces a helix. Its curvature is constant and is equal to $a/(a^2+b^2)$, and its torsion is also a constant and is equal to $b/(a^2+b^2)$. If $b>0$, that is, if the torsion is positive, then the helix is a right-handed curve. If $b<0$, then the torsion is negative and the helix is left-handed. See Figure 3.12(a) for a right-handed helix with $a=1$ and $b=1$, and see Figure 3.12(b) for a left-handed helix with $a=1$ and $b=-1$. Since torsion, including its sign, is an intrinsic property of the curve, the theorem says that two helices that have the same a and the same magnitude but different signs for b cannot be congruent. See Example 2.17 for a similar discussion on helix.

3.4 EXERCISES

1. Determine the types of the following conics. Plot them to verify your results.

   (a) $3x^2+2xy+3y^2-6x+14-101=0$
   (b) $3x^2-10xy+3y^2+16x-16y+8=0$
   (c) $x^2+4xy+4y^2-4\sqrt{5}x-3\sqrt{5}y=0$
   (d) $xy+x+y+6=0$
   (e) $40x^2+36xy+25y^2+8x-64y-101=0$
   (f) $9x^2+24xy+16y^2-10x+70y-75=0$

2. Common household flashlights have a parabolic reflector. How would you dimension such a reflector? What are the design considerations in the assembly of bulb, reflector, and support casing?
3. Find some information about the planar cubic curves called *cissoid of Diocles, folium of Descartes,* and *witch of Agnesi*. How would you dimension them?
4. The shape of a gear tooth is derived from *involutes*. Find out how they are defined. How are the gear teeth dimensioned?
5. Define a parametric curve using basis functions that do not partition unity in the interval of interest. Show that it is not intrinsically invariant under rigid motion.
6. Prove the assertion that that an nth-degree Bézier curve needs $2n-1$ dimensions if it lies in a plane and $3n-3$ dimensions if it lies in space.
7. Figure 3.11 is for a quadratic B-spline curve. Repeat this exercise for a cubic B-spline curve.
8. Give a rational Bézier representation for a quarter of an ellipse.
9. Prove that the converse of the free-form curve congruence theorem (Theorem 3.5) is false. (*Hint:* Construct a simple counterexample using a Bézier curve. It remains the same if the control points are merely labeled in the reverse order.)

3.5 NOTES AND REFERENCES

The conics classification theorem is so well established in literature that we don't bother to prove it here. Apollonius of Perga wrote extensively about conics as early as the third century B.C. He and other ancient Greeks recognized the latus rectum of a parabola as its *parameter*. It is of historic interest to us to note the origin of this term and how it was used in the context of geometry. An analytic treatment of conics was undertaken only in the last few centuries. Sommerville (1951) is still a sound source for a comprehensive, if somewhat dated, study of conics. A brief, charming introduction to conics can be found in Hilbert and Cohn-Vossen (1983). Struik (1953) gives a nice type classification of conics that we have adopted. A recent, readable account of reducing conics to their canonical form is given by Rutter (2000). His book can be consulted for further information on the canonical reduction. Our examples and exercises for canonical reduction of general second-degree curves are intended to develop a better feel for these curves. We rarely perform these calculations in practice, because modern CAD systems exploit the classification theorem directly to define the nondegenerate conics.

Compared to conics, free-form curve representations are new, and their popularity can be traced to the development of computer-aided design in the last 40 years. Farin (1993) is a standard reference for this material. Our key result on the free-form curve invariance is a special case of a more general invariance theorem. In fact, it can be shown that a free-form curve is invariant under any affine transformation of its control points. But we needed only the result that pertains to rigid motions.

The fundamental existence and uniqueness theorem for curves is one of the major results in differential geometry Our treatment of it is very brief, and we give it here mainly for the sake of completeness. Lipschutz (1969) is a highly readable yet rigorous reference for this theorem.

4
Dimensioning Elementary Surfaces

Surfaces are the geometric objects through which engineering parts interact with each other and with the environment. As such, surfaces play a crucial role in engineering functionality. Following the approach adopted for curves, we start with elementary surfaces and look for their classification and congruence theorems.

The unbounded plane is the simplest surface. It can be defined as the point-set

$$\{(x, y, z): Ax+By+Cz+D=0\} \tag{4.1}$$

for real coefficients *A, B, C,* and *D,* where at least one of *A, B, C* is nonzero. All planes in the Euclidean space are congruent, which leaves the plane with no intrinsic dimension. Next in the hierarchy of complexity are second-degree surfaces called *quadrics,* to which we turn our attention.

4.1 QUADRICS

Quadrics are algebraic surfaces of the second degree. They can be represented implicitly as the set of points satisfying a general second-degree equation in coordinates *x, y,* and *z* as

$$\{(x, y, z): Ax^2+By^2+Cz^2+Dxy+Eyz+Fzx+Gx+Hy+Kz+L=0\} \tag{4.2}$$

for real coefficients *A, B, C, D, E, F, G, H, K,* and *L,* where at least one of *A, B, C, D, E, F* is nonzero. Again, a soft analysis of this equation is fruitful. Although the 10 coefficients in Eq. (4.2) can take arbitrary real values, the equation remains unaltered if the coefficients are multiplied by the same factor. Hence only the ratios of these 10 coefficients are significant. This means that a quadric can have, in general, nine independent parameters (or degrees of freedom), out of which six—three translational and three rotational—are accounted for rigid motion in space. So, intrinsically, a quadric surface has a maximum of three independent parameters.

Classical analytic geometry gives a rigorous support for the foregoing soft analysis in the form of the following classification theorem.

Theorem 4.1: Quadrics Classification Theorem *Any surface of second degree governed by an equation of the form of Eq. (4.2) can be moved by purely rigid motion in space so that its transformed equation can assume one and only one of the 17 canonical* *forms given in Table 4.1.*

Of the 17 canonical equations in Table 4.1, only 12 correspond to surfaces in the real space and so we will ignore the imaginary ones. Out of these 12 real surfaces, nine have nonzero curvature and are illustrated in Figure 4.1, while the remaining three (intersecting planes, parallel planes, and coincident planes) are special collections of a pair of planes.

The quadrics classification theorem leads to the following congruence theorem.

Theorem 4.2: Quadrics Congruence Theorem *Two quadrics are congruent if and only if they have the same canonical equation.*

If two quadrics have the same classification and the intrinsic parameters (listed in the last column of Table 4.1) in their canonical equations assume the same values, then they are congruent. So we can dimension a quadric by declaring its type (from Table 4.1) and assigning numerical values to its intrinsic parameters. Fortunately, we need to consider only a few major types of quadrics, most of which are illustrated in Figure 4.1.

TABLE 4.1 Classification of Quadrics

Quadric type	Canonical equation	Intrinsic parameters
1 Real ellipsoid	$\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}=1, a \geq b \geq c$	*a, b, c*
2 Imaginary ellipsoid	$\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}=-1$	Not relevant
3 Hyperboloid of one sheet	$\frac{x^2}{a^2}+\frac{y^2}{b^2}-\frac{z^2}{c^2}=1, a \geq b$	*a, b, c*
4 Hyperboloid of two sheet	$\frac{x^2}{a^2}+\frac{y^2}{b^2}-\frac{z^2}{c^2}=-1, a \geq b$	*a, b, c*
5 Real quadric cone	$\frac{x^2}{a^2}+\frac{y^2}{b^2}-\frac{z^2}{c^2}=0, a \geq b$	*a/c, b/c*
6 Imaginary quadric cone [with real apex (0,0,0)]	$\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}=0$	Not relevant
7 Elliptic paraboloid	$\frac{x^2}{a^2}+\frac{y^2}{b^2}-2z=0, a \geq b$	*a, b*
8 Hyperbolic paraboloid	$\frac{x^2}{a^2}-\frac{y^2}{b^2}+2z=0$	*a, b*
9 Real elliptic cylinder	$\frac{x^2}{a^2}+\frac{y^2}{b^2}=1, a \geq b$	*a, b*
10 Imaginary elliptic cylinder	$\frac{x^2}{a^2}+\frac{y^2}{b^2}=-1$	Not relevant
11 Hyperbolic cylinder		*a, b*

	$\frac{x^2}{a^2}-\frac{y^2}{b^2}=1$	
12 Real intersecting planes	$\frac{x^2}{a^2}-\frac{y^2}{b^2}=0,\ a \geq b$	*b/a* or $\tan^{-1}$*(b/a)*
13 Imaginary intersecting planes [intersecting at a real line]	$\frac{x^2}{a^2}+\frac{y^2}{b^2}=0$	Not relevant
14 Parabolic cylinder	$y^2-2lx=0$	*l*
15 Real parallel planes	$x^2=a^2$	
16 Imaginary parallel planes	$x^{2=}-a^2$	Not relevant
17 Coincident planes	$x^2=0$	None

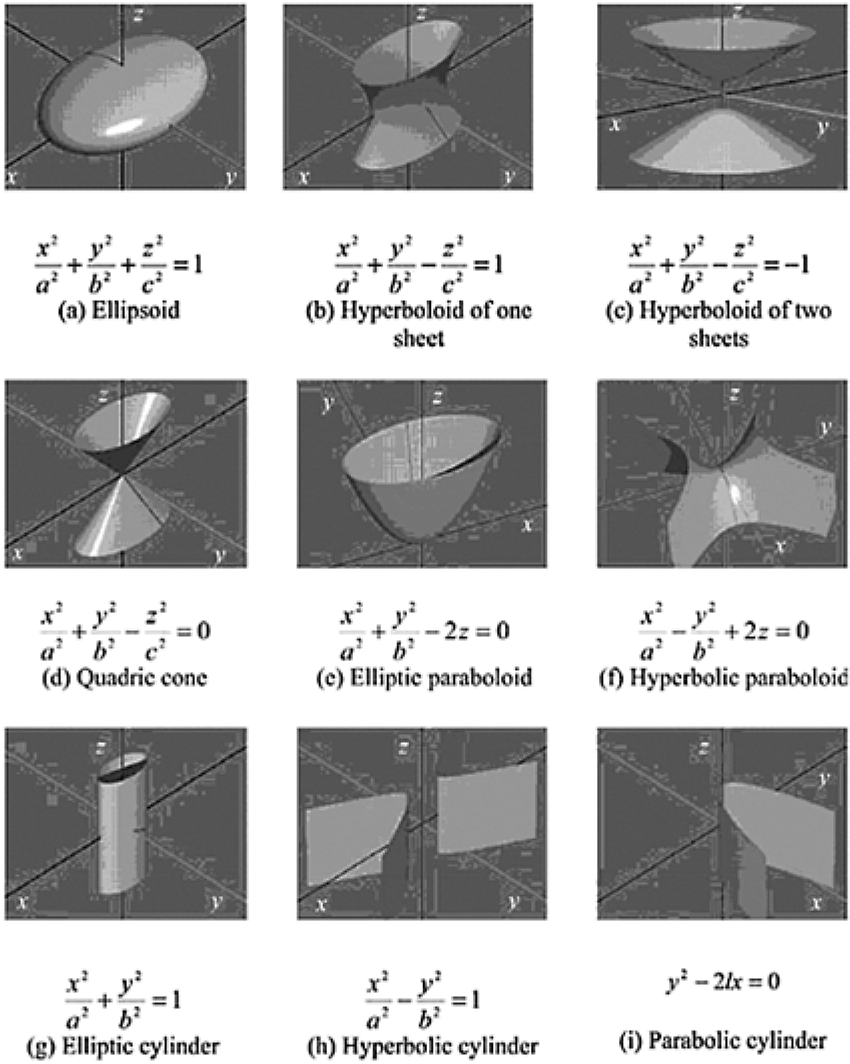


FIGURE 4.1 Nine real quadrics with nonzero curvature.

4.1.1 Ellipsoid

The ellipsoid is the only bounded quadric surface. It belongs to a 3-parameter family and has three planes of reflexive symmetry. See Figure 4.1(a). In its canonical form, it cuts off line segments of lengths 2*a,* 2*b,* and 2*c* from the *x*-, *y*-, and *z*-axes, respectively. These are called the axes of the ellipsoid. The semiaxes are the intrinsic parameters *a, b,* and *c* listed in Table 4.1, thus providing a geometrical interpretation for these parameters. The ellipsoid doesn't have the focal points described for ellipses in Chapter 3. However, the ellipsoid intersects the planes of reflexive symmetry in ellipses. Its projection onto these planes also yields the same result.

The three axes of the ellipsoid can be dimensioned directly on a three-dimensional view, as in Figure 4.1(a). Alternatively, we can dimension the projected views of the ellipsoid. By projecting the ellipsoid on two of its planes of reflexive symmetry, we reduce the ellipsoid dimensioning problem to that of dimensioning two different ellipses.

An important special case of the ellipsoid is the sphere, when a=b=c=the radius. When only two of the semiaxes are equal, say, b=c, we have a surface of revolution called a *spheroid*. It can be obtained by taking the ellipse of Figure 3.2(a) and rotating it about the x-axis. A spheroid is called *oblate* if the third axis is shorter than the first two (like the earth), and *prolate* if the third axis is longer than the first two (like an egg). The spheroid has two focal points. Its directrix is a plane. All the techniques for dimensioning an ellipse can be applied to the spheroid.

In summary, an ellipsoid has three intrinsic dimensions, a spheroid has two, and a sphere has only one.

4.1.2 Hyperboloids of One and Two Sheets

Figures 4.1(b) and 4.1(c) illustrate the two hyperboloids. The hyperboloid of two sheets consists of two disjoint surfaces, whereas the hyperboloid of one sheet has only one connected surface. They have three planes of reflexive symmetry. Both belong to 3-parameter family of surfaces. When a plane parallel to the xy-plane intersects the hyperboloid, it does so at a point or in an ellipse.

Each section of a hyperboloid of one sheet by a plane parallel to the yz-plane or the zx-plane is a hyperbola or a degenerate hyperbola containing two intersecting straight lines. The one-sheeted hyperboloid has two transverse axes ($2a$ and $2b$) and one conjugate axis ($2c$), thus providing a geometrical interpretation for its intrinsic parameters. These can be dimensioned in sectional or projected views. When the transverse axes are equal, we have a one-sheeted hyperboloid of revolution; it is the same as the one obtained by rotating a hyperbola in Figure 3.3(a) about the y-axis, that is, the conjugate axis. It has two intrinsic dimensions.

Perhaps the most surprising fact about a hyperboloid of one sheet is that it is a *ruled surface*. That is, it can be generated by taking a straight line and moving this line in space in some appropriate manner. Another way to look at it is that the hyperboloid of one sheet contains an infinite number of straight lines. We will see more about the one-sheeted hyperboloid as a ruled surface later, in Section 4.3.3.

Each section of a hyperboloid of two sheets by a plane parallel to the yz-plane or the zx-plane is a hyperbola. The two-sheeted hyperbola has one transverse axis ($2c$) and two conjugate axes ($2a$ and 2b). A two-sheeted hyperboloid of revolution results when the conjugate axes are equal; this result can also be obtained by rotating the hyperbola of Figure 3.3(a) about the x-axis, which is also the transverse axis. It then has two intrinsic dimensions.

4.1.3 Quadric Cone

This surface has reflexive symmetry with respect to three planes, as shown in Figure 4.1 (d). In its canonical form, the *z*-axis is its axis and the origin is its vertex. The cone in Figure 4.1(d) is asymptotic to both the hyperboloids in Figures 4.1(b) and 4.1(c). The surface is a cone because it is generated by moving a line that passes through a fixed point (the vertex). It intersects any plane perpendicular to its axis in an ellipse or a single point; the right circular cone is an important special case when the ellipse becomes a circle. The right circular cone is the cone of revolution. The intersection of the quadric cone with a plane is a conic curve.

A general quadric cone belongs to a 2-parameter family, and the right circular cone belongs to a 1-parameter family. A right circular cone can be dimensioned by specifying its apex angle (vertex angle). Dimensioning a general quadric cone requires more effort. One method is to dimension the ellipse obtained by sectioning the cone by a plane perpendicular to the cone axis and located at a unit distance from the vertex.

4.1.4 Elliptic Paraboloid

Figure 4.1(e) shows an elliptic paraboloid. In its canonical form, the *z*-axis is the axis of the elliptic paraboloid and the origin is its vertex. It has two planes of reflexive symmetry. When a plane perpendicular to the *z*-axis intersects the surface, it does so in an ellipse or a point. The origin is the vertex of the elliptic paraboloid in its canonical form. A paraboloid of revolution is a special case when the ellipse specializes to a circle. The elliptic paraboloid belongs to a 2-parameter family, and the paraboloid of revolution belongs to a 1-parameter family.

An elliptic paraboloid can be dimensioned by dimensioning the ellipse obtained by intersecting the surface by a plane perpendicular to its axis at a distance of 1/2 units from the vertex. A paraboloid of revolution is dimensioned by dimensioning its generating parabola.

4.1.5 Hyperbolic Paraboloid

Arguably, the hyperbolic paraboloid is the most intriguing quadric. It is shown in Figure 4.1(f). This saddle-shaped surface has reflexive symmetry with respect to the *yz*- and *zx*-planes. The intersection of the surface by a plane perpendicular to the *z*-axis is a hyperbola that can degenerate to two intersecting lines at the origin. Intersection by planes parallel to the other coordinate planes are parabolas. The hyperbolic paraboloid belongs to a 2-parameter family. It may be dimensioned by dimensioning the hyperbola obtained by intersecting the surface by the plane $z=1/2$.

Surprisingly, the hyperbolic paraboloid is also a ruled surface. In fact, it is a doubly ruled surface. We will see more about this ruled surface in Sections 4.2 and 4.3.3.

4.1.6 Quadric Cylinders

Elliptic, hyperbolic, and parabolic cylinders, shown in Figures 4.1(g), 4.1(h), and 4.1(i), are obtained by taking an ellipse, a hyperbola, and a parabola, respectively, and sweeping them along a direction perpendicular to the plane that contains these curves. Note that these cylinders extend indefinitely, both forward and backward, in the sweep direction.

Intrinsic parameters for these cylinders are the same as those of the curves being swept, as can be seen from Table 4.1. It is also intuitively clear that when the curves are swept perpendicular to the plane that contains them, no additional dimensions (or parameters) are introduced. Therefore, elliptic and hyperbolic cylinders belong to a 2-parameter family, while the parabolic cylinder belongs to a 1-parameter family. Dimensioning the underlying planar curves also dimensions these cylinders.

An important special case of the elliptic cylinder is the right circular cylinder, when the ellipse is specialized to a circle. It then has only one intrinsic dimension, the radius.

4.1.7 Pairs of Planes

Planes are the degenerate quadrics, just as lines are the degenerate conics. Coincident planes have no intrinsic dimension. Intersecting planes have the included angle as the intrinsic dimension of the pair. The separating distance between parallel planes is the intrinsic dimension of that pair.

4.1.8 Reduction to the Canonical Form

As we did in the case of conics, it is possible to infer the type of the quadric defined by a general second-degree equation using a compact decision table. For this, we first recast Eq. (4.2) as

$$c_{11}x^2+c_{22}y^2+c_{33}z^2+2c_{12}xy+2c_{23}yz+2c_{31}zx+2c_{14}x+2c_{24}y+2c_{34}z+c_{44}=0 \tag{4.3}$$

so that it can be written in a convenient matrix form as

$$[x \quad y \quad z \quad 1]\begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \\ c_{41} & c_{42} & c_{43} & c_{44} \end{bmatrix}\begin{Bmatrix} x \\ y \\ z \\ 1 \end{Bmatrix} = 0 \tag{4.4}$$

where, for the 4×4 symmetric coefficient matrix C_4, $c_{12}=c_{21}$, $c_{31}=c_{13}$, $c_{41}=c_{14}$, $c_{32}=c_{23}$, $c_{42}=c_{24}$, and $c_{43}=c_{34}$. Let's denote the determinant of C_4 by Δ and its rank by ρ_4. The top-left 3×3 submatrix of C_4 is

$$C_3 = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \tag{4.5}$$

and its rank will be denoted by ρ_3. We can then determine the type of the quadric using Table 4.2. In that table, the question "*k*'s same sign?" is answered yes if all the nonzero eigenvalues of C_3 have the same sign; otherwise the answer is no. Similarly, the question "*K*'s same sign?" is answered yes if all the nonzero eigenvalues of C_4 have the same sign; otherwise the answer is no. Once the type has been determined, the surface can be rotated and translated so that it is brought to the canonical form.

Example 4.1 Consider the quadric surface $xy+xz+yz+1=0$. The related matrices are

$$C_4 = \begin{bmatrix} 0 & 0.5 & 0.5 & 0 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{and} \quad C_3 = \begin{bmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{bmatrix}$$

TABLE 4.2 Decision Table for Type Classification of Quadrics

ρ_3	ρ_4	Δ	*k*'s same sign?	*K*'s same sign?	Type of quadric
3	4	<0	Yes		Real ellipsoid
3	4	>0	Yes		Imaginary ellipsoid
3	4	>0	No		Hyperboloid of one sheet
3	4	<0	No		Hyperboloid of two sheets
3	3		No		Real quadric cone
3	3		Yes		Imaginary quadric cone
2	4	<0	Yes		Elliptic paraboloid
2	4	>0	No		Hyperbolic paraboloid
2	3		Yes	No	Real elliptic cylinder
2	3		Yes	Yes	Imaginary elliptic cylinder
2	3		No		Hyperbolic cylinder
2	2		No		Real intersecting planes
2	2		Yes		Imaginary intersecting planes
1	3				Parabolic cylinder
1	2			No	Real parallel planes
1	2			Yes	Imaginary parallel planes
1	1				Coincident planes

Then, we have $\rho_3=3$, $\rho_4=4$, and $\Delta=0.25$. The eigenvalues of C_3 are −0.5, −0.5, and 1.0. The eigenvalues of C_4 are −0.5, −0.5, 1.0, and 1.0. From Table 4.2 we see that the quadric is a hyperboloid of one sheet. By proper rotation and translation, it can be brought to the canonical form

$$\frac{x^2}{2}+\frac{y^2}{2}-z^2=1$$

Example 4.2 Next consider the quadric surface $x^2+2y^2+2z^2+2xy-2xz+2x+6y+2z-13=0$. The relate matrices are

$$C_4=\begin{bmatrix} 1 & 1 & -1 & 1 \\ 1 & 2 & 0 & 3 \\ -1 & 0 & 2 & 1 \\ 1 & 3 & 1 & -13 \end{bmatrix} \quad \text{and} \quad C_3=\begin{bmatrix} 1 & 1 & -1 \\ 1 & 2 & 0 \\ -1 & 0 & 2 \end{bmatrix}$$

Then we have $\rho_3=2$, $\rho_4=3$, and $\Delta=0$. The eigenvalues of C_3 are 3, 2, and 0. The eigenvalues of C_4 are 0, 2.3933, 3.2963, and −13.6896. From Table 4.2 we see that the quadric is a real elliptic cylinder. By proper rotation and translation, it can be brought to the canonical form

$$\frac{x^2}{9}+\frac{y^2}{6}=1$$

Example 4.3 Finally, consider the quadric surface $9x^2+y^2+z^2-6xy+6xz-2yz+18x-6y+6z-7=0$. The related matrices are

$$C_4=\begin{bmatrix} 9 & -3 & 3 & 9 \\ -3 & 1 & -1 & -3 \\ 3 & -1 & 1 & 3 \\ 9 & -3 & 3 & -7 \end{bmatrix} \quad \text{and} \quad C_3=\begin{bmatrix} 9 & -3 & 3 \\ -3 & 1 & -1 \\ 3 & -1 & 1 \end{bmatrix}$$

Then we have $\rho_3=1$, $\rho_4=2$, and $\Delta=0$. The eigenvalues of C_3 are 0, 0, and 11. The eigenvalues of C_4 are 0, 0, 15.4164, and −11.4164. From Table 4.2 we see that the quadric is a pair of real parallel planes. By proper rotation and translation, it can be brought to the canonical form

$$x^2=\frac{16}{11}$$

4.1.9 Summary of Quadrics Dimensioning and Extensions

Of all the quadrics, the most popular are spheres, right circular cylinders, and right circular cones. Each of these has only one dimension (or parameter).

The quadric cylinders of Section 4.1.6 are obtained by translational sweeps of conic curves in a direction perpendicular to the plane that contain these curves. In fact, we encountered them in Section 3.1.6. Similarly, quadrics that possess rotational symmetry about an axis (spheroids, the two hyperboloids of revolution, the paraboloid of revolution, the right circular cone, and the right circular cylinder) are obtained by rotational sweeps of conic curves about their axis of symmetry. In both type of sweeps, no additional dimensions or parameters are introduced. So the problem of dimensioning these swept surfaces reduces to the dimensioning of the underlying planar curves.

If the sweeps are restricted to a finite extent, then an additional dimension (or parameter) is introduced. For example, if a conic is translationally swept only for a finite distance perpendicular to it containing plane, then this distance becomes the height of the quadric cylinder. Similarly, if a conic is rotationally swept about its axis for an angle less than 2π, this angle becomes the additional dimension (or parameter).

Swept surfaces are one popular example of what are called *procedurally* defined geometric objects. They are often used in constructing a geometric model. We will encounter many other procedurally defined objects in later chapters when we explore how to construct complex geometric models from simpler objects.

Some solids in three-dimensional space can be defined using quadric half-spaces. Note that most real, nondegenerate quadrics divide three-dimensional space into two parts. (Exceptions are the hyperboloid of two sheets and the hyperbolic cylinder, which divide space into three separate parts.) The quadric half-spaces can be bounded or unbounded. A general quadric half-space is defined by

$$c_{11}x^2+c_{22}y^2+c_{33}z^2+2c_{12}xy+2c_{23}yz+2c_{31}zx+2c_{14}x+2c_{24}y+2c_{34}z+c_{44}\leq 0 \tag{4.6}$$

A solid spherical ball of unit radius, for example, can be represented by the set $S=\{(x, y, z):x^2+y^2+z^2\leq 1\}$. It is a bounded half-space. A solid, unbounded cylinder of radius 2 can be represented as the set $\{(x, y, z):x^2+ y^2\leq 4\}$. If it were a cylindrical hole, we would represent it as the set $\{(x, y, z):x^2+y^2\geq 4\}$. Dimensioning a quadric half-space is the same as dimensioning its bounding quadric, with an additional indication as to which side of the quadric the solid lies. Recall that Svensen used “positive” and “negative” attributes in Figure 1.1 to indicate the material side.

Finally, we observe that all quadrics and quadric half-spaces are achiral, because each has at least one plane of reflexive symmetry. It is easy to see this from the equations of canonical form in Table 4.1, where there is always one variable that appears in the second degree only. A plane of symmetry is the one in which this variable assumes a zero value.

4.2 FREE-FORM SURFACES

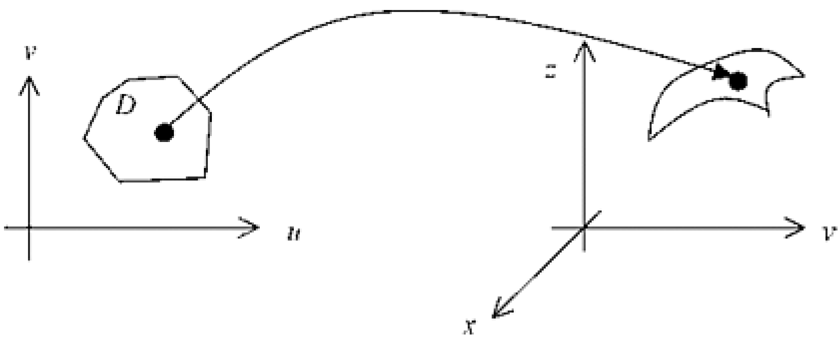
Free-form surfaces are the natural generalization of free-form curves. A surface patch in three-dimensional space can be represented parametrically as

$$x=f_1(u, v),\ y=f_2(u, v),\ \text{and}\ z=f_3(u, v) \tag{4.7}$$

$$p(u, v) = \sum_{l=0}^{N} p_l \varphi_l(u, v), \qquad (u, v) \in D$$

where Here we have two parameters, *u* and *v,* and *D* is their domain of variation. In contrast, a parametric representation for a curve has only one parameter, and it varies just over an interval. Figure 4.2 shows a simple mapping between points in the parametric domain *D* and points on the surface defined by Eq. (4.7).

A surface patch can be represented parametrically as a linear combination of bivariate basis functions, using control points in three-dimensional space as multiplying coefficients, as in



(4.8)

Here $\varphi_l(u, v)$ are the basis functions and p_l are the N+1 control points positioned in space. Following Theorem 3.4, we have the following.

Theorem 4.3: Free-Form Surface Invariance Theorem *A free-form surface represented by Eq. (4.8) is intrinsically invariant under rigid motion of its control points if and only if its basis functions partition unity in the parametric domain D.*

The proof is identical to that of Theorem 3.4. A consequence of Theorem 4.3 is the following congruence theorem.

Theorem 4.4: Free-Form Surface Congruence Theorem *Two free-form surfaces that share the same basis functions that partition unity are congruent if the tuples of their control points are congruent.*

When the basis functions in Eq. (4.8) are written in a separable form

$$\varphi_l(u, v)=\varphi_i(u)\varphi_j(v) \qquad (4.9)$$

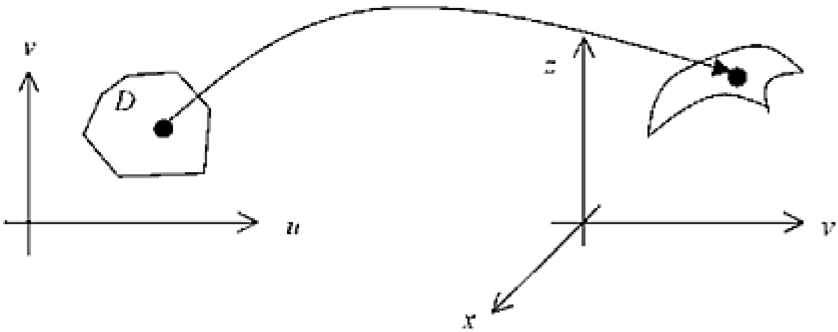


FIGURE 4.2 Mapping between points on the *u*-*v* domain *D* and points on the parametric surface patch.

we have surface patches in *tensor product* form, which can be represented as

$$p(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} p_{i,j} \varphi_i(u) \varphi_j(v) \qquad (4.10)$$

Here $p_{i,j}$ are the control points. When these control points are joined according to the adjacency of their indices in Eq. (4.10), they form a *control net*. See Figure 4.3(a) for an illustration of a control net. The control net is the free-form-surface equivalent of the control polygon for free-form curves.

Assuming the partition-of-unity property of the basis functions in Eqs. (4.9) and (4.10) over the parametric domain *D,* Theorem 4.4 implies that the dimensioning of a free-form surface patch is the same as dimensioning the associated control net.

If we choose $\varphi_i(u)$ and $\varphi_j(v)$ to be Bernstein basis functions, we have a Bézier surface patch, as shown in Figure 4.3(b). In this case, the parametric domain is a square $D=[(0, 1) \times (0, 1)]$, over which it can be shown that the Bernstein basis functions, defined in Eq. (3.16), partition unity; that is,

$$\sum_{i=0}^{m} \sum_{j=0}^{n} B_i^m(u) B_j^n(v) \equiv 1 \qquad (4.11)$$

Similarly, if the basis functions $\varphi_i(u)$ and $\varphi_j(v)$ are B-spline basis functions, then B-spline surface patches result. Rational surface patches can also be

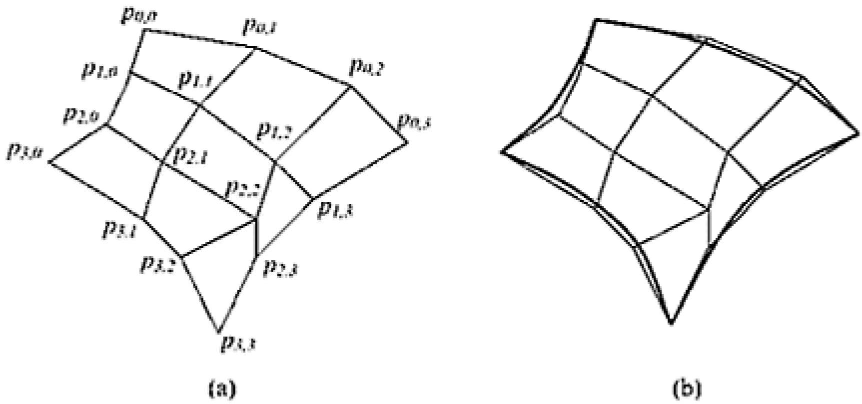


FIGURE 4.3 (a) A Bézier control net and (b) the resulting cubic Bézier surface patch in tensor product form.

created by choosing the basis functions to be of the rational form. In all cases, the surface patch dimensioning problem is reduced to that of dimensioning the associated control net.

A special case of the Bézier surface patch results when we use only four noncoplanar control points. It is illustrated in Figure 4.4. Its parametric representation is given in detail by

$$p(u, v)=p_{0,0}(1-u)(1-v)+p_{0,1}(1-u)v+p_{1,0}u(1-v)+p_{1,1}uv$$

It is a patch of a hyperbolic paraboloid. It is also a ruled surface and has two families of straight-line rulings, one family obtained by setting u=a constant, and the other by setting v=a constant. These rulings are the isoparametric curves on the surface patch.

Sometimes, instead of the tensor product form of Eq. (4.9), a *barycentric* or *triangular form* is used for the basis functions. In the triangular form, the Bernstein basis function is defined as

$$\varphi_l(u, v) = B^n_{i,j,k}(u, v, w) = \frac{n!}{i!\,j!\,k!} u^i v^j w^k \qquad (4.12)$$

where $u+v+w=1$ and $i+j+k=n$, the degree of the polynomial. See Figure 4.5 for an example for such a surface patch, whose parametric representation is given explicitly by

$$p(u, v)=p_{0,0,2}(1-u-v)^2 +p_{1,0,1}2u(1-u-v) +p_{2,0,0}u^2 +p_{0,1,1}2v(1-u-v) +p_{1,1,0}2uv +p_{0,2,0}v^2$$

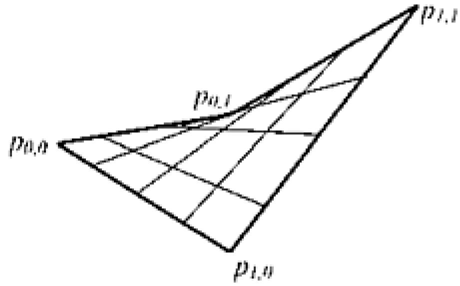


FIGURE 4.4 A Bézier control net and the resulting bilinear Bézier surface patch in tensor product form. The result is a patch of a hyperbolic paraboloid. It is also a doubly ruled surface, as illustrated by the double rulings.

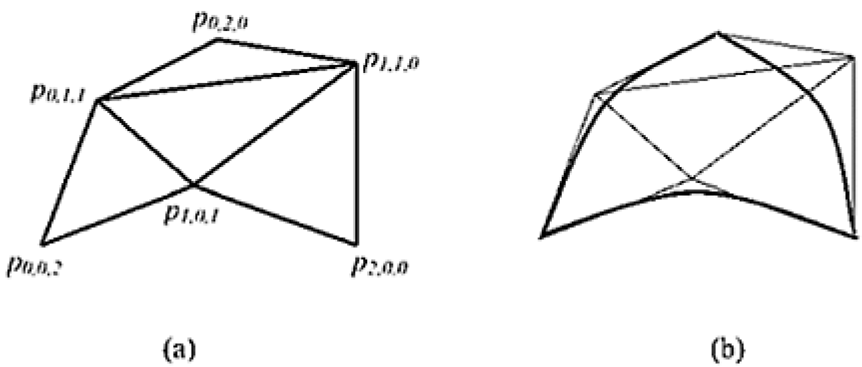


FIGURE 4.5 (a) A Bézier control net and (b) the resulting Bézier surface patch in triangular form.

The resulting surface patch has a triangular net, and the dimensioning and parameterizing problem remains the same.

In general, a free-form surface patch with $N+1$ control points requires $3N-3$ dimensions (or parameters).

4.3 SWEPT SURFACES

We saw in Sections 3.1.6 and 4.1.9 that a surface can be *generated procedurally* by taking a curve and moving it in space. Such surfaces are called *swept surfaces*. Some of the most commonly used surfaces in CAD can be defined procedurally as swept surfaces. In general, the curve being moved can also be deformed along the way. It is possible to give this interpretation for the tensor product surfaces of Eq. (4.10). For example, the cubic Bézier patch in Figure 4.3 can be generated by doing the following.

1. Start with a space cubic Bézier curve, defined by the control points $p_{0,0}$, $p_{0,1}$, $p_{0,2}$, and $p_{0,3}$.
2. Move each of its control points along another Bézier curve. That is, move the control point $p_{0,0}$ along the Bézier curve defined by the control points $p_{0,0}$, $p_{1,0}$, $p_{2,0}$, $p_{2,0}$, and $p_{3,0}$, and similarly move the other control points $p_{0,1}$, $p_{0,2}$, and $p_{0,3}$.

A more restricted, but very useful, procedure is to sweep a curve without deformation by strict translation or strict rotation, leading to *generalized cylinders* and *surfaces of revolution,* as described next.

4.3.1 Generalized Cylinders

Quadric cylinders, discussed in Section 4.1.6, are prime examples of generalized cylinders. These are unbounded surfaces, generated by taking a plane curve and sweeping it in a direction perpendicular to the plane that contains it. These cylinders extend indefinitely, both forward and backward, in the sweep direction. This definition can be stated nonprocedurally, and more formally, as follows (Figure 4.6 illustrates various terms used in the definition).

Definition 4.1: Generalized Cylinder *Let C be a plane curve and let F denote the family of all straight lines through points of C perpendicular to the given plane. The surface consisting of all points of these lines is called a* generalized cylinder. *Each line of the family F is called a* ruling or generator, *and the curve C is called a* directrix. *The cylinder is said to be parallel to its rulings and perpendicular to any plane perpendicular to its rulings.*

The term *directrix* here should not to be confused with the same term used in the definition of conics in Chapter 3. In the context of quadrics, if the directrix is one or more straight lines, the generalized cylinder is one or more planes. Otherwise, it is a curved surface and is usually named after the directrix—as in the elliptic cylinder, the right circular cylinder, the parabolic cylinder, and the hyperbolic cylinder. Other curves, such as free-form curves, can also be swept to generate cylinders.

Since no additional dimension or parameter is introduced in the sweep procedure, dimensioning (or parameterizing) the generalized cylinder is the same as dimensioning (or parameterizing) its directrix, namely, the planar curve *C*. If the sweep is restricted to a finite extent, then the cylinder has a finite height, thereby introducing just one additional dimension or parameter.

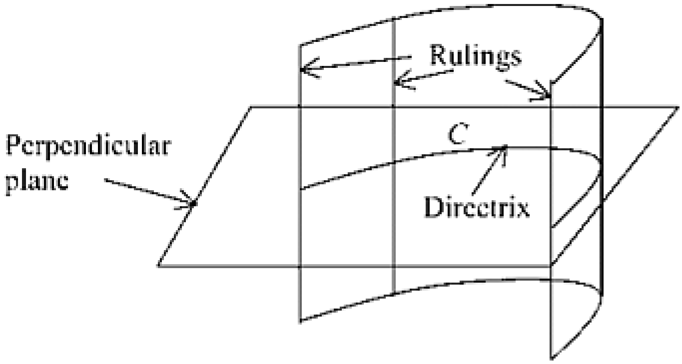


FIGURE 4.6 Illustration of definitions of terms for a generalized cylinder.

4.3.2 Surfaces of Revolution

A surface that is symmetrical with respect to a line and that is cut only in circular sections by planes perpendicular to this line of symmetry is called a *surface of revolution*. (See Section 2.3 for the definition of a line of symmetry for an arbitrary point-set.) Procedurally, we can define the surface of revolution as follows.

Definition 4.2: Surface of Revolution *Let C be a (possibly space) curve and F be a straight line in space. A surface of revolution is generated by revolving C about F. The curve C is called a* generatrix *and the straight line F is called the* axis of revolution *of the resulting surface*.

Quadrics contain several examples of surfaces of revolution, such as the spheroid, the two hyperboloids of revolution, the paraboloid of revolution, the right circular cone, and the right circular cylinder. In these examples, the generatrix is either a line or a conic curve. We already noted how to dimension or parameterize them in Sections 3.1.6 and 4.1.9.

Perhaps the most important surface of revolution that does not belong to the quadrics is the *torus*. It is generated by taking a circle and revolving it about a line in the plane of the circle but that does not pass through the center of the circle. (If the axis passes through the center of the circle, we get a quadric, namely, a sphere, as the surface of revolution.) Figure 4.7 shows examples of the torus as a surface of revolution. Figure 4.8 shows one of its dimensioning schemes. It has two dimensions (or parameters).

1. The first dimension is the radius *r* of the circle being swept. It is an intrinsic dimension of the generatrix.
2. The second dimension is the distance *R* between the center of the generatrix (circle) and the axis of revolution (straight line). It is the relational dimension between the two planar objects—the circle and the straight line. This dimension is often called the radius of the *symmetric axis circle* of the torus. Here is the first important instance where the relative positioning of two simple geometric objects in the plane enters into our dimensioning scheme. (We will say more about relative positioning in the next few chapters.)

Once a torus is constructed, both r and R become its intrinsic dimensions. To prove that this is indeed a valid dimensioning scheme, we need a congruence theorem. It can easily be proved by the simple method of superposition that if two tori have the same generatrix circle radius r and the same distance R between the axis of revolution and center of the generatrix circle, then they are congruent.

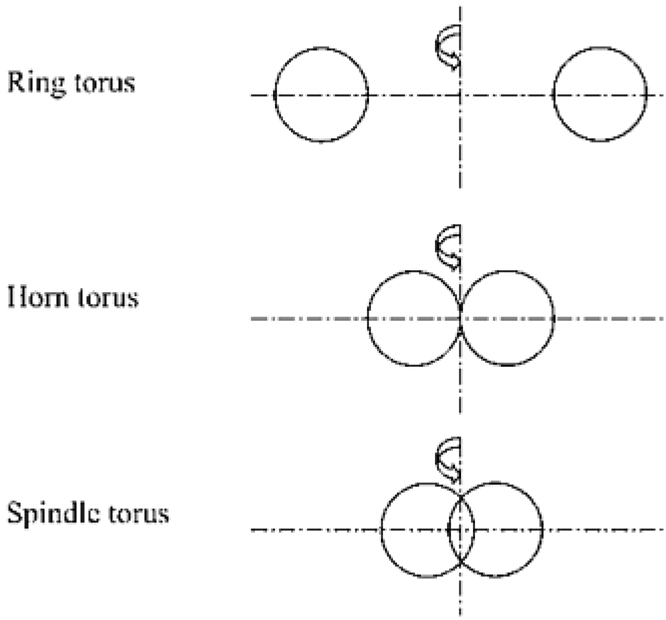


FIGURE 4.7 The torus as a surface of revolution. The ring torus does not have a self-intersection. The horn torus self-intersects at one point. The spindle torus self-intersects at two points.

A torus can be positioned canonically in a three-dimensional Cartesian coordinate system, with the z-axis as its axis of revolution and the origin as its center of symmetry. See Figure 4.8. An implicit equation for such a torus is given by

$$(\sqrt{x^2 + y^2} - R)^2 + z^2 = r^2 \tag{4.13}$$

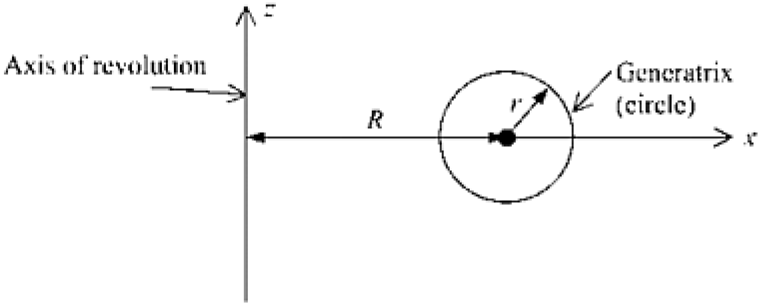


FIGURE 4.8 A dimensioning scheme for the torus, viewed as a surface of revolution. For a ring torus *R*>*r*, for a horn torus *R*=*r*, and for a spindle torus *R*<*r*.

It cannot be brought to any of the canonical equations for quadrics, and hence the torus is not a quadric. It actually belongs to a fourth-degree algebraic surface. Toroidal patches are commonly used as fillets and rounds in geometric models, thus deserving our special attention.

Before we conclude our discussion on surfaces of revolution, we observe that a generatrix curve can be revolved about the axis through an angle $\theta<2\pi$. Then, that angle θ becomes an additional dimension, or parameter, for the swept surface.

4.3.3 Ruled Surfaces

A nonprocedural definition of a ruled surface is as follows.

Definition 4.3: Ruled Surface *A surface is a ruled surface if and only if through every point on the surface there is a straight line* (*called* ruling) *that lies completely on the surface*.

Based on this definition, all generalized cylinders are ruled surfaces. The quadric cone is also a ruled surface; it can be generated by joining every point on a conic curve to a point that is not on the plane containing the conic, using a straight line. In general, we can give a procedural definition of a ruled surface as the one generated by sweeping a straight line in space in some well-defined fashion.

A one-sheeted hyperboloid can be generated procedurally as a ruled surface, as illustrated in Figure 4.9. First, let's see how a one-sheeted hyperboloid of revolution can be generated both as a ruled surface and as a surface of revolution. Consider two skew lines *a* and *g* in space, as in Figure 4.9(a). Treating *a* as the axis of revolution and *g* as the generatrix, a surface of revolution can be obtained, as shown in Figure 4.9(b). It is a one-sheeted hyperboloid of revolution. The same surface can be obtained by choosing a different generatrix, *g*′, shown in Figure 4.9(a). So here we have a *doubly ruled surface;* that is, through each point on the surface we have two distinct rulings that lie completely on the surface. A general hyperboloid of one sheet can be obtained, as shown in Figure 4.9(c), by uniformly stretching the surface of revolution in one horizontal direction.

The foregoing construction of a hyperboloid of one sheet as a ruled surface gives us an alternative way to dimension it. For this, we observe that the relative positioning of the axis line *a* and directrix line *g* is completely specified by the shortest distance *d* between them and the twist angle θ between them. (We will encounter the problem of the relative positioning of a pair of skew lines in detail in the next chapter.) These are the intrinsic dimensions of the

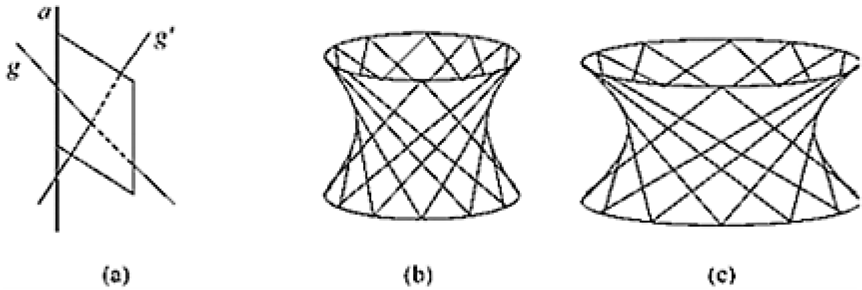


FIGURE 4.9 Generating a one-sheeted hyperboloid as a ruled surface. (a) *a* and *g* are skew lines in space. *g*′ is a reflection of *g* about a plane that contains *a* and the common perpendicular between *a* and *g*. (b) The surface obtained by revolving the generatrix *g* around the axis *a*. The same surface can be obtained by revolving *g*′ instead of *g*. (c) Applying dilation (that is, stretching) along a horizontal direction results in a general hyperboloid of one sheet.

tuple of two skew lines, and they can be used as the two dimensions needed for a one-sheeted hyperboloid of revolution. An additional scale factor along a direction perpendicular to the axis of revolution gives the third dimension, to fully dimension a general hyperboloid of one sheet.

The hyperbolic paraboloid is another quadric that can be generated as a ruled surface. This task is best accomplished as a bilinear Bézier patch, illustrated in Figure 4.4 and described in Section 4.2.

Outside of the quadrics, the *helicoid* is a ruled surface of some importance. All threaded fasteners have helicoidal patches. A simple helicoid (also called a *conoid*) can be defined by the parametric representation

$$x=u\cos(v),\ y=u\sin(v),\ \text{and}\ z=\mu v \tag{4.14}$$

Here, as usual, *u* and *v* are the parameters and μ is the advance per unit twist angle. The surface is generated by first taking two intersecting, perpendicular straight lines—one as a generatrix and the other as the axis. The generatrix line is then revolved helically about the axis line with the pitch μ. If μ is positive we have a right helicoid; if it is negative we have a left helicoid. Figure 4.10 illustrates a right helicoid. For generating helicoids used in threaded fasteners, the generatrix line is inclined at a fixed angle θ different from 90° to the axis line and is revolved helically about the axis with the pitch μ. The resulting helicoid has μ and θ as the two intrinsic dimensions.

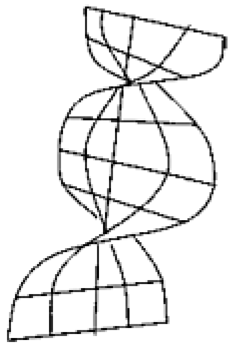


FIGURE 4.10 A right helicoid as a ruled surface.

4.4 EXERCISES

1. Determine the types of the following quadrics.

   (a) $x^2+y^2+z^2-4x+6y-14z+37=0$
   (b) $xy-xz+yz-5y=0$
   (c) $4x^2-2y^2-12z^2+12yz+4xy+4x+2y+3z=0$
   (d) $2x^2+2y^2+5z^2-2xy-4xz+4yz-2x+6z-4=0$

2. Research the use of the one-sheeted hyperboloid of revolution in constructing cooling towers. How would you dimension such a structure?
3. Research the so-called hypar (hyperbolic paraboloid) structures used in the construction of roofs. How would you dimension such hypar surface patches?
4. Prove that the converse of the free-form surface congruence theorem (Theorem 4.4) is false.
5. Prove the assertion that "a free-form surface patch with $N+1$ control points requires $3N-3$ dimensions."
6. Helical springs, helical washers, and helical threads are commonly used in mechanical engineering applications. Find our how their geometries are defined in their suppliers' technical literature. How are these definitions related to helical sweeps?

4.5 NOTES AND REFERENCES

The classification of quadrics is well covered in books on solid analytic geometry. Olmsted (1947) is a comprehensive reference on this topic. Struik (1953) and Hilbert and Cohn-Vossen (1983) make good supplemental reading. We have adopted Olmsted's quadrics type classification table. His book can be consulted for more details on the canonical reduction of quadrics. Farin (1993) is a good reference for free-form surfaces.

The counterpart of Theorem 3.6 is the fundamental existence and uniqueness theorem of surfaces. It can be found in differential geometry books, such as Lipschutz (1969). It is based on fundamental forms, which are intrinsic characteristics of surfaces. A general, elegant congruence theorem for surfaces can be formulated using these intrinsic characteristics. Unfortunately, this is of little engineering use in dimensioning surfaces.

5
Dimensioning Relative Positions of Elementary Objects

Thus far our attention has been focused on intrinsic dimensioning of elementary curves and surfaces. These intrinsic dimensions are the characteristics that remain invariant when the curve or surface is subjected to rigid motion. In exploring them using classification and congruence theorems from classical analytic geometry, we unwittingly stumbled on cases involving a pair of lines while dealing with conics and on a pair of planes when we studied quadrics.

Individually, unbounded lines and unbounded planes do not posses intrinsic dimensions. But when one line is placed relative to another line in a plane, we have the task of dimensioning their relative position. Similarly, when two planes are considered in space, the relative position of these two planes needs to be specified using a dimension. As we saw in previous chapters, these tasks are accomplished rather easily.

The conics classification theorem in Chapter 3 reduces the number of cases involving a pair of lines to just three: coincident lines, parallel distinct lines, and intersecting lines. The classification also supplied the dimensions—separating distance in the case of parallel lines and included angle for intersecting lines. An identical set of classification and dimensions was repeated in Chapter 4 for a pair of planes in the quadrics classification theorem.

We extend these results in this chapter to all pairs involving points, lines, planes, and—here is a surprise—helices. These results are useful by themselves. In addition, we will show in Chapter 7 that any two geometric objects can be positioned relative to each other using these elementary objects. That makes the results of this chapter all the more important. For these reasons, we devote this chapter exclusively to a special theory of relative positioning.

5.1 DISTANCES AND ANGLES

The distance between two points p_1 and p_2 was defined in Eq. (2.4). Note that this distance is never negative and is symmetrical; that is, $d(p_1, p_2)=d(p_2, p_1)$. When the distance is zero, the points coincide. We say that the condition that two points coincide is an *incidence* constraint imposed on the points. We will now define other distances between points, lines, and planes.

The distance between a point p and a straight line l is defined as the minimum distance between p and any point in l. That is,

$$d(p, l) = \min_{q \in l} d(p, q) \tag{5.1}$$

This distance is also nonnegative, and we will treat this distance symmetrically; that is, $d(p, l)=d(l, p)$. As long as this distance is nonzero, there is a unique plane that contains both p and l. We can drop a perpendicular from p to l, and the distance is then the distance between p and the foot of the perpendicular. See Figure 5.1(a). An incidence constraint can be invoked by demanding that the point lie on the line, in which case the distance vanishes.

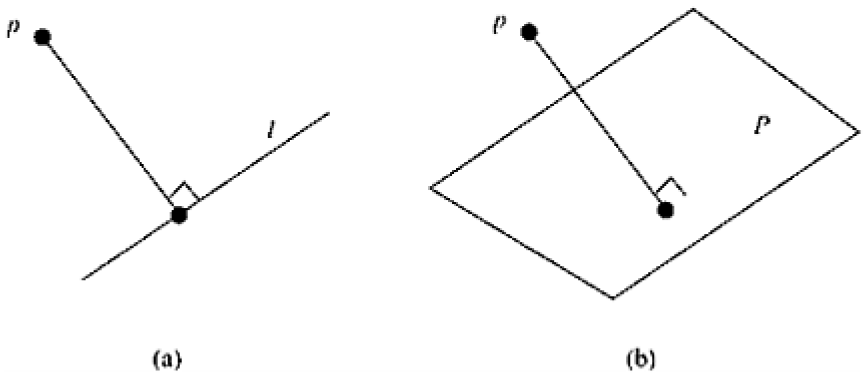


FIGURE 5.1 (a) Distance between a point and a line and (b) distance between a point and a plane.

The distance between a point p and a plane P can be defined similarly as

$$d(p, P) = \min_{q \in P} d(p, q) \tag{5.2}$$

It is the shortest distance between the point and the plane. It is also the distance between p and the foot of the perpendicular dropped from the point p to the plane P, as shown in Figure 5.1(b). The distance is nonnegative and is symmetrical; that is, $d(p, P)=d(P, p)$. When p lies on P, this distance goes to zero. The condition of the point lying on the plane is an incidence constraint.

When two lines l_1 and l_2 in a plane intersect, the angle between them is denoted $\theta(l_1, l_2)$, and it is the smaller of the two complementary angles between the lines. See Figure 5.2(a). Its value lies between 0° and 90°. If the lines are oriented (that is, directed), then they have a unique angle whose value lies between 0° and 180°, as shown in Figure 5.2 (b). The angle between an oriented line and a nonoriented line is the same as between two nonoriented lines.

5.2 SOME CASES INVOLVING POINTS

The following congruence theorems involving points are easily proved.

Theorem 5.1 *Let p_1, p_2, p'_1, and p'_2 be points, in a plane or in space. Then (p_1, p_2) is congruent to (p'_1, p'_2) if and only if $d(p_1, p_2) = d(p'_1, p'_2)$.*

Theorem 5.2 *Let p, p′ be points and l, l′ be lines, in a plane or in space. Then (p, l) is congruent to (p′, l′) if and only if d(p, l)=d(p′, l′).*

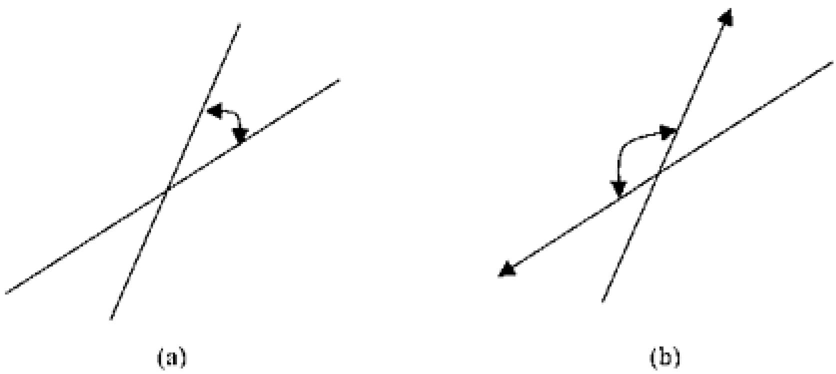


FIGURE 5.2 (a) Angle between two lines and (b) angle between two oriented (directed) lines or vectors.

Theorem 5.3 *Let p, p′ be points and P, P′ be planes in space. Then (p, P) is congruent to (p′, P′) if and only if d(p, P)=d(p′, P′).*

Therefore, the distance between two points is their relative position parameter, and its value is their relative position dimension. Similarly, the distance between a point and a line is their relative position parameter, and its value is their relative position dimension; the distance between a point and a plane is their relative position parameter, and its value is their relative position dimension.

It is noteworthy that we are able to dimension these cases involving points with just (unsigned) distances, because of the following facts of chirality.

1. A tuple of two points is achiral, either in a plane or in space. (A tuple of three distinct points is chiral in the plane that contains them, but it is achiral in space. We have seen triangles as examples of this fact. A tuple of four or more distinct points in three-dimensional space is chiral. A tetrahedron is a good example of this fact.)
2. A tuple of a point and a line is achiral, either in the plane that contains them or in space.
3. A tuple of a point and a plane is achiral.

Incidence constraints, such as a point lying on a line (denoted *p* on *l*) and a point lying on a plane (denoted *p* on *P*), reduce Theorems 5.2 and 5.3 to the following corollaries.

Corollary 5.1 *Let p, p′ be points and l, l′ be lines, in a plane or in space. Also, let there be incident constraints of p on l and p′ on l′. Then (p, l) is congruent to (p′, l′).*

Corollary 5.2 *Let p, p′ be points and P, P′ be planes in space. Also, let there be constraints of p on P and p′ on P′. Then (p, P) is congruent to (p′, P′).*

We will have need for these results later.

5.3 RELATIVE POSITIONING TWO LINES

Consider two straight lines l_1 and l_2 in space. If they both lie in a plane, then they are called *coplanar* lines. Otherwise they are called *skew* lines.

5.3.1 Coplanar Lines

Let's consider coplanar lines first. If the two lines are coincident, we have no relative positioning issue. This condition can also be invoked as an incident constraint. If the lines are distinct and parallel, then they are coplanar. This condition is also referred to as a *parallelism* constraint. The angle $\theta(l_1, l_2)$ between the parallel lines is clearly zero. In this case, it is meaningful to talk about the distance $d(l_1, l_2)$ between the lines as the shortest distance between the two lines. When two lines intersect at a point, then the two lines are coplanar. We will now show that these are the only three cases possible for a pair of lines in a plane; in each case we will find their relative position dimension.

Recall that in Chapter 3 a pair of straight lines was considered as a special case of the conics classification. A more direct theory of a system of two lines in a plane can be formulated by considering their general equations

$$\begin{aligned} a_1x+b_1y+c_1&=0\\ a_2x+b_2y+c_2&=0 \end{aligned} \qquad (5.3)$$

where at least one of a_1, b_1 is nonzero and at least one of a_2, b_2 is nonzero. Now define two ranks r and R as

$$r = \text{rank of}\begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix} \qquad \text{and} \qquad R = \text{rank of}\begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \end{bmatrix} \qquad (5.4)$$

Clearly, $R \geq r$, $\max(r)=2$, $\min(r)=1$, $\max(R)=2$, and $\min(R)=1$. (Why?) These conditions leave us with only three possibilities.

1. $r=1$, $R=1$. This is the case of two coincident lines. This is an incident constraint and we have no other relative position dimension between these two lines.
2. $r=1$, $R=2$. This is the case of two parallel, distinct lines. This is a parallelism constraint. The relative position dimension between these two lines is the distance between them because we can easily prove the following theorem.

Theorem 5.4 *Let l_1, l_2 be two parallel lines, and l'_1, l'_2 be two other parallel lines. Then*

(l_1, l_2) is congruent to (l'_1, l'_2) *if and only* $d(l_1, l_2) = d(l'_1, l'_2)$.

3. r=2, R=2. This is the case of two lines intersecting at a point. The relative position between these two lines is the angle between them because we can again prove the following theorem.

Theorem 5.5 *Let l_1, l_2 be two intersecting lines and* l'_1, l'_2 *be two other intersecting lines. Then (l_1, l_2) is congruent to* (l'_1, l'_2) *if and only if* $\theta(l_1, l_2) = \theta(l'_1, l'_2)$.

Note the close similarity between this case analysis and the classification of a pair of lines, described in Chapter 3.

5.3.2 Skew Lines

We will now consider two lines l_1 and l_2 that are skew. We can still refer to $d(l_1, l_2)$ as the shortest distance between the two lines. It is obtained by talking a point on each of the lines and searching for the minimum of the distance between them. That is, for skew lines

$$d(l_1, l_2) = \min_{p \in l_1, q \in l_2} d(p, q) \tag{5.5}$$

This minimum occurs along a unique common perpendicular line l_3 between the skew lines. It is the distance between the points of intersection of l_3 with l_1 and l_2. See Figure 5.3. Perpendicular to l_3 are two distinct parallel planes P_1 and P_2 that contain l_1 and l_2, respectively. We can then translate one of these lines, say, l_1, parallel to itself along the common perpendicular till it lies in the plane P_2 and intersects l_2; the angle between l_2 and the translated version of l_1 is called the *twist angle* $\theta(l_1, l_2)$.

Before we proceed further, we observe one interesting fact about a pair of skew lines: A *tuple of two skew lines is chiral,* unless their twist angle is 90°. That is, if you take two skew lines whose twist angle is different from 90° and weld them together by an invisible welding material, then the resulting object is chiral. Its mirror image is not congruent to the original object. Note that this is not true for the case of a pair of coplanar lines.

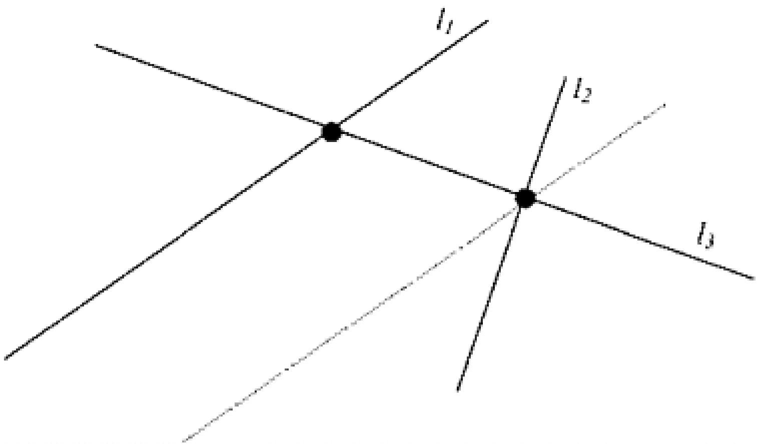


FIGURE 5.3 l_1 and l_2 are skew lines in space. l_3 is the line perpendicular to both the skew lines. The dotted line is parallel to l_1. The distance between the skew lines is the distance between the two points of intersection shown. The twist angle $\theta(l_1, l_2)$ is the angle between l_2 and the dotted line.

We can now use the method of superposition to prove the following congruence theorem involving skew lines.

Theorem 5.6 *Let l_1, l_2 be two skew lines in space and l'_1, l'_2 be two other skew lines in space. Then (l_1, l_2) is congruent to (l'_1, l'_2) if and only if they have the same chirality, $d(l_1, l_2) = d(l'_1, l'_2)$ and $\theta(l_1, l_2) = \theta(l'_1, l'_2)$.*

It is possible to encode the chirality of a tuple of skew lines in the sign of their twist angle, as shown in Figure 5.4. A positive twist angle corresponds to a counterclockwise rotation (a right-handed chirality for the tuple of skew lines), and a negative twist angle corresponds to a clockwise rotation (a left-handed chirality for the tuple of skew lines). Such encoding of chirality is common. Recall that the chirality of a helix can be encoded in the sign of its torsion (see Sec. 3.3); a positive torsion corresponds to a right-handed helix, and a negative torsion corresponds to a left-handed helix.

So the relative positioning of two lines involves two parameters: their shortest distance and the signed twist angle. Assigning numerical values to them dimensions their relative position. When the twist angle is zero, the lines are parallel and we need to dimension only the distance between them. When

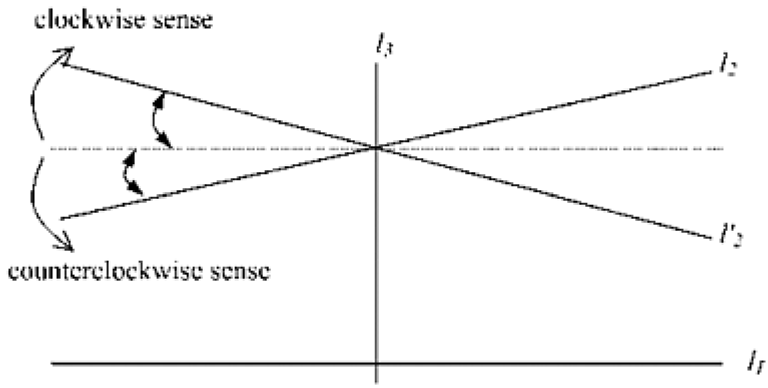


FIGURE 5.4 Assigning sign to the twist angle between skew lines. The twist angle between l_1 and l_2 is taken to be positive because, as l_1 is translated to the dotted line and twisted to coincide with l_2, the combination of translation and rotation follows the right-hand rule. One can also think of this as a counterclockwise rotation. The twist angle between l_1 and l'_2 is taken to be negative, because it follows the left-hand rule (or, equivalently, a clockwise rotation).

the shortest distance between them is zero, the lines intersect and we need to dimension only the angle between them.

5.4 RELATIVE POSITIONING A LINE AND A PLANE

When a line *l* does not lie in a plane *P,* it can either be parallel to it or intersect it at a point. If the line lies in the plane, this condition is an incidence constraint. Otherwise, if *l* is parallel to *P,* then the shortest distance between them is the distance *d(l, P),* which can be obtained by taking any point on *l* and finding its shortest distance to the plane *P*.

If *l* is perpendicular to *P,* then we define the angle between them to be θ*(l, P)*=90°. If *l* is neither parallel nor perpendicular to *P,* let *l** be the projection of *l* on *P*. We can obtain *l** by taking every point of *l* and perpendicularly projecting it on the plane *P*. Clearly *l* and *l** are coplanar and they intersect at the point of intersection between *l* and *P*. See Figure 5.5. The angle between *l* and *l** is then denoted θ*(l, P)*. With these measures, we have the following congruence theorem.

Theorem 5.7 *Let l, l′ be lines and P, P′ be planes. If l is parallel to P and l′ is parallel to P′, then (l, P) is congruent to (l′, P′) if and only if d(l, P)=d(l′, P′). If l is not parallel to P and l′ is not parallel to P′, then (l, P) is congruent to (l′, P′) if and only if θ(l, P)=θ(l′, P′).*

We are able to get away with just unsigned distances and angles here because a tuple of a line and a plane is achiral.

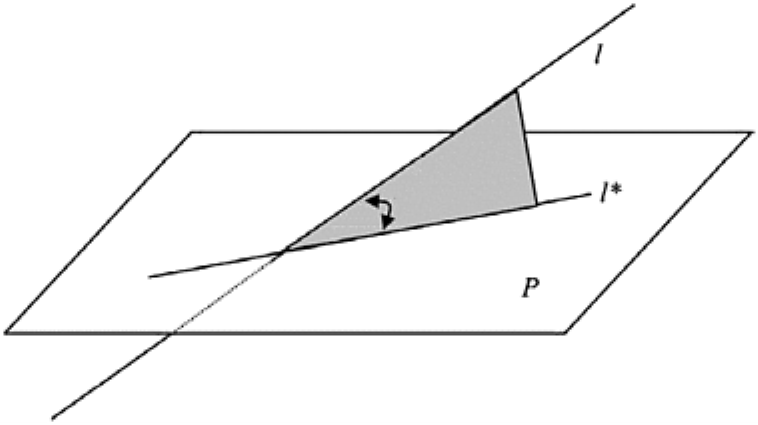


FIGURE 5.5 Angle between a line and a plane in space.

So relative positioning a line and a plane depends on a simple case analysis. If the line lies in the plane, then it is an incidence relationship and there is no further relative positioning issue. If they are not incident and if the line is parallel to the plane, then the (shortest) distance between them is the relative position parameter. If the line is not parallel to the plane, then the angle between them is the relative position parameter. In either case, we have only one dimension to deal with.

Theorem 5.7 specializes to the following corollary if we invoke incidence constraints, such as a point lying on a line (*p* on *l*) and a line lying on a plane (*l* on *P*).

Corollary 5.3 *Let p, p′ be points, l, l′ be lines, and P, P′ be planes. Also, let there be constraints of p on l, p′ on l′, l on P, and l′ on P′. Then (l, P) is congruent to (l′, P′) and (p, l, P) is congruent to (p′, l′, P′).*

5.5 RELATIVE POSITIONING TWO PLANES

If two planes P_1 and P_2 are not coincident, then they are parallel or they intersect in a line. If they are coincident, then no further relative positioning is required. This condition is also an incidence constraint. If the planes are distinct and parallel, then the shortest distance between them is the distance $d(P_1, P_2)$. It can be obtained by taking any point in one plane and finding its shortest distance to the other plane.

If the planes are distinct and not parallel, then they intersect in a line l. Any plane P perpendicular to l will intersect P_1 and P_2 in lines l_1 and l_2, respectively. Then the angle $\theta (P_1, P_2)$ is the angle between l_1 and l_2, as shown in Figure 5.6.

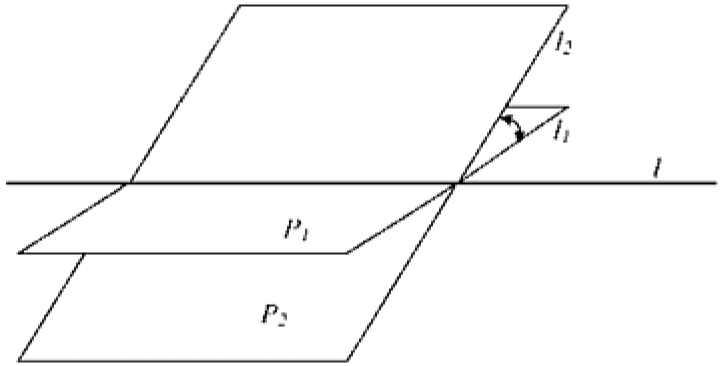


FIGURE 5.6 Angle between two intersecting planes in space.

The results just quoted are not new. They were obtained in Chapter 4 as special cases in the classification of quadrics. But we can also formulate an independent theory of a system of two planes. The procedure is the same as the one adopted in Section 5.3.1 for a system of two coplanar lines. Let the two planes be defined generally by the equations

$$a_1x+b_1y+c_1z+d_1=0$$
$$a_2x+b_2y+c_2z+d_2=0 \tag{5.6}$$

where at least one of a_1, b_1, c_1 is nonzero and at least one of a_2, b_2, c_2 is nonzero. Again, define two ranks r and R as

$$r = \text{rank of}\begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \end{bmatrix} \quad \text{and} \quad R = \text{rank of}\begin{bmatrix} a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \end{bmatrix} \tag{5.7}$$

Clearly, we have the conditions that $R \geq r$, max$(r)=2$, min$(r)=1$, max$(R)=2$, and min$(R)=1$. (Why?) We then have only the following possibilities.

1. $r=1$, $R=1$. The two planes are coincident. This is an incident constraint. We have no further relative positioning problem.
2. $r=1$, $R=2$. The two planes are distinct and parallel. This is a parallelism constraint. As we will see, the relative positioning in this case involves just the distance between the two parallel planes.
3. $r=2$, $R=2$. The two planes intersect in a line. In this case, as we will see, the relative positioning depends only on the angle between the two planes.

Based on our case analysis, we have the following congruence theorem, which can be proved easily.

Theorem 5.8 *Let P_1, P_2, P'_1, and P'_2 be planes. If P_1 is parallel to P_2 and P'_1 is parallel to P'_2 then (P_1, P_2) is congruent to (P'_1, P'_2) only if $d(P_1,$*

$P_2) = d(P'_1, P'_2)$ *If* P_1 *is not parallel to* P_2 *and* P'_1 *is not parallel to* P'_2*, then* (P_1, P_2) *is congruent to* (P'_1, P'_2) *if and only if* $\theta(P_1, P_2) = \theta(P'_1, P'_2)$.

We are able to get away with just unsigned distances and unsigned angles here because a tuple of two planes is achiral.

Therefore, the relative positioning problem of two planes reduces to the following simple cases. Distinct, parallel planes have only one relative position parameter: the distance between them. Intersecting planes have only the angle between them as the relative position parameter. In either case, we have only one dimension for their relative position. Note the close similarity of this case analysis with Chapter 4's classification of a pair of planes.

5.6 CASES INVOLVING ORIENTED LINES AND ORIENTED PLANES

In earlier sections we saw how binary information about chirality of helix and pair of skew lines can be encoded using signed values. Similarly, we can encode binary information about the so-called "material side" using oriented lines and oriented planes, as illustrated in Figure 5.7. An oriented line can be embedded in space or it can lie in a specified plane. In Figure 5.7(a) the oriented line lies in a specified plane. To one side of this line lies a two-dimensional region, the so-called material region, that is of interest to us. By convention, we assume that the material lies to the left of the oriented line. We can then define an outward normal, which is a unit vector in the plane, perpendicular to the oriented line and directed away from the material side. In Figure 5.7(b) a three-dimensional region, that is, a material region, lies on one side of an oriented plane in space. By convention, we again choose a unit normal perpendicular to the plane and directed away from the material side as the outward normal, and we also use it to orient the plane. Thus an oriented line in a plane and an oriented plane in space can be used to encode the so-called material side.

Additionally, oriented lines and oriented planes are useful to set up coordinate reference systems. These need not be full three-dimensional coordinate frames. An oriented line, for example, can be used as a one-dimensional coordinate reference line by specifying an origin point on the line. A single unit vector, such as an outward normal to a line in a plane or an outward normal to a plane, can also serve the same purpose, because it can be taken as the sole basis vector to define one-dimensional vectors. These lead to the use of signed distances and signed angles as dimensions, thus providing the theoretical basis for coordinate dimensioning first mentioned in Section 1.3. Recall that we used coordinate dimensioning for free-form curves and surfaces, as in the example of Figure 3.10(b).

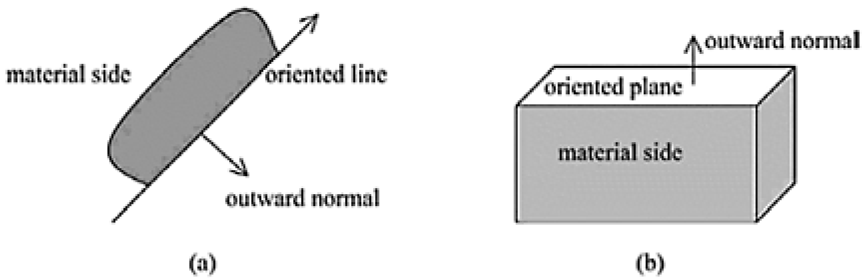


FIGURE 5.7 Encoding the material side using oriented lines and oriented planes. (a) An oriented line in a plane. The material side is assumed to be on the left of the oriented line. An outward normal is perpendicular to the oriented line and points away from the material side. (b) An oriented plane, whose normal also serves as an outward normal by pointing away from the material side.

While considering oriented lines in a plane and oriented planes in space, it is useful to consider an important fact about their chirality. An oriented plane is achiral in space. However, an oriented line embedded in a plane and associated with an outward normal, as we have already defined, is chiral in that plane. This is due to the fact that here we have also defined an outward normal with an oriented line, and these two form an orthogonal doublet that is chiral in the plane. That is, a mirror image (using a line in the plane as the "mirror") of the orthogonal doublet cannot be moved in the plane by pure rigid motion to coincide with the original doublet. We can make them coincide by flipping the doublet out of the plane; but remember, that it is not allowed.

In this section we will examine how the dimensioning problem is affected if we have oriented lines and oriented planes among elementary objects. Our examination will depend on careful case analyses. Congruence theorems to support all the cases are not difficult to state and prove, but they become somewhat repetitive and tedious. They are best left as exercises. We will denote oriented lines and oriented planes by underscores, as in an oriented line $\underline{l}$ and oriented plane $\underline{P}$.

5.6.1 Cases Involving Points

A point, obviously, has no orientation. Figure 5.8(a) shows an oriented line in a plane and two points in the same plane. It also illustrates how the relative position of each of the points can be dimensioned (or parameterized) using a signed distance in the form of a coordinate dimension. Figure 5.8(b) gives a similar illustration for dimensioning (or parameterizing) the relative position of a point and an oriented plane, again using signed distances in the form of coordinate dimensions. The sign encodes the information as to which side of the line or the plane the point lies on. In both cases, the outward unit normal serves as the basis vector and the origin lies in the oriented line or in the oriented plane.

If we have just an oriented line in space without any specified plane that contains it, we don't have any outward normal to work with. So in this case, a

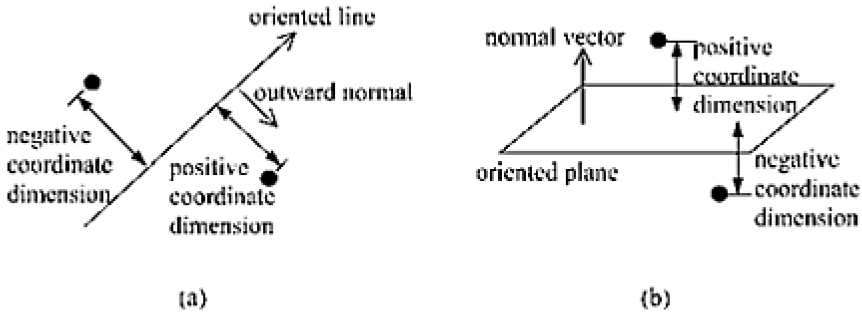


FIGURE 5.8 Positioning a point relative to (a) an oriented line and (b) an oriented plane. The relative position dimension can have a positive or negative coordinate, depending on which side of the line or plane the point is located.

point can be positioned relative to this line as though it is not oriented, as we did in Section 5.2.

5.6.2 Two Lines

Here again we need to consider two cases. In the first case the two lines, at least one of which is oriented, are coplanar. We will then assume that each oriented line also has an outward normal that lies in the plane. In the second case the two lines, at least one of which is oriented, are skew. Each oriented line in this case is embedded in space, and it does not have a unique outward normal. As we will see, this simplifies the dimensioning of oriented skew lines.

5.6.2.1 Coplanar Lines

Let's consider coplanar lines first. As already remarked, at least one of these lines is oriented. If the lines coincide and only one of them is oriented, then we have no further relative dimensioning problem. If the coincident lines are both oriented, then we need to specify whether they are oriented in the same direction or in opposite directions. Otherwise, the lines either are distinct and parallel or intersect at a point.

Figures 5.9 and 5.10 illustrate cases when the two lines are distinct and parallel. When only one of the two lines is oriented, the relative position dimension between the two parallel lines can be captured as a positive or a negative coordinate dimension, depending on which side of the oriented line the nonoriented line lies, as shown in Figure 5.9.

When both the parallel lines are oriented, they can be pointing in the same direction, the coparallel case, as in Figure 5.10(a), or in opposite directions, the

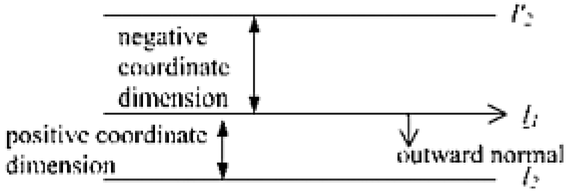


FIGURE 5.9 Dimensioning the relative position of an oriented line l_1 and a nonoriented line, such as l_2 or l'_2, when they are parallel.

antiparallel case, as in Figure 5.10(b). When they are coparallel, the distance between them can lead to positive or negative coordinate dimension, depending on which line is chosen as the origin. See Figure 5.10(a) for further details. When they are antiparallel, the relative position dimension between them is deemed to be a positive coordinate dimension if the outward normals are pointing toward each other and a negative coordinate dimension if the outward normals are pointing away from each other. Figure 5.10(b) shows these possibilities.

The cases of intersecting lines are illustrated in Figures 5.11 and 5.12. Here the angle between them is the dimension. When only one of the lines is oriented, the angle between them is the smallest angle, as defined in Figure 5.2(a) ignoring the orientation. This angle dimension is then given a positive or negative sign, depending on whether a counterclockwise or clockwise rotation

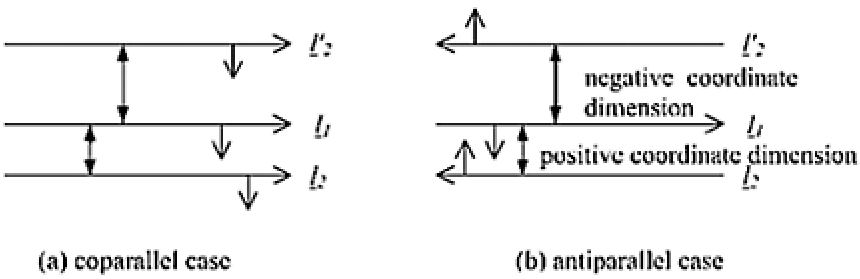


FIGURE 5.10 Dimensioning the relative position of two oriented, parallel lines. The outward normals are indicated. (a) When the lines are coparallel, the relative position dimension between l_1 and l_2 can be either a positive coordinate dimension or a negative coordinate dimension, depending on which oriented line is chosen as the origin. The same comment applies to the relative positioning of l_1 and l'_2. (b) When the lines are antiparallel, positive and negative dimensions can be assigned that do not depend on which oriented line is chosen as the origin.

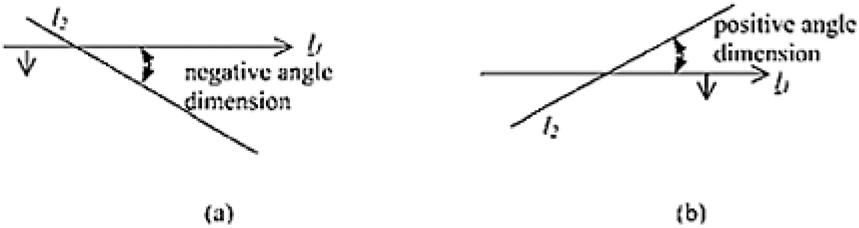


FIGURE 5.11 Dimensioning the relative position of an oriented line l_1 and a nonoriented line l_2, when they intersect at a point. The smaller angle between the two lines, ignoring the orientation of l_1, is dimensioned. (a) A clockwise rotation from the oriented line to the nonoriented line leads to the negative dimension. (b) A counterclockwise rotation from the oriented line to the nonoriented line leads to the positive dimension.

is involved in moving from the oriented line to the nonoriented line. See Figure 5.11 for more details. When both lines are oriented, the angle dimension between them is defined by Figure 5.2(b). Depending on which oriented line is chosen as the origin, the angle dimension indicated in Figure 5.12 can become a positive or a negative dimension.

5.6.2.2 Skew Lines

Compared to the coplanar case, skew lines require a surprisingly simple case analysis. When only one of the lines is oriented, then there is no special care required—we can treat both as nonoriented lines and apply the results of Section 5.3.2. When both skew lines are oriented, there are only four possible cases, illustrated in Figure 5.13. The dimensions (or parameters) are the unsigned shortest distance between the two skew lines and the signed twist angle between them. The indicated angle dimensions are positive or negative,

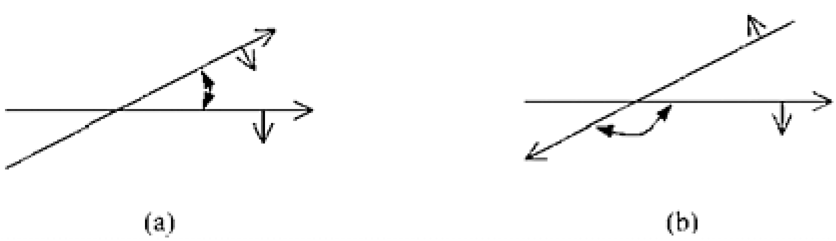


FIGURE 5.12 Dimensioning the relative position of two intersecting, oriented lines. See also Figure 5.2. Depending on which line is chosen as the origin, the angle dimension indicated can be a positive or a negative dimension.

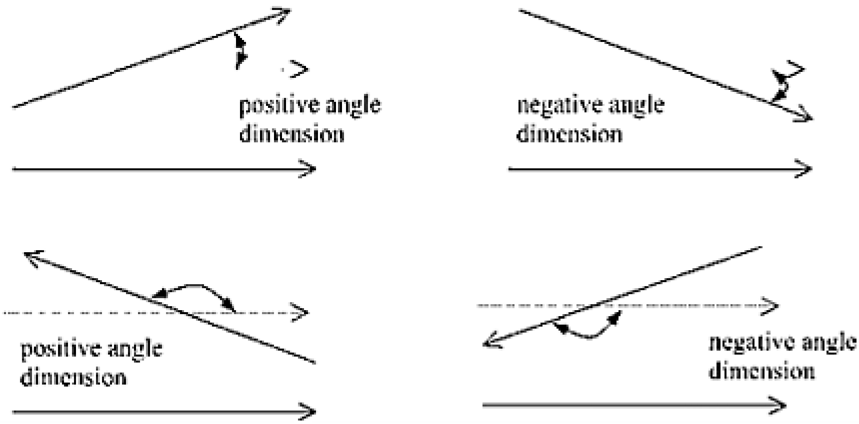


FIGURE 5.13 Dimensioning the relative position of two oriented, skew lines.

depending on whether we need a counterclockwise or clockwise rotation, as discussed previously in Section 5.3.2.

5.6.3 Line and Plane

Here we assume that at least one of the elements is oriented. When the line lies in the plane, there is no further relative positioning problem. If only the line has orientation, then its relative position with respect to a nonoriented plane is the same as that of a nonoriented line, and we can use the results of Section 5.4.

If only the plane has orientation, then we need to consider two cases.

1. If the nonoriented line is parallel to the oriented plane, then the relative position dimension between them can be given a positive or negative value, as shown in Figure 5.14. This case is very similar to that of a pair of lines illustrated in Figure 5.9.

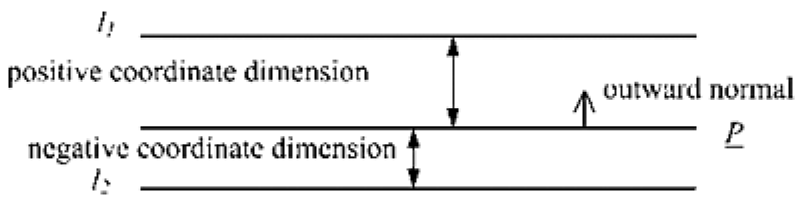


FIGURE 5.14 Dimensioning the relative position of a nonoriented line and an oriented plane when they are parallel.

2. If they intersect at a point, then the relative positioning problem is the same as that of a nonoriented line and a nonoriented plane. We then use the results of Section 5.4.

When the line and the plane are both oriented, we have the following two cases.

1. If they are parallel, then the dimensioning problem is the same as the one illustrated in Figure 5.14. We just ignore the orientation of the line.
2. If they intersect at a point, the angle dimension can be given a positive or negative value, as shown in Figure 5.15.

5.6.4 Two Planes

Here again we assume that at least one of the planes is oriented. When the planes coincide and only one of the planes is oriented, we have no further relative positioning problem. If the coincident planes are both oriented, then we have to specify whether their normals are oriented in the same direction or in opposite directions. Assume that the planes are not coincident in what follows.

If only one of the planes is oriented, then we have the following two cases.

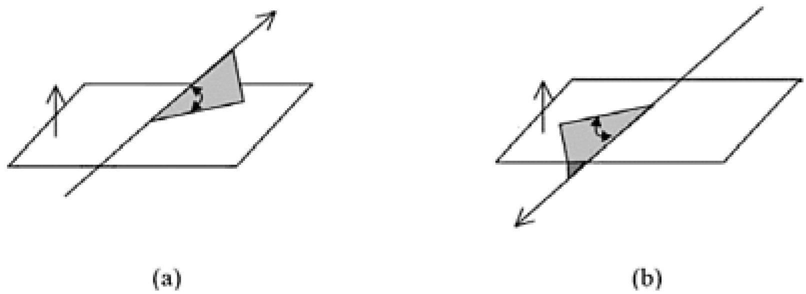


FIGURE 5.15 Dimensioning the relative position of an oriented line and an oriented plane when they intersect at a point. See Figure 5.5 for comparison. (a) The angle dimension is considered to be a positive dimension because the line orientation has a positive component along the outward normal of the plane. (b) The angle dimension is considered to be a negative dimension because the line orientation has a negative component along the outward normal of the plane.

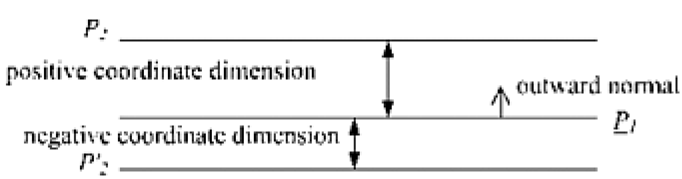


FIGURE 5.16 Dimensioning the relative position of a nonoriented plane and an oriented plane when they are parallel.

1. If the planes are distinct and parallel, then the relative position dimension between

them can be given a positive or negative value, as shown in Figure 5.16. This case is also very similar to that of a pair of lines illustrated in Figure 5.9.
2. If the planes intersect in a line, then their relative positioning problem is the same as that of two nonoriented planes discussed in Section 5.5. Note that this case is simpler than the corresponding case of oriented lines illustrated in Figure 5.11.

When both planes are oriented, we have the following two cases.

1. If the planes are distinct and parallel, then their normals can be coparallel or antiparallel, as shown in Figure 5.17. The same figure also shows how their relative position can be dimensioned. Note the similarity between this and the case of oriented, parallel lines illustrated in Figure 5.10.
2. If the planes intersect in a line, then their relative position is dimensioned by dimensioning the angle between their normals.

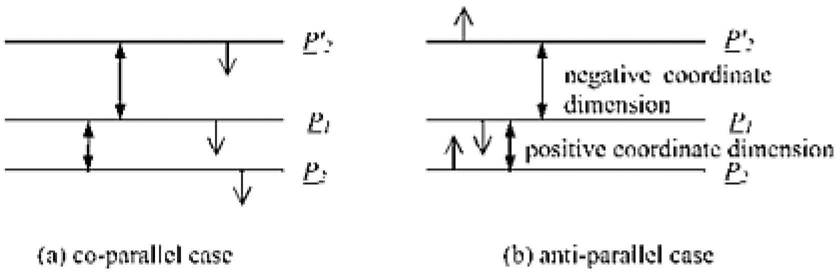


FIGURE 5.17 Dimensioning the relative position of two oriented, parallel planes. The outward normals are indicated. (a) When the normals are coparallel, the relative position dimension between $\underline{P}_1$ and $\underline{P}_2$ can be either a positive coordinate dimension or a negative coordinate dimension, depending on which oriented plane is chosen as the origin. The same comment applies to the relative positioning of $\underline{P}_1$ $\underline{P}'_2$. (b) When the normals are antiparallel, positive and negative dimensions can be assigned that do not depend on which oriented plane is chosen as the origin.

There is no need to assign any sign to this angle. Note that this case is also simpler than the corresponding case of oriented lines illustrated in Figure 5.12.

5.7 CASES INVOLVING HELICES

It is easy to make the case for points, lines, and planes to be considered elementary objects, because that is the way they are treated in classical geometry. Is it reasonable to consider helices also as elementary objects, at par with points, lines, and planes? The case for this is not obvious at the outset. We will see in the next chapter that helices play a basic theoretical role, along with points, lines, and planes. In this section we will consider the helix to the extent that we need to dimension its relative position with respect to a point, a line, a plane, or another helix.

Intrinsically, a helix is completely defined by specifying its chirality (right-handed or left-handed), the diameter of its base cylinder, and its pitch. We define the axis of the base cylinder to be the axis of the helix. See Figure 5.18. There is an important one-to-one mapping between points on the axis of the helix and points on the helix. This mapping is also illustrated in

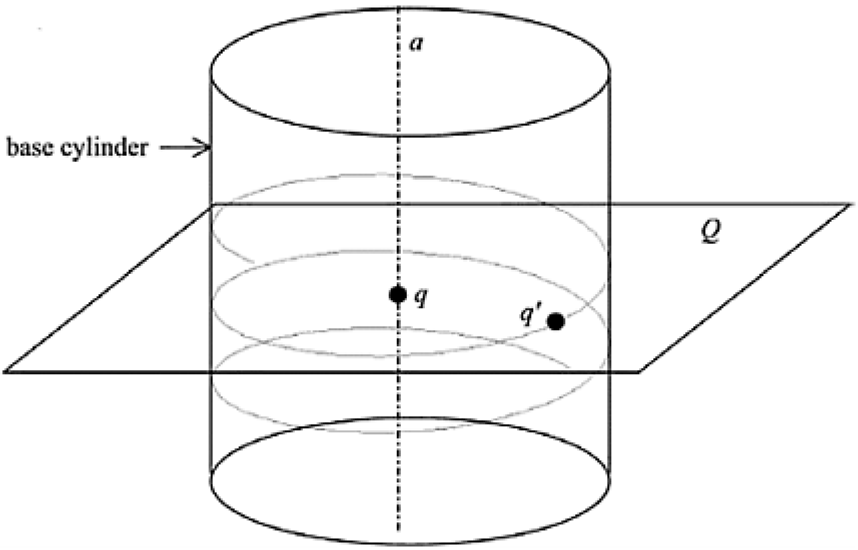


FIGURE 5.18 Mapping between a point q on the axis a of a helix and a point q' on the helix. The plane Q is perpendicular to the axis a and intersects it at q. The plane also intersects the helix at the unique point q'.

Figure 5.18. Through any point q on the axis a of the helix, a unique plane Q perpendicular to the axis can be constructed. This plane intersects the helix at one and only one point q'. Similarly, for any point q' on the helix, we can find its unique projection q on to the axis. Thus we have established a one-to-one mapping between q and q'. We call q the *image* of q', and q' the *preimage* of q.

In addressing the dimensioning problem, we perform a series of case analyses. Congruence theorems to support all these cases are not difficult, but they are somewhat tedious and are best left as exercises.

5.7.1 Point and Helix

Let p be a point and h be a helix. If p lies on the axis of h, we will treat this as an incidence constraint, and there is no further relative positioning issue. If p does not lie on the axis of h, then the relative position of p and h is dimensioned as in Figure 5.19. It involves the following two dimensions (or, equivalently, parameters).

1. The distance between *p* and its projection *q* onto the axis of the helix. This dimensions the relative position between *p* and the axis of the helix.
2. The unsigned angle between *qp* and *qq′*, where *q′* is the preimage of *q*. This dimensions the relative position between two intersecting,

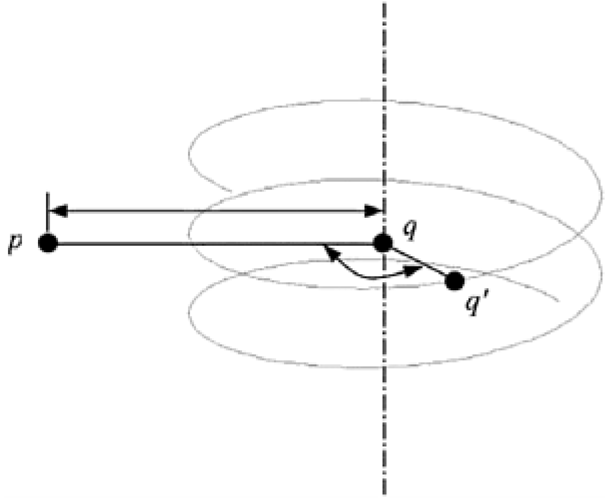


FIGURE 5.19 Dimensioning the relative position of a point *p* and a helix. *q* is the projection of *p* onto the axis of the helix. On the helix, *q′* is the preimage of *q*.

directed lines: one line directed from *q* to *p,* and the other line directed from *q* to *q′*. Also see Figure 5.2(b) for the definition of the angle between directed lines.

5.7.2 Line and Helix

Let *l* be a line and *h* be a helix. Assume that *l* does not coincide with the axis *a* of the helix. We then have three cases.

1. If *l* is parallel to, but distinct from, the axis *a* of the helix, then dimensioning the relative position between *l* and *h* is the same as dimensioning the relative position between *l* and *a*. This involves only one dimension (or parameter).
2. If *l* and *a* intersect at a point *q,* then the dimensioning can be split into two tasks:

   (a) Dimensioning the relative position between the two intersecting (nonoriented) lines *a* and *l*. This involves only one dimension (or parameter).
   (b) Dimensioning the relative position between two oriented lines. One of them is the oriented line directed from *q* to its preimage *q′* on the helix. The other oriented line is denoted $\underline{l}^*$ in Figure 5.20 and is defined as follows. Consider the plane that contains *a* and *l,* and let $\underline{l}^*$ be a line perpendicular to this plane and passing

through *q*. The orientation of *l̲** is determined by the right-hand rule as we apply a rotation from *a* to *l*. This involves only one dimension (or parameter).

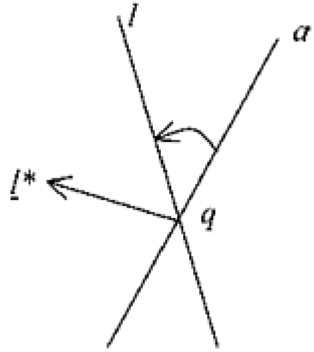


FIGURE 5.20 Defining an oriented, common perpendicular *l̲** to *a* (axis of a helix) and *l* (a line). *a* and *l* interset at *q*. The orientation of *l̲** is determined by applying the right-hand rule. As we apply a rotation from *a* to *l* (via the smallest angle between the two) with our right-hand fingers, *l̲** points along the right-hand thumb.

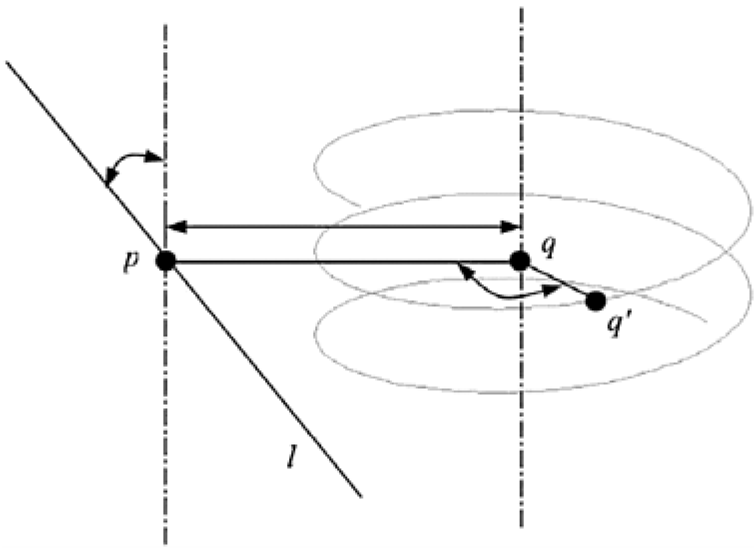


FIGURE 5.21 Dimensioning the relative position of a line l and a helix. *pq* is the common perpendicular between *l* and the axis of the helix. On the helix, *q′* is the preimage of *q*.

So a total of two dimensions (or parameters) are needed in this second case.

3 If *l* and *a* are skew, then again the dimensioning can be split into two tasks. See Figure 5.21 for an illustration. The common perpendicular between *l* and *a* passes through $p \in l$ and $q \in a$ On the helix, q' is the preimage of *q*.

(a) Dimensioning the relative position between the two skew lines *a* and *l*. This involves two dimensions (or parameters). Note that chirality is important here.
(b) Dimensioning the relative position between the directed line from *q* and *p* and the directed line from *q* and q'. This involves just one dimension (or parameter).

So a total of three dimensions (or parameters) are involved in this third case.

5.7.3 Plane and Helix

Let *P* be a plane and *h* be a helix. Assume that the axis *a* of the helix does not lie completely in *P*. Then there are only two possibilities.

1. *a* is parallel to *P*. This reduces the dimensioning problem to that of dimensioning the relative position between line *a* and plane *P*. This involves just one dimension (or parameter).
2. *a* intersects *P* at a point *q*. If *a* is perpendicular to *P,* then there is no further dimensioning issue. Otherwise, we can split the dimensioning problem into two tasks:

(a) Dimensioning the relative position between line *a* and plane *P*. This involves just one dimension (or parameter).
(b) Dimensioning the relative position between two oriented lines. One of them is the oriented line directed from *q* to its preimage q' on the helix. The other oriented line is denoted l^* in Figure 5.22 and is defined as follows. Let l^* be a line in *P* that is also perpendicular to *a*. The orientation of l^* is determined by the right-hand rule as we apply a rotation from *a* to the projection of *a* on to *P*. This also involves just one dimension (or parameter).

So a total of two dimensions (or parameters) are involved in this second case.

5.7.4 Two Helices

Let h_1 and h_2 be the two helices whose axes are a_1 and a_2, respectively. We have the following four cases based on the relative position between a_1 and a_2.

1. a_1 and a_2 are coincident. Let *q* be any point on the coincident axis, q'_1 be its preimage in h_1, and q'_2 be its preimage in h_2. We then have to

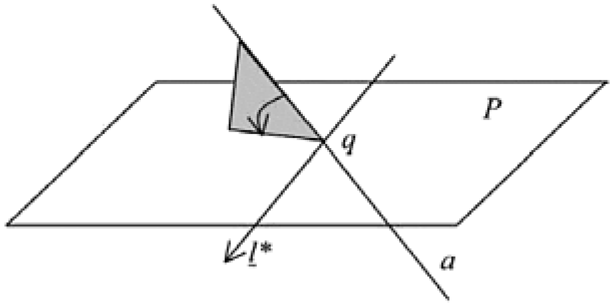


FIGURE 5.22 Defining an oriented line $\underline{l}^*$ lying in a plane P and perpendicular to a (axis of a helix). P and a intersect at q. The orientation of $\underline{l}^*$ is determined by applying the right-hand rule. As we apply a rotation from a to the projection of a onto P (via the smallest angle between the two) with our right-hand fingers, $\underline{l}^*$ points along the right-hand thumb.

dimension only the relative position between two coplanar, directed lines: one directed from q to q'_1 and the other directed from q to q'_2. This involves only one dimension (or parameter).

2. a_1 and a_2 are distinct but parallel. Let q_1 be any point on a_1 and q_2 be its projection onto a_2. Also let q'_1 be the preimage of q_1 in h_1 and q'_2 be the preimage of q_2 in h_2. The dimensioning problem can be split into two parts.

   (a) Dimensioning the relative position between a_1 and a_2. This involves just one dimension (or parameter).

   (b) Dimensioning the relative position between two coplanar, directed lines: one directed from q_1 to q'_1 and the other directed from q_2 to q'_2. This also involves just one dimension (or parameter).

So this second case involves a total of two dimensions (or parameters).

3. a_1 and a_2 intersect at a point q. Let q'_1 be the preimage of q in h_1 and q'_2 be the preimage of q in h_2. Now arbitrarily assign orientations to a_1 and a_2, and denote the oriented lines (that is, axes) as $\underline{a}_1$ and $\underline{a}_2$. Let $\underline{l}^*$ be an oriented line that is perpendicular to the plane containing $\underline{a}_1$ and $\underline{a}_2$ and passing through q. The orientation of $\underline{l}^*$ is determined by the right-hand rule as we apply a rotation from $\underline{a}_1$ to $\underline{a}_2$. Then the dimensioning problem is split into the following three tasks.

   (a) Dimensioning the relative position between two intersecting, oriented lines $\underline{a}_1$ and $\underline{a}_2$. This involves just one dimension (or parameter).

   (b) Dimensioning the relative position of the line joining q and q'_1 as a polar angle from $\underline{l}^*$. This is a polar coordinate dimension and its value lies in the interval [0, 2π). Figure 5.23 illustrates

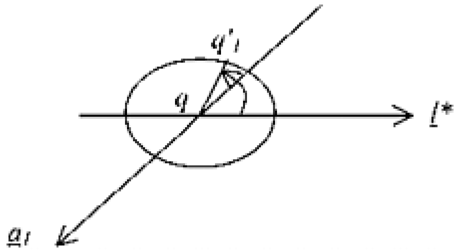


FIGURE 5.23 Polar angle dimensioning. The polar angle is between the line joining q and q'_1 and the directed line $\underline{l}$*. It is measured from the directed line $\underline{l}$* in the counterclockwise direction determined by the oriented line $\underline{a}_1$.

how this angle is determined. This also involves just one dimension (parameter).

(c) Dimensioning the relative position of the line joining q and q'_2 as a polar angle from $\underline{l}$*. This is again a polar coordinate dimension and can be determined as explained earlier. This also involves just one dimension (parameter).

So this third case involves a total of three dimensions (or parameters).

4. a_1 and a_2 are skew. Let their common perpendicular intersect a_1 at q_1 and a_2 at q_2. Also, let q'_1 be the preimage of q_1 in h_1 and q'_2 be the preimage of q_2 in h_2. Now arbitrarily assign orientations to a_1 and a_2, and denote the oriented lines (that is, axes) as $\underline{a}_1$ and $\underline{a}_2$. Let $\underline{l}$* be the oriented, common perpendicular to $\underline{a}_1$ and $\underline{a}_2$. The orientation of $\underline{l}$*

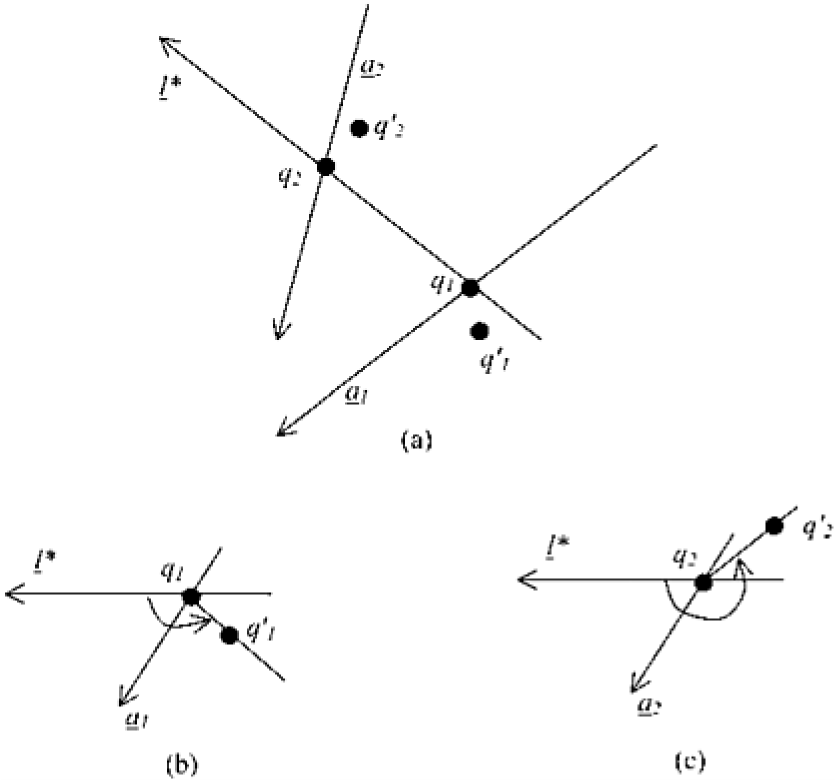


FIGURE 5.24 Polar coordinate dimensioning employed while dimensioning the relative position of two helices. (a) A general position of the two helices. (b) Polar coordinate dimension for q'_1. (c) Polar coordinate dimension for q'_2.

is determined by directing it from q_1 to q_2. See Figure 5.24(a). The dimensioning problem can be split into the following three tasks.

(a) Dimensioning the relative position between the oriented skew lines $\underline{a}_1$ and $\underline{a}_2$. This involves two dimensions (or parameters).

(b) Dimensioning the relative position of the line joining q_1 and q'_1 as a polar angle from $\underline{l}$*. See Figure 5.24(b). This is a polar coordinate dimension and it involves just one dimension (or parameter).

(c) Dimensioning the relative position of the line joining q_2 and q'_2 as a polar angle from $\underline{l}$*. See Figure 5.24(c). This is also a polar coordinate dimension and it involves just one dimension (or parameter).

So this fourth case involves a total of four dimensions (or parameters).

5.8 SUMMARY

Table 5.1 summarizes the maximum number of dimensions needed in relative positioning points, lines, planes, and helices. A number in a cell in the table is the maximum number of dimensions needed for relative positioning an element from the corresponding first row and an element from the corresponding first column. These numbers are the maximum in the sense we may need less if the relative positions of the elements are constrained in some manner. For example, if two lines are coplanar, we don't need the two dimensions required for skew lines and indicated in the table; we can do with just one. These special cases have been analyzed in detail in the indicated sections.

TABLE 5.1 Summary of the Maximum Number of Dimensions Needed for Relative Positioning Elementary Objects

	Point	Line	Plane	Helix
Point	1 (Sec. 5.2)	1 (Sec. 5.2)	1 (Sec. 5.2)	2(Sec. 5.7.1)
Line		2 (Sec. 5.3)	1 (Sec. 5.4)	3 (Sec. 5.7.2)
Plane			1 (Sec. 5.5)	2 (Sec. 5.7.3)
Helix				4 (Sec. 5.7.4)

Also indicated are the section numbers that cover these cases. The table is symmetric.

The treatment of helices required consideration of oriented lines, which explains why the cases involving oriented lines were introduced ahead of helices. In most engineering drawings and geometric models, helices are treated in a simpler manner than found in this chapter. Helical threads, for example, are seldom modeled with all the geometrical details; instead, simple stylized indications are used to convey the information that there are some standardized helical threads. In relative positioning problems, these helical elements are then treated as lines that correspond to their axes. We have provided a detailed treatment of helices mainly for theoretical completeness. It also turns out that there are some special applications, such as Archimedes screw pumps, where the details involving helices presented here are useful.

5.9 EXERCISES

1. In a plane, let l be a line defined by $ax+by+c=0$ and p be a point with coordinates (x_0, y_0). Set up a minimization problem per Eq. (5.1) and solve it to find the distance $d(p, l)$ in terms of a, b, c, x_0, and y_0. Show that this is a unique solution.
2. Repeat Exercise 1 to find the distance between a point and a plane per the minimization of Eq. (5.2).

3. Euclid's fifth postulate, known as the parallel postulate, is equivalent to saying "Given any straight line and a point not on it, there exists one and only straight line that passes through the point and parallel to the first line." What is the relationship between the parallel postulate and Theorem 5.4?
4. Give a carefully argued proof of Theorem 5.6.
5. State and prove a theorem in support of the relative positioning of a point and a helix, presented in Section 5.7.1.
6. How would you deal with the relative positioning of an oriented line and a helix? Also, how would you deal with the relative positioning of an oriented plane and a helix?

5.10 NOTES AND REFERENCES

Olmsted (1947) provides a brief theory of systems of planes. Oriented lines and planes are discussed in Struik (1953). Use of the rank of matrices to analyze system of lines in a plane and system of planes in space provides a sound theoretical basis for some of the case analyses presented in this chapter. Skew lines pose an interesting challenge, and the chirality of skew lines is not well known.

Screw threads provided the first practical use of helical geometry. Archytas of Tarentum (428 BC–350 BC), a contemporary of Plato, is considered to be the inventor of screw threads, which were initially used in presses to extract oil from olives and wine from grapes. Archimedes (287 BC–212 BC) is credited with inventing the famous Archimedes screw used to pump water. High precision helical threads were not made in mass quantities until the late 18th century. This enabled precision instruments to be made, which then permitted the manufacture of steam engines and machine tools.

6
Symmetry

In our first encounter with rigid motion in Chapter 2 it was shown that, in general, it contains three independent translations and three independent rotations. It was also observed at that time that a rigid body has a maximum of six degrees of freedom in three-dimensional space. But sometimes it can have less. For example, a sphere seems to have only three degrees of freedom, owing to its spherical symmetry, which renders all rotations about its center irrelevant as far as its geometry is concerned. This has an important practical consequence because it affects the way we dimension the relative position of a sphere with respect to, say, a plane.

Symmetry is commonly associated with reflexive symmetry, described in Chapter 2. But we want to focus on a particular class of symmetry involving rigid motions. So we want to look more closely at rigid motions and how geometric objects with various symmetries behave under the rigid motion. Recall that a rigid motion is a transformation that moves a geometric object to a congruent copy of the same object in Euclidean space. The collection of rigid motions is an example of a more general mathematical abstraction called *groups*. Appendix 2 gives a brief introduction to groups, which should be read along with this chapter.

6.1 GROUPS

A collection G of elements with a binary group operation • is called a group if the following axioms are satisfied.

Axiom G1: Closure *For any* $g_1, g_2 \in G$, $g_1 \bullet g_2 \in G$.
Axiom G2: Associativity $g_1 \bullet (g_2 \bullet g_3) = (g_1 \bullet g_2) \bullet g_3$.
Axiom G3: Identity *There exists an identity element* $e \in G$ *such that* $g \bullet e = e \bullet g = g$ *for all* $g \in G$.
Axiom G4: Inverse *For each* $g \in G$ *there is an inverse element* $g^{-1} \in G$ *such that* $g \bullet g^{-1} = g^{-1} \bullet g = e$.

It can be shown that if an identity exists then it is unique and that if an element g has an inverse then it is unique. In these terms, then, a group is a collection of elements that is algebraically closed under an associative binary operation with an identity element and with each element having an inverse. Often, explicit representation of the group operation is suppressed, so we will write g_1g_2 rather than $g_1 \bullet g_2$ if the operation is clear from the context. Note that the group operation need not be commutative. So g_1g_2 can be different from g_2g_1, and hence the order of application is important. We will now examine several examples of groups that are of interest to us.

Example 6.1 All $n \times n$ invertible real matrices form a group, with the matrix multiplication as the group operation. Going through Axioms G1 through G4 in reverse order, we observe the following. By assumption each matrix has an inverse. The identity matrix I (with unity in all diagonal elements and zeros everywhere else) serves as the identity element. Matrix multiplication is associative. The product of two invertible real matrices is invertible.

This group of $n \times n$ invertible real matrices is called the *general linear group* and is denoted by $Gl(n, \mathbb{R})$. Here $\mathbb{R}$ stands for the set of real numbers between $-\infty$ and $+\infty$.

Example 6.2 All $n \times n$ real orthogonal matrices form a group under matrix multiplication. Recall that a matrix is orthogonal if its transpose is its inverse. Again, going through Axioms G1 through G4 in reverse order, we observe the following. The existence of the transpose ensures the existence of the inverse. The identity matrix is also orthogonal. Orthogonal matrix multiplication is, of course, associative. Not so obvious is the fact that the product of two orthogonal matrices is also orthogonal.

The group of $n \times n$ real orthogonal matrices is called the *orthogonal group* and is denoted by $O(n, \mathbb{R})$. Since the orthogonal matrices are only special cases of invertible square matrices, the orthogonal group is a *subgroup* of the general linear group.

In general, if H is a subset of a group G and H by itself satisfies all the conditions to be a group, then H is said to be a *subgroup* of G. Of particular interest to us are *normal subgroups,* which are defined as follows. If $g \in G$ and B is a subset of G, let $gB = \{gb : b \in B\}$ and $Bg = \{bg : b \in B\}$ If B and C are subsets of G, then let $BC = \{bc : b \in B, c \in C\}$ If H is a subgroup of G, then H is said to be *normal* if $gH=Hg$ for all $g \in G$.

We will often speak of a product of groups. This product has to be defined carefully. It is not the same as group operation or matrix multiplication. If H is a normal subgroup of G and K is a subgroup of G, then $HK=KH$ is a subgroup of G, and we say that HK or KH is the *product* of H and K. Note that the order is not important in this product.

Sometimes we may deal with two groups defined by different symbols but whose group structures are identical. This idea is formally captured by the notion of *isomorphism*. If f is a one-to-one mapping from a group G with binary operation $\circ$ onto a group G' with binary operation $\circ'$ such that $f(x \circ y) = f(x) \circ' f(y)$ for all x and y in G, then f is an isomorphism and G and G' are isomorphic. For the purposes of group theory, isomorphic groups are equivalent.

Our coverage of groups in this section is all too brief. As mentioned earlier, Appendix 2 is devoted to groups. The reader is encouraged to review that appendix while reading this chapter.

6.1.1 Rotation Group

The 3×3 square matrix A in Eq. (2.1) was defined as a rotation matrix. It is a representation of rotation group, which is a subgroup of the orthogonal group. Note that the determinant of an orthogonal matrix can only be +1 or −1. In the following example, we consider those orthogonal matrices whose determinants are +1.

Example 6.3 All $n \times n$ real orthogonal matrices whose determinants are +1 form a group under matrix multiplication. This follows the reasoning in Examples 6.1 and 6.2 and the fact that the determinant of a product of two matrices is the product of their determinants.

Such a group is called the *special orthogonal group* and is denoted by $SO(n, \mathbb{R})$. Note that its complement in the orthogonal group, namely, all $n \times n$ real orthogonal matrices whose determinants are −1, do not form a group. (Why?). $SO(3, \mathbb{R})$ is of special interest to us because the rotation matrix in Eq. (2.1) belongs to this group. We often shorten the notation and use just $SO(3)$ to denote this rotational group in three-dimensional space. Let's look at several subgroups of $SO(3)$.

Example 6.4 Consider a 1-parameter family of 3×3 matrices defined by

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (6.1)$$

where ϕ is an arbitrary real number. Geometrically these matrices represent all rotations about the z-axis and ϕ can be viewed as the angle of rotation expressed in radians. It can be easily verified that these matrices form a group because they meet all four conditions. (Verify this.) These are also orthogonal matrices whose determinants equal +1 for any real value of ϕ. Since these matrices are a special case of 3×3 orthogonal real matrices that have +1 for their determinant, these matrices form a subgroup of $SO(3)$.

Example 6.4 demonstrates that all rotations about the z-axis form a group. Can we then say that rotations about any arbitrary, but fixed, line in space also form a group? Intuitively the answer is yes, because, via a mere change of coordinate bases, we can make this line as the z-axis without loss of any apparent generality. More formally we state that the rotations about any arbitrary but fixed axis is *isomorphic* to the rotations about the z-axis. So both are groups, and they are completely equivalent from an abstract group theoretic point of view.

Example 6.4 illustrates a continuous subgroup of $SO(3)$, in the sense that the matrix elements in Eq. (6.1) vary continuously due to the smoothness of the trigonometric functions. It is also a 1-dimensional subgroup because, loosely speaking, it belongs to a 1-parameter family. But continuity is not necessary to qualify as a group, as the following example shows.

Example 6.5 Consider the set of rotations about the positive z-axis (i.e., counterclockwise rotation with respect to the positive z-axis) by an angle from the finite set (0°, 90°, 180°, 270°}. In matrix form this can be represented as the set of just four rotation matrices

$$\left\{\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix},\begin{bmatrix}0&-1&0\\1&0&0\\0&0&1\end{bmatrix},\begin{bmatrix}-1&0&0\\0&-1&0\\0&0&1\end{bmatrix},\begin{bmatrix}0&1&0\\-1&0&0\\0&0&1\end{bmatrix}\right\} \tag{6.2}$$

This set forms a group, as can be easily verified. This is a finite subgroup of $SO(3)$ because the matrices are orthogonal and their determinants equal+1. But it is not a continuous subgroup of $SO(3)$

It is important to emphasize that the rotation angles encountered in these examples are not the groups; the matrices parameterized by the angles are the groups. Example 6.5 is a special case of the *cyclic group,* which can be defined as the set of n matrices

$$\begin{bmatrix}\cos\phi_i & -\sin\phi_i & 0\\ \sin\phi_i & \cos\phi_i & 0\\ 0 & 0 & 1\end{bmatrix}, \qquad i = 0, 1, \ldots, n-1 \tag{6.3}$$

where $\phi_i = (360/n)i$ degrees, or $(2\pi/n)i$ radians. It is a finite subgroup of $SO(3)$ and is often denoted C_n. So the matrices of Eq. (6.2) represent C_4.

Example 6.6 Augment the matrices of Example 6.5 by four additional matrices so that we have a set of eight special orthogonal matrices

$$\left\{\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix},\begin{bmatrix}0&-1&0\\1&0&0\\0&0&1\end{bmatrix},\begin{bmatrix}-1&0&0\\0&-1&0\\0&0&1\end{bmatrix},\begin{bmatrix}0&1&0\\-1&0&0\\0&0&1\end{bmatrix},\begin{bmatrix}1&0&0\\0&-1&0\\0&0&-1\end{bmatrix},\begin{bmatrix}0&1&0\\1&0&0\\0&0&-1\end{bmatrix},\begin{bmatrix}-1&0&0\\0&1&0\\0&0&-1\end{bmatrix},\begin{bmatrix}0&-1&0\\-1&0&0\\0&0&-1\end{bmatrix}\right\} \tag{6.4}$$

The last four are rotation matrices, each obtained by a 180° rotation (half-turn) about a line in the *xy*-plane. As shown in Figure 6.1(a), these lines are the *x*-axis, the line l_1, the *y*-axis, and the line l_3, respectively. The set of matrices in Eq. (6.4) forms a group. It is a finite subgroup of *SO*(3).

Example 6.6 is a special case of the *dihedral group,* which can be represented as the set of *2n* matrices

$$\begin{bmatrix} \cos\phi_i & -\sin\phi_i & 0 \\ \sin\phi_i & \cos\phi_i & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} \cos\phi_i & \sin\phi_i & 0 \\ \sin\phi_i & -\cos\phi_i & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (6.5)$$
$$i = 0, 1, \dots, n-1$$

where $\phi_i = (360/n)i$ degrees, or $(2\pi/n)i$ radians. The second matrix in Eq. (6.5) is a rotation matrix that represents a 180° rotation (half-turn) about a line l_i in the *xy*-plane, illustrated in Figure 6.1(b). The entire set of *2n* matrices in Eq. (6.5) forms a group. It is a finite subgroup of *SO*(3) and is often denoted by D_n. So the matrices of Eq. (6.4) represent D_4.

Returning to the full SO(3), one way to parameterize it is as follows.

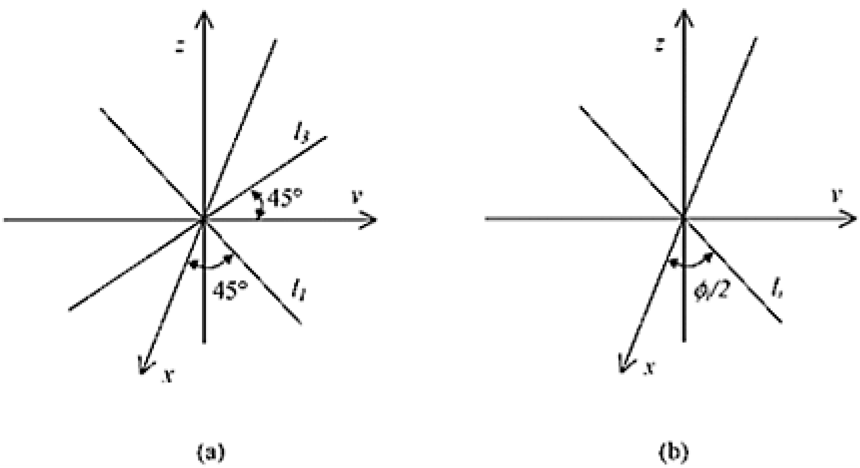


FIGURE 6.1 Lines in the *xy*-plane that serve as rotation axes to obtain the dihedral groups.

Example 6.7 Consider a 3-parameter family of 3×3 matrices

$$\begin{bmatrix} \cos\psi\cos\phi - \cos\theta\sin\phi\sin\psi & -\sin\psi\cos\phi - \cos\theta\sin\phi\cos\psi & \sin\theta\sin\phi \\ \cos\psi\sin\phi + \cos\theta\cos\phi\sin\psi & -\sin\psi\sin\phi + \cos\theta\cos\phi\cos\psi & -\sin\theta\cos\phi \\ \sin\psi\sin\theta & \cos\psi\sin\theta & \cos\theta \end{bmatrix} \quad (6.6)$$

parameterized by the popular Euler angles ϕ, θ, and ψ, called *precession, nutation,* and *spin,* respectively. These angles were introduced by Euler to study the motion of gyroscopes. These matrices are orthogonal, and their determinants equal +1 for all values of the Euler angles. They form a group because they satisfy all four conditions. (Verify this.) Therefore they are *SO*(3).

There are other ways to parameterize *SO*(3), one of which follows.

Example 6.8 Consider a 3-parameter family of 3×3 matrices

$$\frac{1}{1+\lambda^2}\begin{bmatrix} 1+\lambda_1^2-\lambda_2^2-\lambda_3^2 & 2(\lambda_1\lambda_2+\lambda_3) & 2(\lambda_1\lambda_3-\lambda_2) \\ 2(\lambda_1\lambda_2-\lambda_3) & 1-\lambda_1^2+\lambda_2^2-\lambda_3^2 & 2(\lambda_2\lambda_3+\lambda_1) \\ 2(\lambda_1\lambda_3+\lambda_2) & 2(\lambda_2\lambda_3-\lambda_1) & 1-\lambda_1^2-\lambda_2^2+\lambda_3^2 \end{bmatrix} \quad (6.7)$$

parameterized by the Rodrigues parameters λ_1, λ_2, and λ_3 and $\lambda^2 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2$ These matrices are orthogonal and their determinants equal +1 for all values of the Rodrigues parameters. They form a group because they satisfy all four conditions. (Verify this.) Therefore they are *SO*(3).

6.1.2 Translation Group

Although $n \times n$ matrices provide excellent representation for groups, vectors can also serve as groups.

Example 6.9 All *n*-dimensional real vectors form a group, with the group operation being vector addition. This group is actually a vector space, usually denoted by $\mathbb{R}^n$ Here the identity element is the zero vector. When two vectors are added the result is another vector. Vector addition is obviously associative.

The inverse of a vector is just its negative.

$\mathbb{R}^3$ is of special interest to us because our translation vector in Eq. (2.1) belongs to this group. It can be represented by a 3-parameter family of vectors $[d_x, d_y, d_z]^T$, where $d_x, d_y, d_z \in \mathbb{R}$ We will also use *T*(3) to denote this group of translations in three-dimensional space.

> **Example 6.10** Consider a 1-parameter family of translations represented by the vector $[0, 0, d]^T$, where $d \in \mathbb{R}$ This is all translations along the *z*-axis. Vector addition is the group operation. Closure and associativity are easy to establish. The identity element is the zero vector $[0, 0, 0]^T$. The inverse of $[0, 0, d]^T$ is given by $[0, 0, -d]^T$. Therefore this is a group, and it is a continuous subgroup of *T*(3). It is also a 1-dimensional subgroup because it belongs to a 1-parameter family.

The group of translations along the *z*-axis is isomorphic to the group of translations along any arbitrary but fixed straight line in space.

> **Example 6.11** Consider a 2-parameter family of translations represented by the vector $[d_x, d_y, 0]^T$, where $d_x, d_y \in \mathbb{R}$ This is all translations in the *xy*-plane. Again, vector addition is the group operation. Closure and associativity are easy to establish. The identity element is the zero vector $[0, 0, 0]^T$. The inverse of $[d_x, d_y, 0]^T$ is given by $[-d_x, -d_y, 0]^T$. Therefore this is a group, and it is a continuous subgroup of *T*(3). It is a 2-dimensional subgroup because it belongs to a 2-parameter family.

The group of translations in the *xy*-plane is isomorphic to the group of translations in any arbitrary but fixed plane in space. A discrete subgroup of *T*(3) follows.

> **Example 6.12** Consider the set of translations represented by the vector $[0, 0, d]^T$, where $d \in \mathbb{Z}$ Here $\mathbb{Z}$ is the set of all integers (positive and negative, including 0). Following the same reasoning as in Example 6.10, we see that this set is a group. It is a discrete subgroup of *T*(3).

6.1.3 Rigid Motion Group

The rigid motion group, which is often denoted simply by *R,* can now be defined as the product of the translation group *T*(3) and the rotation group *SO*(3). This is possible because we can show that *T*(3) is a normal subgroup of *R*. The structure of the rigid motion group is better revealed by matrices, as we show in the following.

A rigid motion can be represented by the 4×4 matrix in Eq. (2.9), which used homogeneous coordinates to represent points. Using this matrix notation we can show that the set of all rigid motions is a group under a group operation that composes two successive rigid motions. To show this we will denote a 3×3 rotation matrix by *A* and a 3×1 translation vector by *t,* so the 4×4 rigid motion matrix can be written as

$$R = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix} \tag{6.8}$$

where it is understood that the zero extends for three columns. If R_1 and R_2 are two rigid motions, then applying them in sequence is equivalent to the matrix multiplication

$$R_2R_1 = \begin{bmatrix} A_2 & t_2 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} A_1 & t_1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} A_2A_1 & A_2t_1 + t_2 \\ 0 & 1 \end{bmatrix} \tag{6.9}$$

The result is a rigid motion, because the product of two rotation matrices is a rotation matrix and the product of a rotation matrix and a translation vector gives a translation vector. The associativity of rigid motions is ensured by that of the matrix multiplication. The identity motion is the 4×4 identity matrix. That leaves us with the task of establishing the inverse. Since

$$\begin{bmatrix} A^T & -A^Tt \\ 0 & 1 \end{bmatrix}\begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix} = I \tag{6.10}$$

we have found an inverse for a rigid motion within the rigid motion group. Thus, the set of rigid motions is a group. It is also known as the *special Euclidean group* and is denoted by *SE*(3).

The 4×4 matrix representation of the rigid motion clearly shows that the rigid motion group is a subgroup of the general linear group *Gl*(4, $\mathbb{R}$). A closer look at Eqs. (6.9) and (6.10) shows that the translation group and the rotation group are subgroups of the rigid motion group; it also shows that the translation group is a normal subgroup. (How?) The rigid motion group has several other subgroups as well, and some of these will be explored in the following examples.

Example 6.13 Consider all combinations of translations along the *z*-axis and rotations about the *z*-axis. It can be represented by the 2-parameter family of matrices

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.11}$$

where ϕ and *d* are two independent parameters for rotation and translation, respectively. It can be seen that these matrices form a group. It is a subgroup of the rigid motion group. It is also a continuous subgroup of the rigid motion group.

Example 6.14 Consider the screw motion about the *z*-axis, which can be represented by the 1-parameter family of matrices

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & \mu\phi \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.12)$$

where $\begin{bmatrix} \cos\phi & -\sin\phi & 0 & d_x \\ \sin\phi & \cos\phi & 0 & d_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ is the independent parameter and μ is a constant called the *pitch* of the screw (advance per unit turn). These matrices form a group, and it is a continuous subgroup of the rigid motion group.

Example 6.15 Consider the following set of rigid motions in the *xy*-plane. It consists of all translations in the plane and rotations about the *z*-axis. It can be represented by the 3-parameter family of matrices

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & d_x \\ \sin\phi & \cos\phi & 0 & d_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.13)$$

where ϕ, d_x, $d_y \in \mathbb{R}$ form the independent parameters. These matrices form a group, and it is a continuous subgroup of the rigid motion group.

6.2 SYMMETRY GROUPS

We are now ready to define symmetry formally. Let the collection of automorphisms of a point-set *S* in Euclidean space, denoted Aut*(S),* be the set of rigid motions that leave *S* invariant. That is, Aut $(S) = \{r \in R : rS = S\}$. [Recall that *R* is the short form for the rigid motion group; that is, *SE*(3)]. Also denote by $\mathrm{Aut}_0(S)$ the connected component of Aut*(S)* that contains the identity element in the rigid motion. The notion of automorphism captures the notion of symmetry and, in fact, is used synonymously in the literature. Our automorphisms are somewhat more restricted than general automorphisms, which allow any transformation *r* in the given definition; we permit only rigid motion and do not allow reflection.

We note that Aut*(S)* forms a group, called the *symmetry group,* because of the following.

1. If $r_1, r_2 \in \mathrm{Aut}(S)$ then $r_1 \bullet r_2(S) = r_1(S) = S$, so $r_1 \bullet r_2 \in$ Aut*(S)*.
2. If $r_1, r_2, r_3 \in \mathrm{Aut}(S)$ Aut*(S),* then $r_1 \bullet (r_2 \bullet r_3) = (r_1 \bullet r_2) \bullet r_3$ because r_1, r_2,

$r_3 \in R$

3. Since *I(S)=S,* we have $I \in \mathrm{Aut}(S)$
4. Since $r \in R$ there is an inverse $r^{-1} \in R$ such that $r^{-1}(S)= r^{-1} \bullet r(S)=I(S)$, so $r^{-1} \in \mathrm{Aut}(S)$

Therefore, Aut*(S)* is a subgroup of the rigid motion group *R*. *This crucial fact establishes the link between symmetry and subgroups of the rigid motion group.* We will now illustrate the idea of symmetry groups using several examples.

Example 6.16 Consider the point-set $S=\{(x, y, z): x^2+y^2=4\}$. It is a cylinder of radius 2 units, and its axis is the *z*-axis. *S* is invariant with respect to translations along the *z*-axis and rotations about the *z*-axis. So Aut*(S)* can be represented by the set of matrices given in Example 6.13. It is a continuous subgroup of the rigid motion.

Example 6.17 Consider the point-set $S = \{(x, y, z): x = 2\cos\phi$ *S*={*(x, y, z)*:*x*=2 cosφ, $y = 2\sin\phi, z = \phi/(2\pi)\}$ It is a space curve—a right-handed helix whose axis is the *z*-axis. Its base cylinder radius is 2 units, and it has a pitch of 1 unit (advance per revolution). The screw motion of Example 6.14 keeps the point-set *S* invariant. Specifically, the matrices in Eq. (6.12) with $\mu=1/(2\pi)$ form Aut*(S)*. It is a continuous subgroup of rigid motion.

Example 6.18 Consider the solid pyramid *S* shown in Figure 6.2. It has a square base with vertices at coordinates (1, 1, 0), (−1, 1, 0),

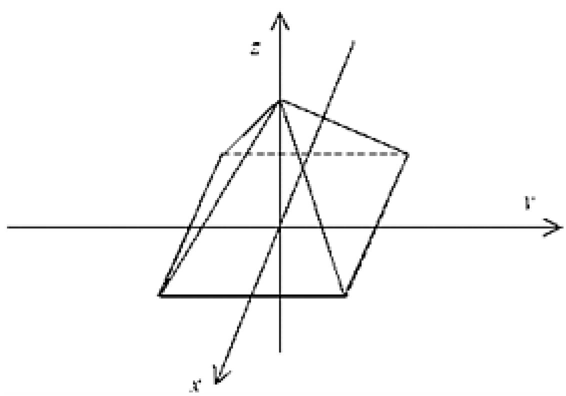


FIGURE 6.2 A pyramid with square base. It has discrete rotational symmetry about the *z*-axis.

(− 1, −1, 0), and (1, −1, 0). The apex of the pyramid is at the vertex (0, 0, 1). The pyramid has a fourfold rotational symmetry about the *z*-axis. Aut*(S)* is given by the rotational motion and associated matrices in Example 6.5. It is a discrete subgroup (C_4) of the rigid motion.

Example 6.19 Consider a square prism represented by the point-set $S=\{(x, y, z): \max(|x|, |y|)=1\}$ and illustrated in Figure 6.3. It is unbounded in the *z*-

direction. It has a fourfold rotational symmetry about the z-axis, and it is also invariant under half-turn (180°) rotations about the x-axis, the y-axis, and the lines in the xy-plane labeled l_1 and l_3 in Figure 6.1(a). In addition it has a translational symmetry along the z-axis. So the set of rigid motions that leave the point-set S invariant can be represented by the set of 1-parameter family of matrices

$$\left\{\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&d\\0&0&0&1\end{bmatrix},\begin{bmatrix}0&-1&0&0\\1&0&0&0\\0&0&1&d\\0&0&0&1\end{bmatrix},\right.$$
$$\begin{bmatrix}-1&0&0&0\\0&-1&0&0\\0&0&1&d\\0&0&0&1\end{bmatrix},\begin{bmatrix}0&1&0&0\\-1&0&0&0\\0&0&1&d\\0&0&0&1\end{bmatrix},$$

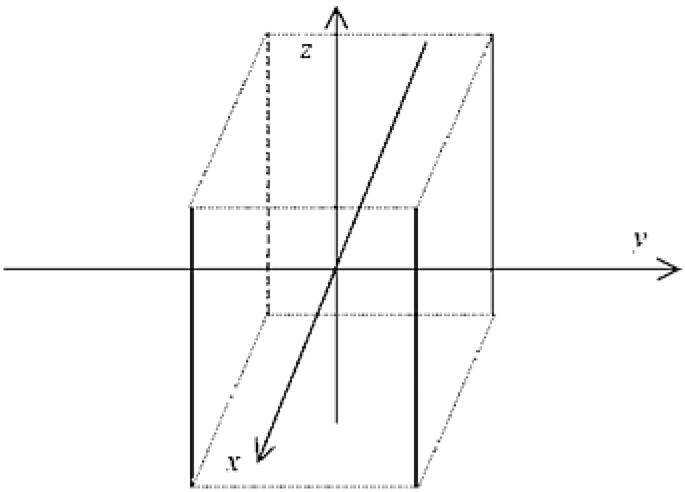


FIGURE 6.3 A square prism that extends all the way up and down the z-axis.

$$\left.\begin{bmatrix}1&0&0&0\\0&-1&0&0\\0&0&-1&d\\0&0&0&1\end{bmatrix},\begin{bmatrix}0&1&0&0\\1&0&0&0\\0&0&-1&d\\0&0&0&1\end{bmatrix},\begin{bmatrix}-1&0&0&0\\0&1&0&0\\0&0&-1&d\\0&0&0&1\end{bmatrix},\begin{bmatrix}0&-1&0&0\\-1&0&0&0\\0&0&-1&d\\0&0&0&1\end{bmatrix}\right\} \tag{6.14}$$

where $d \in \mathbb{R}$ This set forms a group and is the Aut*(S)*. It is a continuous subgroup of rigid motion having eight distinct components, as seen in the matrices of Eq. (6.14). Only the first component, that is,

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&d\\0&0&0&1\end{bmatrix} \tag{6.15}$$

contains the identity matrix and hence is denoted by $\mathrm{Aut}_0(S)$. Among Aut*(S)* it is called the *connected component of the identity* and is a group by itself. Arbitrarily small rigid motions that keep the prism of Figure 6.3 invariant are to be found only in $\mathrm{Aut}_0(S)$, that is, in the matrices of Eq. (6.15).

Example 6.20 Consider the point-set $S = \{(x, y, z) \colon z \in \mathbb{Z}\}$ It consists of all planes parallel to the *xy*-plane that intersect the *z*-axis at integral values (positive and negative, including zero). This point-set remains invariant under a set of rigid motions represented by the matrices

$$\begin{bmatrix}\cos\phi & -\sin\phi & 0 & d_x\\ \sin\phi & \cos\phi & 0 & d_y\\ 0&0&1&d_z\\ 0&0&0&1\end{bmatrix} \tag{6.16}$$

where ϕ, d_x, $d_y \in \mathbb{R}$ and $d_z \in \mathbb{Z}$ These matrices form a group, and this group is Aut*(S)*. It is a continuous subgroup of the rigid motion group. It contains many connected components, and only that component for which d_z=0, that is, the set of rigid motions represented by the matrices

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & d_x \\ \sin\phi & \cos\phi & 0 & d_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.17}$$

contains the identity element. So the connected component of the identity, $\mathrm{Aut}_0(S)$, is represented by the matrices of Eq. (6.17).

The preceding examples demonstrate the close relationship between geometric objects with symmetry and subgroups of the rigid motion group. Therefore it is natural to expect that any classification of symmetry in geometric objects will depend on the classification of the rigid motion group. In fact it has been found that the best way to classify geometric objects on the basis of symmetry is first to seek a classification of the rigid motion group. Perhaps the most famous attempt in this direction is the classification of the finite subgroups of the rotation group $SO(3)$. It has been shown that there are only the following finite subgroups of $SO(3)$.

1. The cyclic group C_n, illustrated in Example 6.5. Right pyramids with regular polygons as their base exhibit this type of symmetry, as shown in Example 6.18. (Recall that a regular polygon is one in which all sides are equal and all angles are also equal.)
2. The dihedral group D_n, illustrated in Example 6.6. Right prisms constructed by translational sweeps of regular polygons exhibit this type of symmetry. Example 6.19 provides an unbounded version of such an object. In fact, right prisms with finite height will just do as examples of objects with dihedral symmetry.
3. The group of rotations of a regular solid (also known as a *platonic* solid). There are three groups in this category.

   (a) The tetrahedral group has 12 elements. A regular tetrahedron exhibits this symmetry and hence the name. See Figure 6.4(a).
   (b) The octahedral group has 24 elements. A cube and a regular octahedron (a polyhedron having eight identical triangular faces) exhibit this type of symmetry. See Figure 6.4(b).
   (c) The icosahedral group has 60 elements. A regular icosahedron (a polyhedron having 20 identical triangular faces) and a regular pentagon-dodecahedron (a polyhedron with 12 identical pentagonal faces) exhibit this type of symmetry. See Figure 6.4(c).

The compact classification of regular solids into just the five (tetrahedron, cube, octahedron, pentagon-dodecahedron, and icosahedron) is considered one of the major accomplishments of the ancient Greek geometers. In fact, the last theorems of Euclid's *Elements* are devoted to these, and Book XIII concludes with the remark "I say next that no other figure, besides the said five figures, can

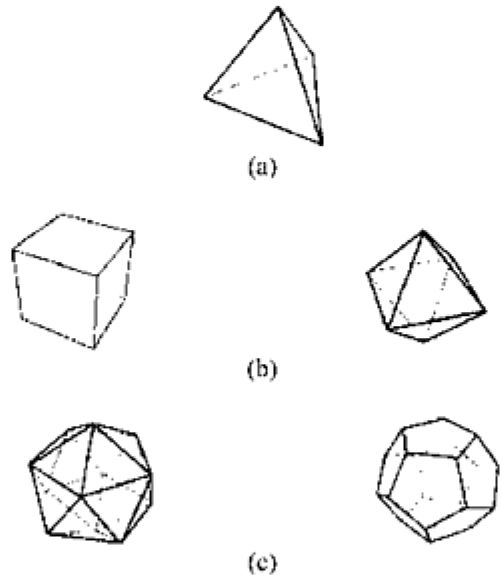


FIGURE 6.4 Five platonic solids.

be constructed which is contained by equilateral and equiangular figures equal to one another."

It turns out that a similar compact classification of geometric objects exists on the basis of their continuous symmetry. For this we depend on the continuous subgroups of the rigid motion group, to which we now turn.

6.3 CONNECTED LIE SUBGROUPS OF THE RIGID MOTION GROUP

First, we need to define the notion of *continuous* groups more precisely. Here is where *Lie* groups, which are groups endowed with manifold structure such that the group operations are C^∞ functions, prove useful. Intuitively, the group operations in a Lie group are smoothly continuous. Examples 6.13, 6.14, and 6.15 illustrate rigid motions that are continuous and also form groups. These are Lie groups. In contrast, Examples 6.5, 6.6, and 6.12 illustrate subgroups of rigid motions whose elements are not continuous and therefore are not Lie groups, even though they all are subgroups of the rigid motion group. We are mainly interested in Lie subgroups of the rigid motion group in three-dimensional space, which we will address now. A specialization of these results for the rigid motion in two-dimensional space follows afterwards.

6.3.1 Three-Dimensional Case

To start with, the full rigid motion group R is a Lie group. It is the Lie subgroups of R that we seek. Specifically, we ask a sharp mathematical question: What are the connected Lie subgroups of the rigid motion group? The answer to this question is surprisingly simple, although its proof is not. It has been rigorously shown that only the following are the connected Lie subgroups of the rigid motion group. Some of the subgroups are defined as products of other subgroups; in these cases, the corresponding normal subgroup has been identified by appropriate reference.

1. *The group of rigid motions itself.* As we saw earlier it is the product of the translation group *T*(3) and the rotation group *SO*(3). It is the full rigid motion group. It is also called the special Euclidean group and is denoted by *SE*(3). For simplicity, we often denote this group by *R*. It has six independent parameters, three for translation and three for rotation. It can be represented by 4×4 matrices, as in Eq. (2.9). See Example A2.9 in Appendix 2 for the normal subgroup of this group.
2. *The identity alone.* It is the identity element in the rigid motion group. It has only one operation, which can be represented by the 4×4 identity matrix.
3. *The group of rotations about an arbitrary but fixed point.* It is the full rotation group *SO*(3). It has three independent parameters. It can be represented by matrices in Examples 6.7 and 6.8.
4. *The group of translations in space.* It is the full translation group *T*(3), also denoted by $\mathbb{R}^3$. It has three independent parameters.
5. *The group of translations in an arbitrary but fixed plane.* See Example 6.11, which illustrates a case isomorphic to this subgroup. It has two independent parameters.
6. *The group of translations along an arbitrary but fixed line.* See Example 6.10, which illustrates a case isomorphic to this subgroup. It has one independent parameter.
7. *The product of the group of rotations about an arbitrary but fixed line and the group of translations along the same line.* Example 6.13 illustrates a group that is isomorphic to this. This group has two independent parameters. See Example A2.10 in Appendix 2 for normal subgroups of this group.
8. *The product of the group of rotations about an arbitrary but fixed line and the group of translations in space.* We have not seen an example of this subgroup. It is isomorphic to the rigid motion group that can be represented by the 4-parameter family of matrices

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & d_x \\ \sin\phi & \cos\phi & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.18}$$

   where ϕ, d_x, d_y $\in \mathbb{R}$ The rigid motion of Eq. (6.18) is a rotation about the *z*-axis and translations along all the three axes. It can be shown that they form a group, and it is a continuous subgroup of the rigid motion group. It has four independent parameters. See Example A2.9 in Appendix 2 for a normal subgroup of this group.
9. *The screw group of pitch* μ *about an arbitrary but fixed line.* A screw group of pitch μ about a line is the group of rotations about the line that are accompanied by translations along the line by the amount μ per revolution. It is illustrated in Example

6.14. Within a particular screw group, the pitch μ is treated as a constant. Therefore, members of a particular screw group belong to a 1-parameter family.

10. *The product of the screw group of pitch* μ *about an arbitrary but fixed line and the group of translations along a plane orthogonal to the line.* We have not seen an example of this subgroup either. It is isomorphic to the rigid motion group represented by

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & d_x \\ \sin\phi & \cos\phi & 0 & d_y \\ 0 & 0 & 1 & \mu\phi \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.19}$$

where ϕ, d_x, d_y $d_y \in \mathbb{R}$ and μ is the pitch. It can be shown that these motions form a group and it is a continuous subgroup of the rigid motion group. See Example A2.11 in Appendix 2 for a normal subgroup of this group.

If an engineer, as opposed to a mathematical purist, does the counting, it is customary to consider specialization of subgroups 9 and 10 when the screw pitch μ=0. So we can have 12 connected Lie subgroups in all, with the last two being as follows.

11. *The group of rotations about an arbitrary but fixed line.* Example 6.4 illustrates this subgroup. It is also a subgroup of *SO*(3). It has one independent parameter.
12. *The product of the rotation group about an arbitrary but fixed line and the group of translations along a plane orthogonal to the line.* It is the two-dimensional rigid motion. Example 6.15 illustrates this subgroup. It has three independent parameters. See Example A2.11 in Appendix 2 for a normal subgroup of this group.

Before we conclude this section, a couple of observations are in order. First, we note that we asked only for the *connected* Lie subgroups of the rigid motion. These are the connected components of all Lie subgroups of *R* that contain the identity. They correspond to what engineers and physicists loosely call "instantaneous" or "small" motions. Second, these subgroups themselves are not the object of our interest; they are necessary intermediate classifications that lead us directly to a classification based on continuous symmetry.

6.3.2 Two-Dimensional Case

It is easy to obtain a two-dimensional specialization of the results of Section 6.3.1. These are presented here because several applications use two-dimensional objects. The rigid motion group in a plane is the product of *T*(2) and *SO*(2), and it is a Lie group. See Example 6.15 for a matrix representation of this group. It consists of translations in the plane, denoted by *T*(2), and rotation about a fixed point in the plane, denoted by the special orthogonal group *SO*(2). *T*(2) is a normal subgroup, and hence the product of *T*(2) and *SO*(2) is well defined. The only connected Lie subgroups of the two-dimensional rigid motion group are the following.

1. *The group of rigid motions itself in the plane.* It is the product of $T(2)$ and $SO(2)$. It has three independent parameters, two for translation and one for rotation. It can be represented by a 3×3 matrix of the form

$$\begin{bmatrix} \cos\phi & -\sin\phi & d_x \\ \sin\phi & \cos\phi & d_y \\ 0 & 0 & 1 \end{bmatrix} \tag{6.20}$$

   where ϕ, d_x, d_y $d_y \in \mathbb{R}$ The rigid motion of Eq. (6.20) is a counterclockwize rotation about the origin and translations along all the x- and y-axes.
2. *The identity alone.* It is the identity element in the rigid motion group. It can be represented by the 3×3 identity matrix.
3. *The group of translations in the plane.* It is the translation group $T(2)$, also denoted by $\mathbb{R}^2$. It has two independent parameters.
4. *The group of rotations about an arbitrary but fixed point in the plane.* It is the special orthogonal group $SO(2)$. It has only one parameter.
5. *The group of translations along an arbitrary but fixed line in the plane.* This is a one-dimensional translation along a straight line. It has only one parameter.

We can now turn to a classification based on continuous symmetry.

6.4 CLASSIFICATION OF CONTINUOUS SYMMETRY GROUPS

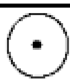
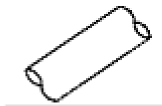
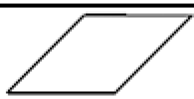
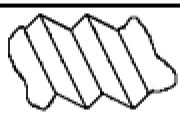
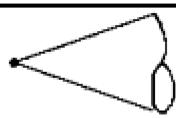
As we remarked earlier, our search for all connected Lie subgroups of the rigid motion group is only a means to an end, which is a compact classification of continuous symmetry of geometric objects. We will achieve this objective in this section. We begin with a discussion of the three-dimensional case; a simple specialization to two-dimensional case will follow afterwards.

6.4.1 Three-Dimensional Case

The fact that there are only 12 connected Lie subgroups of the rigid motion group in three-dimensional space is encouraging because if a point-set S were to possess continuous symmetry, it should come from one of these 12 subgroups. So we turn the problem around and ask for point-sets that remain invariant under the action of each of the 12 subgroups. To do that, we first observe that for the type of point-sets encountered in engineering, the set of continuous rigid motions that keep the point-set invariant are indeed Lie groups. Formally, we say that if the point-set S is what we encounter in geometric modeling, then Aut*(S)* is a Lie group.

So, for all engineering applications we can assume that Aut*(S)* is a Lie subgroup of the rigid motion group. It then follows that $\text{Aut}_0(S)$ is a connected Lie subgroup of the rigid motion group, and, therefore, must belong to one of the 12 classes listed in Section 6.3.1. But some of the listed subgroups cannot preserve the point-set they act on and hence cannot be automorphic. For example, if a subgroup contains $T(3)$, that is, all translations in space, then its action on any proper subset S of Euclidean space can sweep the entire space and, therefore, cannot leave S invariant. This condition applies to subgroups of the rigid motion group listed under classifications 1, 4, and 8 in Section 6.3.1. Classifications 10 and 12 involve translations in a plane perpendicular to a line as well as rotations about this line. It can be shown that these two subgroups also cannot leave the point-set S invariant.

TABLE 6.1 Seven Classes of Symmetry

Class	Surface (example) S	$Aut_0(S)$	dim$(Aut_0(S))$	Reference element or tuple
Spherical		Rot(3)	3	PT
Cylindrical		Product of Tr(1) and Rot(1)	2	SL
Planar		Product of Tr(2) and Rot(1)	3	PL
Helical		Screw group with pitch μ	1	HX
Revolute		Rot(1)	1	(SL, PT)
Prismatic		Tr(1)	1	(PL, SL)
General		[	0	(PL, SL, PT)

PT: point; SL: straight line; PL: plane; HX: helix; $Aut_0(S)$: automorphism of S under small motion; Rot(n): n independent rotations; Tr(n): n independent translations; I: identity motion; dim $(Aut_0(S))$: dimension of the automorphism group.

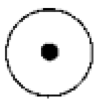
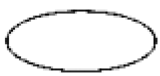
Thus a total of five out of the 12 subgroups of the rigid motion group cannot leave a point-set S on which they act invariant. The remaining seven can, and they form the seven classes of symmetry shown in Table 6.1. The first column lists the names of the seven classes. Example sets in the form of surfaces are shown in the second column. These are merely examples; the point-set *S* need not be connected and need not even be a surface. The third column lists the subsets of the rigid motion that form the automorphism groups for *S,* and the fourth column shows the dimension of this group, that is, how many independent parameters are involved in that subgroup of the rigid motion group. The last column provides for each of the seven classes simple geometric elements, sometimes combined in the form of a tuple, that belong to the same automorphism group as any set in that class. These simple elements and tuples consist of just points, straight lines, planes, and helices. Since the symmetry classification of Table 6.1 is the basis for a general theory of relative positioning in Chapter 7, each row in the table deserves a closer look.

1. A sphere remains invariant under three rotations about its center and thus belongs to the spherical class. The dimensionality of $\mathrm{Aut}_0(S)$ is 3 for this class. The center of the sphere, which is a point, also remains invariant under rotational motions about it. Therefore the center is the reference element for this class.
2. A cylinder is invariant under translational motion along its axis and rotational motion about the axis; it belongs to the cylindrical class. The dimensionality of $\mathrm{Aut}_0(S)$ is 2 for this class. The axis of the cylinder, which is a straight line, does not change under a translational motion along it and rotational motion about it. This axis is then the reference element for this class.
3. A plane, for example, is invariant under translational motion parallel to the plane and rotational motion about an axis perpendicular to the plane; hence, it belongs to the planar class. Note that a pair of two parallel planes, or any number of parallel planes, also belongs to this class. The dimensionality of $\mathrm{Aut}_0(S)$ is 3 for this class. For a geometric object in the planar class, we can always find a plane that remains invariant under the automorphic motion for that object; in fact, any plane parallel to the chosen plane will also satisfy this condition. This plane is the reference element for this class.
4. A helical surface remains invariant if, while being rotated about its axis, it is advanced along the axis by a distance determined by its pitch; it belongs to the helical class. The dimensionality of $\mathrm{Aut}_0(S)$ is 1 for this class. For a geometric object in the helical class, any helix with the same axis, the same right- or left-handedness (that is, the same chirality), and the same pitch will do as the reference element.
5. A cone, for example, remains invariant under rotational motion about its axis and hence belongs to the revolute class. The dimensionality of $\mathrm{Aut}_0(S)$ is 1 for this class. For an object in the revolute class, if we choose a tuple consisting of the axis of revolution and a point on the axis of revolution, then this tuple will remain invariant under the automorphic motion for that object. (More about automorphism of tuples is discussed in the next section.) It is the reference tuple for this class.
6. An elliptic cylinder, for example, remains invariant under translational motion along its ruling and thus belongs to the prismatic class. The dimensionality of $\mathrm{Aut}_0(S)$ is 1 for this class. For an object belonging to the prismatic class, if we choose a tuple consisting of a straight line parallel to the axis of the prism and a plane containing the straight line, then this tuple will remain invariant under the automorphic motion for that object. This is the reference tuple for this class.

7. A hyperbolic paraboloid, for example, cannot remain invariant under any small rigid motion—its automorphism contains only the identity element and it belongs to the general class. The dimensionality of $\text{Aut}_0(S)$ is 0 for this class. For a simple object belonging to the general class we can find a 3-tuple of a point, a straight line containing the point, and a plane containing the straight line, which remain invariant only under the identity motion. This will serve as the reference tuple for this class.

It is instructive to compare these seven classes with Svensen's six elementary cases of size dimensioning described in Section 1.1. He got the spherical, cylindrical, revolute (specialized as a cone), and prismatic classes right. He also covered the general class under "other solids." He missed the planar and helical classes, but inserted "pyramids" as a separate case. Nevertheless, it is quite impressive that his empirical classification of elementary objects based on considerable practical experience comes very close to the rigorous theoretical classification based on symmetry.

TABLE 6.2 Three Classes of Symmetry for Planar Objects

Class	Curve (example) S	$Aut_0(S)$	dim$(Aut_0(S))$	Reference element or tuple
Circular		Rot(1)	1	PT
Linear		Tr(1)	1	SL
General		1	0	(SL, PT)

Nomenclature the same as in Table 6.1.

6.4.2 Two-Dimensional Case

A specialization of Table 6.1 for the two-dimensional case, that is, for planar objects, is shown in Table 6.2. As would be expected, considerable simplification is achieved in the two-dimensional case. Of the five connected Lie subgroups listed in Section 6.3.2, only three can have geometric objects in a plane that remain invariant under their action. Let's examine them briefly.

1. *Circular class.* A circle remains invariant under rotation about a fixed point in the plane. The center of the circle also remains invariant under this rotational motion. The center is the reference element for this class.
2. *Linear class.* A line remains invariant under translation along the line. It is also its own reference element.
3. *General class.* An ellipse, with unequal major and minor axes, cannot remain

invariant under any small rigid motion in the plane. A tuple of a line and a point on the line serves as a reference element for objects in this class.

6.5 CLASSIFICATION OF TUPLES OF SETS

Often we are interested in a group of rigid motions that leave more than one point-set invariant. This leads us to a consideration of a "collection" or "combination" of point-sets. This notion can be captured mathematically by the concept of a tuple of point-sets introduced earlier. Some examples of tuples that remain invariant under rigid motion were provided when reference tuples were discussed earlier.

Recall that tuples are ordered, rigid collection of point-sets. We can then define a consistent automorphism for a tuple of point-sets as $\text{Aut}(S_1, S_2, \ldots, S_n) = \{r \in R : rS_i = S_i \text{ for all } i\}$. A couple of important properties of the automorphism of tuples follow.

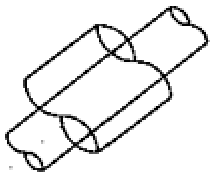
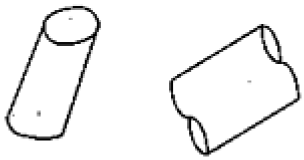
1. From the definition of automorphism for a tuple of point-sets, it follows that $\text{Aut}(S_1, S_2, \ldots, S_n) = \bigcap_{i=1}^{n} \text{Aut}(S_i)$. So by knowing the automorphisms of individual members of a tuple, we can quickly infer the automorphism of the tuple by just collecting the common elements in the automorphisms.
2. Following arguments similar to the automorphism of a single set, we can show that Aut$(S_1, S2, \ldots, S_n)$ is a subgroup of the rigid motion group R. For this, all we need to do is to establish that Aut$(S_1, S_2, \ldots, S_n)$ is a group.

   But is it a Lie subgroup? Not necessarily, even if Aut(Si) is a Lie subgroup for all i. However, it can be shown that if Aut(S_i) is a Lie subgroup for all i, then $\text{Aut}_0(S_1, S_2, \ldots, S_n)$ is a Lie subgroup of R and, therefore, belongs to one of the seven symmetry classes of Table 6.1. In addition, $\text{Aut}_0(S_1, S2, \ldots, S_n)$=the identity component of $\bigcap_{i=1}^{n} \text{Aut}_0(Si)$.

Thus, the symmetry-based classification is closed, in the sense that if we take any two point-sets whose classification according to Table 6.1 is already known, then the symmetry classification of the 2-tuple of these two point-sets also belongs to one of the seven from Table 6.1.

Example 6.21 Table 6.3 shows an example of classifying a 2-tuple (S_1, S_2) of two cylindrical surfaces. Both S_1 and S_2 belong to the cylindrical class, and each has an automorphism group that is a product of a group of translations along its axis and a group of rotations about this axis. Classification of the tuple depends on the relative orientation and location of the cylinder axes.

TABLE 6.3 Classification of a 2-Tuple $(S_{1,} S_2)$ of Two Cylindrical Surfaces

Case	Illustration	Class
parallel axes		prismatic
coincident axes		cylindrical
skew axes		general

1. If the axes are parallel but distinct, the tuple belongs to the prismatic class because the intersection $\text{Aut}_0(S_1) \cap \text{Aut}_0(S_2)$ of the two cylindrical automorphism groups results in just a translation along the common axial direction.
2. If the axes are coincident, then the two automorphisms are identical and their intersection yields an automorphism group that belongs to the cylindrical class.
3. If the axes are skew, then the only intersection of the two automorphisms is the identity, which places the tuple in the general class.

Thus a simple case analysis is sufficient to infer the 2-tuple classification.

Classification results for tuples, such as those just presented, are sometimes referred to as *reclassification* to emphasize the fact that the symmetry classification is closed for the tuples. Table 6.4 enumerates the reclassification of 2-tuples, that is (S_1, S_2), from a knowledge of the classification of S_1 and S_2. It has seven rows and seven columns, and each of the 49 cells in the table is subjected to a case analysis similar to that shown in Table 6.3. Since the classification of (S_1, S_2) is the same as the classification (S_2, S_1), because of the commutativity of set intersection $\text{Aut}_0(S_1) \cap \text{Aut}_0(S_2) = \text{Aut}_0(S_2) \cap \text{Aut}_0(S_1)$, the reclassification table is symmetric.

In the case analyses of Example 6.21 and in Table 6.4, we have tacitly assumed that the relative positioning of S_1 and S_2 can be accomplished by relative positioning their respective reference elements or tuples. In Chapter 7 we will show that it is indeed possible, and this will provide a justification for our case analyses.

The classification of any n-tuple can be derived from the 2-tuple reclassification table by simple iteration. We can do this because, as we saw earlier, $\text{Aut}_0(S_1, S2, \ldots, S_n)$=the identity component of $\bigcap_{i=1}^{n}$ $\mathrm{Aut}_0(S_i)$, and the intersection can be computed incrementally with two groups at a time.

6.6 CLASSIFICATION OF LOWER-ORDER KINEMATIC PAIRS

Geometrical objects can be classified on the basis of their symmetry, per Table 6.1. But do these classifications have anything to do with the functionality or manufacture of these objects? In this section we will provide an affirmative answer to this question.

A kinematic joint involves two parts and permits relative motion between the parts by maintaining contact between their boundary elements. A ball-and-socket joint is a typical example of a kinematic joint. If the mutual contact between the part boundaries during the relative motion occurs over a surface (as opposed to over a line or a point), then the kinematic pair is called a *lower pair*. Mechanisms containing lower-pair kinematic joints are widely used in practice, especially in machine tools and robots. Engineers have long speculated that the only possible lower pairs are the following.

1. *Spherical pair.* It allows rotation about a point.
2. *Cylindrical pair.* It allows rotation about the cylinder axis and a translation along the axis.
3. *Planar pair.* It allows translation along the plane and rotation about an axis perpendicular to the plane.
4. *Screw pair.* It allows a helical movement in terms of a fixed axial advance per revolution.
5. *Revolute pair.* Both contact surfaces are surfaces of revolution, generated by any profile. It allows rotation about its axis.
6. *Prismatic pair.* Both contact surfaces are generalized cylinders other than the right circular cylinder. It allows a translation parallel to a ruling.

This classification is strikingly identical to the first six of the seven classes in Table 6.1. It shouldn't be surprising, because lower-pair motions require that the contact surfaces remain invariant under relative motion. Proof for the classification of Table 6.1 also serves as the proof for the lower-pair classification.

We can now see some practical benefits of classifying surfaces per Table 6.1. First consider their manufacture. These surfaces are commonly manufactured by machine tools that employ guided cutting and forming tools through a chain of lower-pair kinematic joints. Drilling, turning, planing, milling, and broaching provide some examples of manufacturing operations that produce the so-called "manufacturing features" that are subsets of surfaces that belong to the cylindrical, revolute, planar, helical, or prismatic classes.

Next, consider functionality. Parts in a mechanism have surfaces to enable kinematic pairing that fall under the lower-pair classification; such surfaces are important for designing mechanisms. Even if parts are assembled to form static systems, these assemblies have "mating features," such as cylindrical pins and cylindrical holes, which belong to the lower pairs that again fall under these classes. Thus a classification based on symmetry provides a

TABLE 6.4 Complete Symmetry Group Classification of a 2-Tuple (S_1, S_2)[a]

	Spherical p_2	Cylindrical l_2	Planar P_2	Helical h_2	Revolute (l_2, p_2)	Prismatic (P_2, l_2)	General (P_2, l_2, p_2)
Spherical p_1	If p_1=p_2, then spherical; otherwise revolute	If p_1 is on l_2, then re volute; otherwise general	Revolute	General	If p_1 is on l_2, then revolute; otherwise general	General	General
Cylindrical l_1		Case l_1=l_2, then cylindrical; Case l_1//l_2, then prismatic; otherwise general	Case l_1//P_2, then prismatic; Case $l_1 \perp P_2$ then revolute; otherwise general	If axes coincide, then helical; otherwise general	If l_1=l_2, then revolute; otherwise general	If l_1//l_2, then prismatic; otherwise general	General
Planar P_1			If P_1//P_2, then planar; otherwise prismatic	General	If $l_2 \perp P_1$ then revolute; otherwise general	If l_2//P_1, then prismatic; otherwise general	General
Helical h_1				If h_1C*h_2, then helical; otherwise general	General	General	General
Revolute (l_1, p_1)					If l_1=l_2, then revolute; otherwise general	General	General
Prismatic (P_1, l_1)						If l_1//l_2, then prismatic; otherwise general	General
General (P_1, l_1, p_1)							General

[a]S_1 is a member from a class in the first column and S_2 is a member from a class in the first row, based on case analyses of constraints on reference elements and tuples. Parallelism constraint is denoted by // and perpendicularity constraint is denoted by $\perp$. In the (helix, helix) case, C* stands for a special constraint where the axes of the helices are coincident and the helices have the same chirality and pitch. The lower triangular cells are not filled because the table is symmetric.

convenient grouping of these features in a static assembly or in an assembled mechanism that executes a motion.

6.7 SUMMARY

The most important results of this chapter are summarized in the symmetry classification of Table 6.1 and the tuple classification of Table 6.4. Quadrics supply examples for six of the seven classes in Table 6.1; the helical class is the exception. The real power of the continuous symmetry classification lies with the reclassification shown in Table 6.4, which establishes the fact that any tuple of geometric objects belongs to one and only one of the seven continuous symmetry classes. This, along with the reference elements and tuples listed in Table 6.1, forms the basis for a general theory of relative positioning described in the next chapter.

6.8 EXERCISES

1. Apply the definition of groups to verify that the matrices defined by Eq. (6.6) form a group. Repeat the same for the matrices defined by Eq. (6.7).
2. Identify the continuous symmetry classification of each of the 12 real quadrics in Table 4.1. Include special cases as well.
3. Identify the discrete symmetry classification of the prisms and the pyramids (both positive and negative) shown in Figure 1.1.
4. Identify the continuous symmetry classification of each of the elementary cases (both positive and negative) shown in Figure 1.1.
5. Give detailed proofs of Exercises A2.9, A2.10, A2.11, and A2.13 in Appendix 2.
6. From first principles, prove that Aut$(S_1, S_2, \dots, S_n)$ is a subgroup of the rigid motion group.
7. Classify the continuous symmetry of a 2-tuple consisting of a plane and a cylinder. Record your results in a table similar to Table 6.3.
8. Consider a 3-tuple consisting of a plane, a cylinder, and a cone. What is the classification of the continuous symmetry of this 3-tuple? Consider all possible outcomes.
9. Prepare a reclassification table (a la Table 6.4) for planar objects. This is the two-dimensional specialization of Table 6.4. (*Hint:* Start with Table 6.2.)
10. Research and list some of the higher-order kinematic pairs. How are they applied in engineering?
11. Study the manufacturing processes in turning, drilling, shaping, planing, milling, and broaching, and list the type of “manufacturing features” produced by each of these machining operations. Classify these features according to their continuous symmetry.

6.9 NOTES AND REFERENCES

Weyl (1952) gives a nice account of symmetry for nonspecialists. Denavit and Hartenberg (1955) provided an empirical classification of lower-order kinematic pairs that mirrored the symmetry classification of Table 6.1. The French pioneered the classification of continuous symmetry using Lie groups, for application in mechanical engineering. Clement, Riviere, and Temmerman (1994) give the results for continuous symmetry classification and reclassification, which we have adopted. They use the acronym TTRS (topologically and technologically related surfaces) to refer to the symmetry-based classification of surfaces. A mathematical proof of correctness of these classifications can be found in O'Connor, Srinivasan, and Jones (1996). The continuous symmetry classification has also been standardized recently by ISO in their document ISO/TS 17450–1 (2003).

Whether Aut(S) is a Lie group for any arbitrary point-set *S* is an open problem. But for most cases, we can show that Aut*(S)* is a Lie group. Specifically, the following has been shown by O'Connor, Srinivasan, and Jones (1996).

If *S* is closed, then Aut*(S)* is a Lie group.
If cl*(S)**S* is closed, then Aut*(S)* is a Lie group.

Here, cl stands for set closure and \ denotes set difference. The last condition is very generous; it states that if *S* has a limit set that is closed, then Aut*(S)* is a Lie group. All embedded submanifolds of Euclidean space satisfy this condition. So do algebraic and analytic surfaces and even semialgebraic and semianalytic varieties. In fact, the sets and their boundary elements used in geometric modeling satisfy this condition. This should do for our purpose.

7
General Theory of Dimensioning Relative Positions

It was observed in earlier chapters that positioning an arbitrary three-dimensional rigid body in a stationary reference frame (for example, a global coordinate system) requires six independent parameters—three for location and three for orientation—that are loosely called the degrees of freedom (DOF). Now consider positioning a sphere in a stationary reference frame. It appears that we can accomplish this with considerably fewer parameters. We only require three DOF, namely, for the location of the center of the sphere in the stationary frame, because the symmetry of the sphere, as we saw in Chapter 6, renders the three rotations about its center irrelevant for the positioning task. In fact, we can generalize this type of reasoning for any geometric object, whether a point-set or a tuple of point-sets, provided we know its symmetry group classification, per Table 6.1. The fourth column in that table tells us the dimension of its automorphism group. When this number is subtracted from 6, which is the maximum number of DOF for any rigid object in a stationary reference frame, we obtain the number of DOF for this object. Thus, a cylinder has 6−2=4 DOF, a cone has 6−1=5 DOF, and so on.

Let's take the example further and examine the *relative* positioning of a sphere S and a plane P. Notice that we no longer need to refer to the stationary reference frame explicitly because the relationship between the plane and the sphere doesn't seem to care where the stationary frame is. Put another way, the relationship between the plane and the sphere is unaltered as long as we move both the plane and the sphere as a whole (that is, as a single rigid body) within the stationary reference frame. Further thought shows that the relative positioning of S and P depends only on the distance between the center of S and P, thereby requiring only one relative degree of freedom (RDOF). This additional reduction for the sphere, from three DOF in a stationary reference frame to one RDOF with respect to a plane, seems to arise from some symmetry inherent in the plane P itself, for moving the plane P in any direction parallel to itself leaves P unchanged.

This informal exercise hints at two basic facts that will be formalized shortly First, the relative positioning of two geometric objects S_1 and S_2 does not depend on how the rigid collection (S_1, S_2) is positioned in a stationary reference frame. Second, the number of RDOF of S_1 and S_2 is reduced from the maximum number, 6, due to symmetries in S_1 and S_2. We also note that the relative positioning problem is order independent—that is, positioning S_1 relative to S_2 is the same as positioning S_2 relative to S_1.

We can develop our intuition further by examining several more examples: positioning a cylinder relative to a plane, positioning a sphere relative to a cone, and so on. As we examine more such cases, a third subtle, but important, fact emerges: In all these cases the relative positioning problem seems to reduce to that of relative positioning points, lines, and planes. (Recall that in relative positioning a sphere and a plane, we only had to position a point—namely, the center of the sphere—relative to the plane.) We will see that this fact generalizes to any two arbitrary geometric objects S_1 and S_2, and this provides a powerful mathematical formalism for using just points, lines, planes, and helices for dimensioning relative positions of geometric objects. Thus, this chapter is devoted to a general theory of relative positioning.

7.1 TUPLE CONGRUENCE

We will now attempt a mathematical formalism for the three major concepts encountered earlier and progressively build a mathematical formalism for relative positioning. The main question we will pose and answer is *whether the relative positioning of two geometric objects has changed when each of them is subjected to arbitrarily different rigid motions*. We call this the *tuple congruence question*.

Consider an ordered rigid collection of *n* geometric objects denoted as a tuple $(S_1, S_2, ..., S_n)$. Recall that applying a rigid motion to a tuple of geometric objects formalizes the notion of moving these objects as though they belong to "a single rigid body." That is, applying a rigid motion to a tuple results in a congruent copy of that tuple. Using these definitions we can formally define when a relative positioning has changed and when it has not.

We say that the relative positioning of S_1 and S_2 has not changed if a rigid motion r is applied to the tuple (S_1, S_2). That is, we say that in a 2-tuple (S_1, S_2), the relative positioning of S_1 and S_2 is the same as the relative positioning of rS_1 and rS_2 for any rigid motion r.

More generally, given rigid motions r_1 and r_2, we say that the relative positioning of r_1S_1 and r_2S_2 is the same as the relative positioning of S_1 and S_2 if the 2-tuple (r_1S_1, r_2S_2) is congruent to the 2-tuple (S_1, S_2), that is, if we can find a rigid motion r such that $r(S_1, S_2)=(r_1S_1, r_2S_2)$. Trivially, we can find such a rigid motion if $r_1=r_2$. What is interesting is that we may find one even if $r_1 \neq r_2$. This is due to the fact, described in Chapter 6, that a geometric object S may possess some symmetry so that, for some rigid motion t, $tS=S$ in a point-set theoretic sense. For example, applying any rotation about the center of a sphere reproduces the sphere.

We now get back to the original tuple congruence question of determining whether the relative positioning of two geometric objects has remained the same or has changed under rigid motions applied to them. More formally, our tuple congruence question is whether the 2-tuple (r_1S_1, r_2S_2) is congruent to the 2-tuple (S_1, S_2). This is answered by the following theorem. We will use R to denote the rigid motion group.

Theorem 7.1: 2-Tuple Congruence Theorem *(r_1S_1, r_2S_2) is congruent to (S_1, S_2) if and only if we can find rigid motions $r \in R$ $\alpha_1 \in Aut(S_1)$ and $\alpha_2 \in Aut(S_2)$ such that $r_1=r \bullet \alpha_1$ and $r_2=r \bullet \alpha_2$.*

Proof. First, consider the "if" part. Let there be rigid motions $r \in R$ $\alpha_1 \in Aut(S_1)$ and $\alpha_2 \in Aut(S_2)$ such that $r_1=r \bullet \alpha_2$ and $r_2=r \bullet \alpha_2$. Then, $(r_1S_1, r_2S_2)= (r \bullet \alpha_1S_1, r \bullet \alpha_2S_2)=(rS_1, rS_2)=r(S_1, S_2)$. Therefore, they are congruent.

Next, consider the "only if" part. If the tuples are congruent, we have $(r_1S_1, r_2S_2)=r(S_1,$

$S_2)=(rS_1, rS_2)$ for a rigid motion $r \in R$ Therefore $r_1S_1=rS_1$, which implies that $r^{-1} \bullet r_1S_1=S_1$. Hence $r^{-1} \bullet r_1 \in \mathrm{Aut}(S_1)$ Using similar argument, we have $r^{-1} \bullet r_2 \in \mathrm{Aut}(S_2)$ Letting $\alpha_1 \in Aut(S_1)$ and $\alpha_2 \in Aut(S_2)$ we have $r^{-1} \bullet r_1=\alpha_1$ and $r^{-1} \bullet r_2=\alpha_2$, leading to the desired result that $r_1=r \bullet \alpha_1$ and $r_2=r \bullet \alpha_2$.

A general n-tuple congruence theorem that can be proved similarly goes as follows.

Theorem 7.2: *n*-Tuple Congruence Theorem *$(r_1S_1, r_2S_2,\ldots, r_nS_n)$ is congruent to $(S_1, S_2,\ldots, S_n)$ if and only if we can find rigid motions* $r \in R$ $\alpha_1 \in Aut(S_1)$ $\alpha_2 \in Aut(S_2),\ldots, \alpha_n \in Aut(S_n)$ *such that* $r_1=r \bullet \alpha_1$, $r_2=r \bullet \alpha_2,\ldots, r_n=r \bullet \alpha_n$.

The 2-tuple congruence theorem provides both the necessary and sufficient conditions to determine if the relative position of two geometric objects has changed. These conditions involve memberships in automorphism groups of both S_1 and S_2 introduced in Chapter 6 to describe symmetry. Often, but not always, we will apply this theorem more strictly to cases where the rigid motions r_1 and r_2 are continuous and can be arbitrarily small. This would then restrict the membership of α_1 and α_2 to $\mathrm{Aut}_0(S_1)$ and $\mathrm{Aut}_0(S_2)$, respectively.

A couple of further technical observations on the 2-tuple congruence theorem are in order. First, its necessary and sufficient conditions are in existential form—they don't provide any concrete procedure to find the rigid motions r, α_1, and α_2. Second, note that the conditions refer only to membership in automorphism groups. This is a very important fact because it leads us to the following corollary.

Corollary 7.1: Tuple Replacement Theorem *The answer to the "tuple congruence question" remains unaltered if we replace the point-sets by those in the same symmetry class—in particular, by their reference elements or reference tuples.*

This tuple replacement theorem enables us to replace S_i by ρ_i in the same symmetry class. Under such replacement, the corollary assures us, the "tuple congruence question" can be rephrased as to whether $(r_1\rho_1, r_2\rho_2)$ is congruent to (ρ_1, ρ_2). If ρ_i is chosen to be a reference element or reference tuple form column 5 of Table 6.1 in the same symmetry class as S_i, we have our simplification. We also have a procedural means to answer the tuple congruence question, as the following examples demonstrate.

Example 7.1 Let S_1 and S_2 be two arbitrary spheres, with centers (their reference elements according to Table 6.1) p_1 and p_2, respectively. Now consider the question of whether the 2-tuple (S_1, S_2) is congruent to the 2-tuple (r_1S_1, r_2S_2) for some arbitrary rigid motions r_1 and r_2. The tuple replacement theorem states that we can answer this tuple congruence question by examining whether (p_1, p_2) is congruent to (r_1p_1, r_2p_2). This can be answered quite easily because of Theorem 5.1, which guarantees congruence if and only if $d(p_1, p_2)=d(r_1p_1, r_2p_2)$, that is, if the center distances are equal. Note that we have a procedural method—in this case, comparing some specific distances—to answer the question.

Therefore, the relative position of two arbitrary spheres can be parameterized by the distance between their centers. Their relative position is dimensioned by choosing a numerical value for this center distance, as shown in Figure 7.1.

Example 7.2 Let *S* be a sphere with center *p* and let *C* be an unbounded cylinder with axis *l*. Note that *p* is the reference element of *S* and *l* is the reference element of *C*. Now subject the sphere and the cylinder to some arbitrary rigid motions r_1 and r_2, respectively. Is the 2-tuple (r_1S, r_2C) congruent to the 2-tuple *(S, C)*? Thanks to the tuple replacement theorem, this question is identical to asking whether the 2-tuple (r_1p, r_2l) is congruent to the 2-tuple *(p, l)*. The answer is yes if and only if $d(r_1p, r_2l)=d(p, l)$, according to Theorem 5.2. We see that the congruence theorems of Chapter 5 are providing us with procedural means to answer the tuple congruence question.

So the relative position of a sphere and a cylinder can be parameterized by the distance between the center of the sphere and the axis of the cylinder. It can be dimensioned by assigning a numerical value to this distance. See Figure 7.2.

Example 7.3 It should be easy by now to see how a sphere is positioned relative to a plane. For sphere *S* its center *p* serves as its reference element. For plane *P* the reference element is *P* itself. So for arbitrary rigid motions r_1 and r_2, (r_1S, r_2P) is congruent to *(S, P)* if and only if (r_1p, r_2P) is congruent to *(p, P)*. We can appeal to Theorem 5.3 that says that this is the case if and only if $d(r_1p, r_2P)=d(p, P)$, that is, if the distance between the center

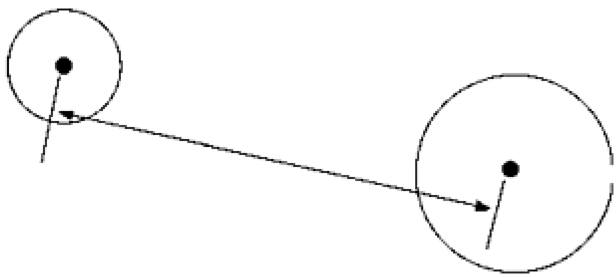


FIGURE 7.1 Dimensioning the relative position of two members of the spherical class.

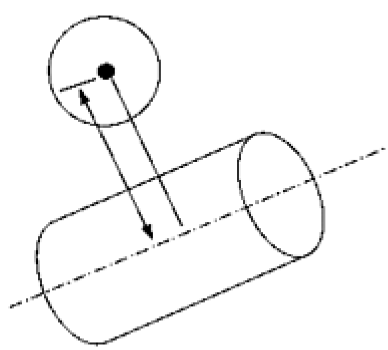


FIGURE 7.2 Dimensioning the relative position of a member of the spherical class and a member of the cylindrical class.

of the sphere and the plane remains the same. In addition to providing a procedural means to check for congruence, we have a parameterization, and hence dimensioning, of the relative position of a sphere and a plane.

Now consider how a sphere can be positioned relative to a planar half-space. The plane can be the reference element for the half-space, but we need to distinguish between the two sides of the plane because one contains the half-space and the other doesn't. So we have the problem of positioning a sphere relative to an oriented plane, as shown in Figure 7.3. The dimension is still the distance between the center of the sphere and the plane, but it is a signed dimension, in the sense that we have to specify whether the center

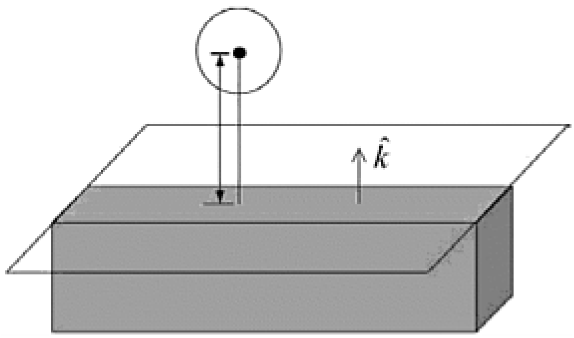


FIGURE 7.3 Dimensioning the relative position of a member of the spherical class and a member of the planar class. The shaded figure is part of a planar half-space. The bounding plane has an outer normal denoted by the unit vector $\hat{k}$.

is located in the positive or negative side indicated by the normal to the plane. In an engineering drawing, the projected view visually encodes the "material side" information, and we don't have to assign a sign to this dimension. But in a CAD system this information is captured by the signed dimension, which is, in fact, a coordinate dimension.

From these examples we begin to see how the relative position of geometric objects that come from spherical, cylindrical, planar, or helical classes can be dimensioned using the congruence theorems and the dimensioning schemes of Chapter 5. If they come from any of the remaining three classes in Table 6.1 we can still dimension their relative positions easily, provided we expand the coverage of Chapter 5 to include reference tuples, which is accomplished in the rest of this chapter. In any case, we have established the fact that the problem of dimensioning relative positions of arbitrary point-sets or tuples of point-sets can be reduced to dimensioning relative positions of simpler entities involving just points, lines, planes, and helices.

It is again instructive to look back on Svensen's scheme for location dimensions described in Section 1.1. He located his "elementary parts" using centers, axes, and reference surfaces. We now know that his approach is also theoretically sound and that he got most of the reference elements right.

7.2 NUMBER OF DIMENSIONS FOR RELATIVE POSITIONS

Before we address the task of relative positioning all reference elements and tuples, we can answer some simple questions about how many dimensions are needed for the task. For this, we first introduce the notion of a *general relative position*.

Definition 7.1 *Two point-sets S_1 and S_2 are in general relative position when $dim(Aut_0(S_1, S_2))$ is a minimum.*

For example, in Table 6.3 the two cylinders are in general relative position when their axes are skew because only in that configuration does $\dim(\mathrm{Aut}_0(S_1, S_2))$ achieve its minimum value of 0. In the other two possible configurations it has a value of 1 when the axes are parallel and distinct, because (S_1, S_2) then belongs to a prismatic class, and a value of 2 when the axes are coincident, because (S_1, S_2) then belongs to the cylindrical class.

When two point-sets S_1 and S_2 are in general relative position, a soft analysis provides the number of dimensions needed to fix this relative position. Let $n_1=\dim(\mathrm{Aut}_0(S_1))$, $n_2=\dim(\mathrm{Aut}_0(S_2))$, and $n_3=\dim(\mathrm{Aut}_0(S_1, S_2))$. Now consider the following sequence of reasoning.

1. Take S_2 first. It will, in general, require six dimensions to fix its position in space because it can have six DOF. But of these six, n_1 will correspond to those rigid motions that leave S_1 invariant. Therefore, only $6-n_1$ dimensions may be needed position S_2 relative to S_1.
2. But of these $6-n_1$ dimensions, n_2 will correspond to those rigid motions that leave S_2 invariant. So we may need only $6-n_1-n_2$ dimensions after all.
3. However, in these calculations we have double counted n_3 because it corresponds to those rigid motions that leave both S_1 and S_2 invariant. So we may actually need $6-n_1-n_2+n_3$ dimensions.

The result of this analysis remains the same if the roles of S_1 and S_2 are reversed. So we have the following theorem.

Theorem 7.3: Relative Degrees of Freedom Theorem *The number of dimensions for relative positioning of S_1 and S_2, when they are in general relative position, is given by $6-\{dim(Aut_0(S_1))+dim(Aut_0(S_2))- dim(Aut_0(S_1, S_2))\}$.*

Let's verify this theorem for the case of two cylinders in Table 6.3. The formula yields a value of 2, which is the number of dimensions required for fixing their relative position when the axes of the cylinders are skew, as illustrated in Figure 7.4. Table 7.1 lists the number of dimensions for all general

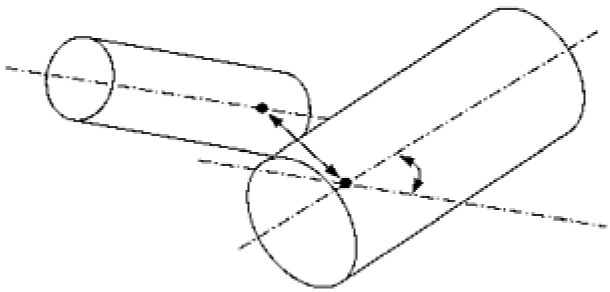


FIGURE 7.4 Dimensioning the relative position of two cylinders whose axes are skew. The twist angle may be given as a positive or negative dimension to indicate chirality.

TABLE 7.1 Number of Dimensions for Relative Positioning of Two Point-Sets When They Are in General Relative Position

	Spherical	Cylindrical	Planar	Helical	Revolute	Prismatic	General
Spherical	1	1	1	2	2	2	3
Cylindrical		2	1	3	3	3	4
Planar			1	2	2	2	3
Helical				4	4	4	5
Revolute					4	4	5
Prismatic						4	5
General							6

This also gives the maximum number of RDOF. Only the upper triangle is filled because the table is symmetric.

relative positions. A comparison with Table 5.1 provides an independent verification for part of the results in Table 7.1.

While our soft analysis has provided some useful information about how many dimensions are needed for relative positioning, it does not tell us what these dimensions are. For this, we need congruence theorems similar to those presented in Chapter 5. As noted earlier, Chapter 5 covered all cases where the geometric objects belong only to spherical, cylindrical, planar, or helical classes. For other cases, we continue by adding one class at a time.

7.3 MORE ON RELATIVE POSITIONING SPHERICAL, CYLINDRICAL, PLANAR, AND HELICAL CLASSES

As Table 6.1 indicates, reference elements for the spherical, cylindrical, planar, and helical classes are the point, the line, the plane, and the helix, respectively. So dimensioning the relative position of two members of these classes reduces to dimensioning the relative position of elementary objects, which we discussed in detail in Chapter 5. There are just a few additional facts about these classes that require our attention.

1. The spherical class is not just one sphere. Any number of concentric spheres, in the form of a tuple, also belong to this class. A spherical ball (all points inside and on a sphere) belongs to this class, and so does its complement (a spherical void). A spherical annulus (points that lie between two concentric spheres) is a member of this class. But in all these cases, the reference element is uniquely defined. It is the center of these spheres.
2. The cylindrical class may contain any number of coaxial cylinders. A solid cylinder, a cylindrical hole, and a hollow cylinder (points that lie between two concentric cylinders) also belong to this class. Again, in all these cases, the reference element is uniquely defined. It is the axis of these cylinders.
3. The planar class may contain any number of parallel planes. A half-space bounded by a plane, an infinite slab (points that lie between two parallel planes), and its complement (an infinite slot) also belong to this class. Mathematically, any plane parallel to these planes can serve as the reference element for this class. This introduces some freedom, as well as ambiguity, in the reference element choice for this class. We will address this issue shortly.
4. The helical class may contain any number of coaxial helices and helicoids. The reference element should have the same axis, chirality, and pitch of this set. The only freedom left is the diameter of the base cylinder of this reference helix. However, in most practical applications (e.g., helical threads), only the axis is used for relative positioning the helical class. This axis is uniquely defined.

So the only case that has some ambiguity in the choice of the reference element is the planar class. Here are some practical tips that engineers use to resolve this ambiguity.

1. If the planar class contains only a planar surface, then that plane itself is chosen as the reference element. Figure 7.5(a) illustrates this case while dimensioning the relative position between a plane and a cylinder.
2. If we have just one half-space bounded by a plane, then the bounding plane is oriented with an outward normal, and this oriented plane is chosen as the reference element. Figure 7.5(b) shows an example of how this can be done.
3. If we have multiple parallel planes, infinite slabs/slots, etc., an attempt is made to find a parallel plane that is also a plane of reflexive symmetry to them. See Figures 7.6(a) and 7.6(b) for some examples. If such a reference plane can be found, it is called the *center plane* and it need not be oriented. Otherwise, any plane parallel to

the

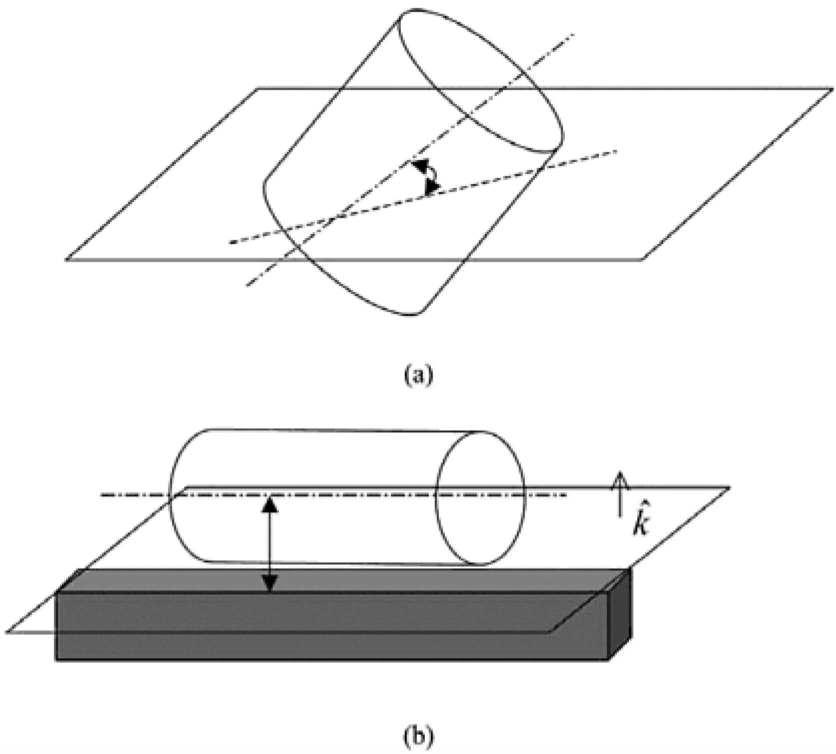


FIGURE 7.5 Dimensioning the relative position between a member of the cylindrical class and a member of the planar class. (a) When the axis of the cylinder is not parallel to the plane, just the angle between the axis and the plane needs to be dimensioned. (b) When the axis of the cylinder is parallel to the plane, the distance between the axis and the plane is dimensioned. In this example it is a signed dimension because the plane may be the reference element for a planar half-space, for example, and may be oriented by the unit outward normal $\hat{k}$.

objects under consideration will do, but we need to orient the plane so that its two sides can be distinguished. Figures 7.6(c) and 7.6(d) illustrate such cases.

7.4 ADDING THE REVOLUTE CLASS

The reference tuple for the revolute class consists of a straight line and a point on the straight line, denoted by (*l, p* on *l*). If *P* is a plane perpendicular to *l* that passes through *p,* then the tuple *(l, P)* can equally serve as the reference tuple for the revolute class.

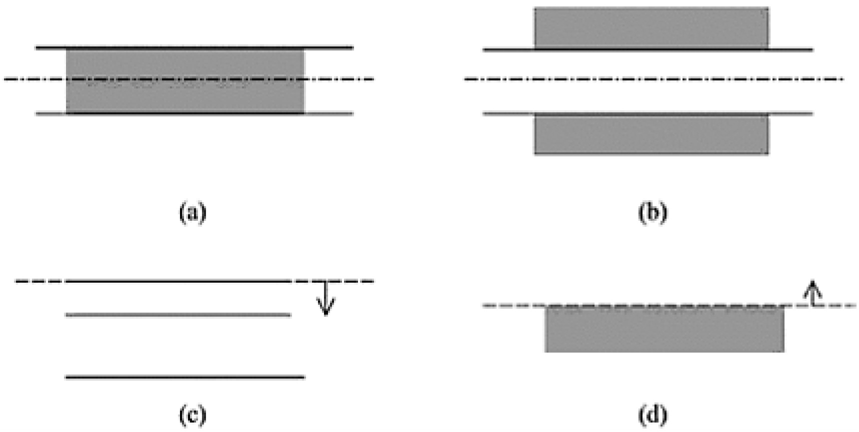


FIGURE 7.6 Choosing the reference element for the planar class. (a) A slab is a three-dimensional region that lies between two parallel planes. It has a plane of reflexive symmetry, called the *center plane,* which has been shown with dashes and dots, and it can serve as the reference element for the slab. (b) A slot is the complement of a slab. Here again, the plane of reflexive symmetry, called the *center plane,* shown with dashes and dots, can serve as the reference element for the slot. (c) A tuple of three parallel planes. Any one of them can serve as the reference plane, but it has to be oriented. (d) A half-space bounded by a plane. The bounding plane can serve as the reference plane, but it has to be oriented.

Since the revolute class contains only objects of revolution, the axis of revolution uniquely defines the reference line *l*. Since any point on *l* can serve as the point *p,* the choice of *p* can be ambiguous. Here are some practical tips to resolve this ambiguity.

1. If the objects in the revolute class have a plane of reflexive symmetry, denoted *P,* that is also perpendicular to the axis of revolution *l,* then *p* is chosen to be the point of intersection of *l* and *P*. Figures 7.7(a) and 7.7(b) show some examples of this case. Then any point *q* on *l* can be positioned relative to *p* by the distance between *p* and *q*.
2. If such a plane cannot be found, then any reasonable choice of *p* on *l* is acceptable. See Figures 7.7(c) and 7.7(d) for examples. Here, the reference tuple is *(p, $\underline{l}$),* where $\underline{l}$ is an oriented line (axis). The orientation gives us the means to properly position other objects relative to the current object in the revolute class. Any point *q* on $\underline{l}$ can be positioned relative to *p* by the signed distance between p and *q,* where the sign is supplied by the orientation direction of $\underline{l}$.

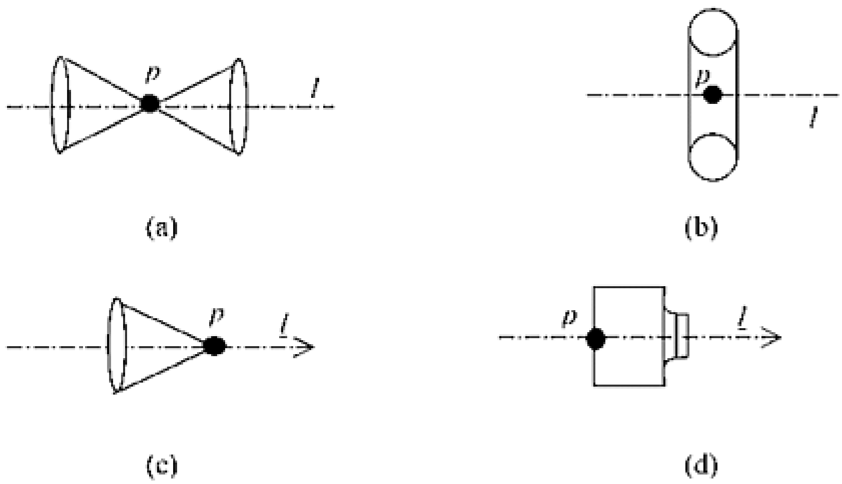


FIGURE 7.7 Choosing the reference point on the axis of revolution *l* for the revolute class. (a) A double cone, which has a plane of reflexive symmetry that is perpendicular to *l*. (b) A torus, which also has a plane of reflexive symmetry that is perpendicular to *l*. (c) Single cone. Here the axis in the reference tuple should be oriented. (d) A stepped shaft of revolution. Here again the axis in the reference tuple should be oriented.

With these preliminaries, we can break down the problem of relative positioning an object S_1 from the revolute class and an object S_2 from the spherical, cylindrical, planar, or helical class into two tasks:

1. Positioning the axis, which may be an oriented or a nonoriented line, relative to the reference elements of S_2. This problem was studied and solved in detail in Chapter 5.
2. Positioning a point q on the axis relative to p and relative to S_2. Positioning q relative to p can be accomplished easily, as discussed earlier. If oriented elements are present in the reference elements or tuples of S_1 or S_2, then this dimension may carry a sign.

Following the foregoing scheme, Figure 7.8 illustrates dimensioning the relative position of a sphere and a cone. Figure 7.9 shows how to dimension the relative position of a planar half-space and a conical half-space.

7.5 ADDING THE PRISMATIC CLASS

The reference tuple for the prismatic class is a plane and a line on the plane, denoted (*P, l* on *P*). The only requirement is that line *l* be along the direction of the translational motion that leaves the object in this class invariant. In fact,

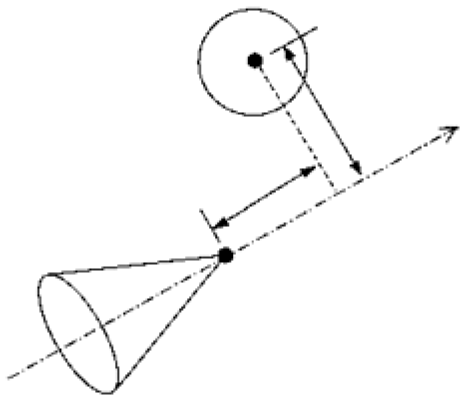


FIGURE 7.8 Dimensioning the relative position of a sphere and a cone. The dimension from the apex of the cone to the projection of the center of the sphere on to the axis of the cone is a signed value.

l and any plane parallel to it can also serve as the reference tuple. This gives us considerable freedom in the choice of *l* and *P,* and it also results in greater ambiguity. In some cases, this ambiguity may be resolved by exploiting reflexive symmetry, as explained next.

If a geometric object (a point-set or a tuple of point-sets) *S* belongs to the prismatic class, then it can also be viewed as a translational sweep of a planar object *C* in a direction perpendicular to the plane *P** that contains *C,*

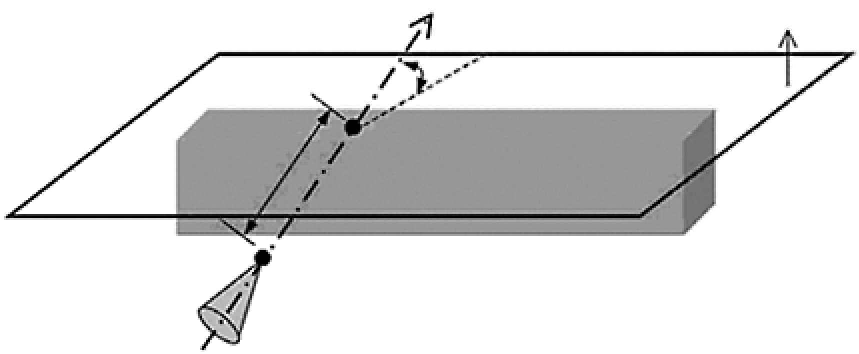


FIGURE 7.9 Dimensioning the relative position of a conical half-space and a planar half-space. It consists of two dimensions: One is the angle between the oriented axis and the oriented plane, and the other is the distance between the apex of the cone and the point of intersection of the axis with the plane. Both dimensions carry signs, and they happen to be positive in this example.

a la generalized cylinders. In fact, *C* can be obtained by intersecting *S* with any plane *P** that is perpendicular to the direction of the translational motion that leaves *S* invariant in the first place. If *C* possesses a line of reflexive symmetry (i.e., an axis) or a point of reflexive symmetry (i.e., a center) in the plane *P*,* then this fact can be exploited in setting up the reference elements for *S*. Figure 7.10 shows several examples of simple planar objects and their reflexive symmetries.

In Figure 7.10(a) we see an isosceles triangle *C* that has only an axis of reflexive symmetry. If this triangle is swept perpendicular to the plane to generate the prismatic object *S,* then the axis is swept to become the plane *P* of reflexive symmetry for *S*. Also, any point on the axis, say, the centroid of the isosceles triangle, is swept as a line *l* that lies in *P* and is along the direction of the translational sweep. The tuple *(P, l)* can then serve as the reference tuple for *S.*

The object *C* in Figure 7.10(b) has a center of symmetry but no axis of reflexive symmetry. We can then choose any convenient line, say, a horizontal line, through this center and sweep the line and the center perpendicular to the plane to obtain the reference plane *P* and the reference line *l,* respectively, we want. The ellipse in Figure 7.10(c) presents an easier problem. It has an axis and a center on the axis. Their sweeps produce the reference plane and reference line we seek. The right triangle in Figure 7.10(d) has neither an axis nor a center. Here we are free to choose any line, say, the one that contains the horizontal edge, and any point on this line, say, the right angle vertex, and sweep them to produce the desired reference plane and the reference line.

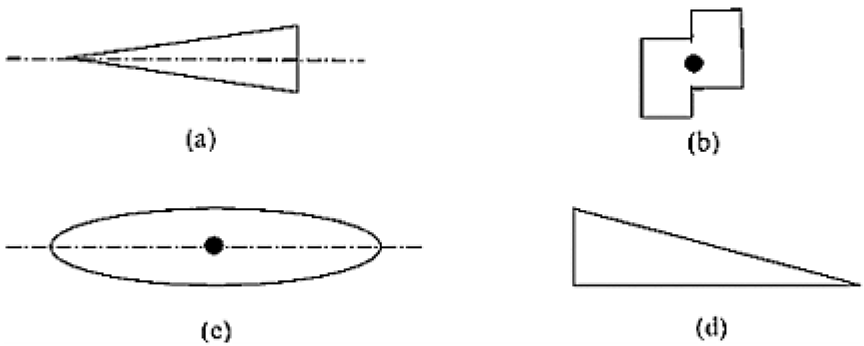


FIGURE 7.10 Planar objects and reflexive symmetry. (a) An isosceles triangle that has an axis of symmetry but no center of symmetry. (b) An object with a center of symmetry but no axis of symmetry. (c) An ellipse that has an axis of symmetry as well as a center of symmetry. In fact, it also has another (minor) axis of symmetry. (d) An object that has neither an axis of symmetry nor a center of symmetry.

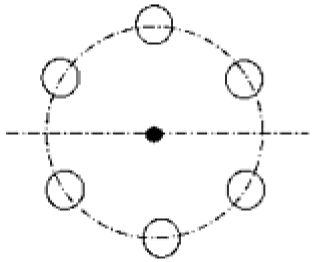


FIGURE 7.11 A symmetric pattern of six holes. When translationally swept perpendicular to the plane that contains them, we get a 6-tuple of parallel cylinders that belongs to the prismatic class.

A common occurrence in engineering is a pattern of parallel cylindrical holes, an example of which is shown in Figure 7.11 as a projected view. In the plane of the figure, these symmetrically placed six circles have a unique center of reflexive symmetry and many axes of reflexive symmetry. One such axis runs horizontally, as shown. When this axis and the center are swept perpendicular to the plane of the figure, they produce the reference plane *P* and the reference line *l* for the 6-tuple of parallel cylinders that belongs to the prismatic class.

A reference tuple (*P, l* on *P*), thus established, may need to be oriented to indicate which is the "material side" of *P* and *l*. The relative positioning problem can then be broken down to positioning the (possibly oriented) plane and the (possibly oriented) line relative to other reference elements and tuples. Figure 7.12 shows how the relative position of a sphere and an unbounded

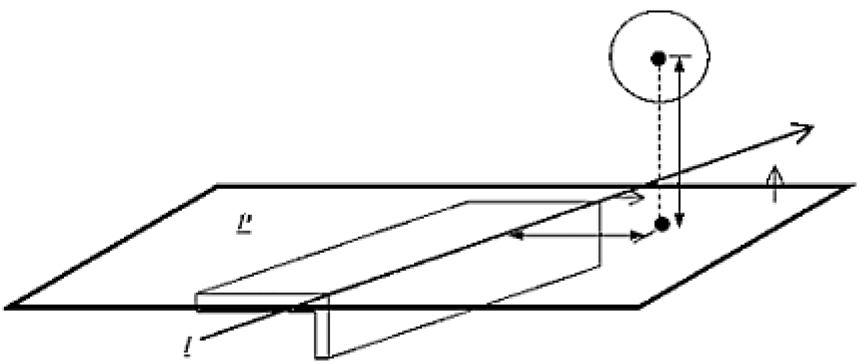


FIGURE 7.12 Dimensioning the relative position of an infinite bracket (prismatic class) and a sphere (spherical class). The reference tuple for the prismatic class is (P, l on P), where P and l are both oriented. The two dimensions shown are signed and can be considered to be coordinate dimensions.

bracket (prismatic class) can be dimensioned. Note that the reference tuple ($\underline{P}$, $\underline{l}$ on $\underline{P}$) for the bracket is oriented. It involves two coordinate dimensions.

7.6 ADDING THE GENERAL CLASS

The reference tuple for the general class is the 3-tuple (*P, l* on *P, p* on *l*). If the reference plane and the reference line are oriented, then the 3-tuple ($\underline{P}$, $\underline{l}$ on $\underline{P}$, *p* on $\underline{l}$) establishes a full three-dimensional reference frame, which can be a Cartesian, cylindrical, or spherical coordinate system. The choice of the reference elements is quire arbitrary, but sometimes the particular object in the general class may have some reflexive symmetry that suggests a preference for these reference elements.

A hyperbolic paraboloid, for example, belongs to the general class. The surface shown in Figure 4.1(f) does suggest some reference elements due to its reflexive symmetry. In this example, there are two planes of reflexive symmetry. They are the *yz*-plane and the *xz*-plane; either of them can serve as the reference plane *P*. In addition, their intersection, the *z*-axis, is an axis of reflexive symmetry and can serve as the reference line *l*. Finally, the origin, which is the sole intersection of *l* with the surface, can serve as the reference point *p*.

Figure 7.13 shows an example of relative positioning a cone (revolute class) and a general object (general class). We can set up a reference tuple ($\underline{P}$, $\underline{l}$, *p*) for the object in the general class, as shown. Table 7.1 indicates that we need five dimensions (or parameters) in this case. We can allocate three of them for positioning the apex of the cone relative to the reference tuple ($\underline{P}$, $\underline{l}$, *p*). The remaining two are for positioning the oriented axis of the cone relative to ($\underline{P}$, $\underline{l}$, *p*).

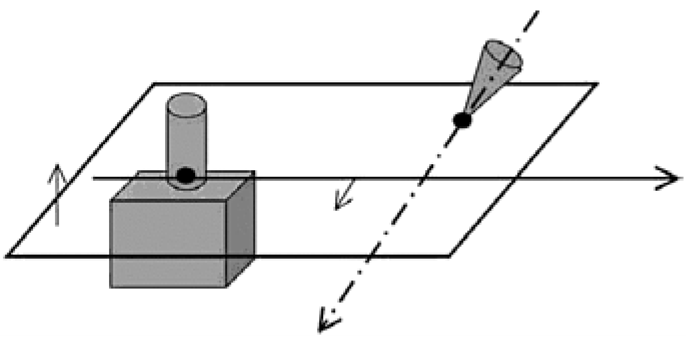


FIGURE 7.13 Relative positioning a member of the revolute class and a member of the general class.

7.7 SUMMARY

The general theory of dimensioning relative positions presented in this chapter is based on the tuple congruence theorem and its corollary, the tuple replacement theorem. The latter theorem reduces the problem of relative positioning geometric objects to that of relative positioning their reference elements or tuples. This means that if we have just a pair of objects, the relative position between them can be dimensioned or parameterized using the results of Chapters 5 and 6 and some additional information provided in this chapter. The reference elements and tuples are also called *datums,* for dimensioning as well as tolerancing purposes.

If we have more than two objects, then their relative position can be dimensioned or parameterized recursively by taking two objects, or tuples of objects, at a time using the results presented thus far. If we want to dimension the relative positions among three or more objects simultaneously, that is, without recursively combining pairwise relative positions explicitly, we enter the domain of dimensional constraints that is discussed in the next chapter.

7.8 EXERCISES

1. Identify reference elements or reference tuples for the elementary objects shown in Figure 1.1.
2. Relate Svensen's location dimensioning scheme illustrated in Figures 1.3 and 1.4 to that based on the general theory of this chapter. Comment in some detail on his location dimension description, presented in Section 1.1, in light of the general theory.
3. Dimension the two parts in Figure 1.2 using the theory of dimensioning developed thus far. Start with dimensioning elementary surfaces. Then position them relative to each other, and build a hierarchy of dimensions of relative positions. Clearly note the incidence, parallelism, and perpendicularity constraints imposed along the way.

7.9 NOTES AND REFERENCES

Clement, Riviere, and Temmerman (1994) provide detailed illustrative examples for relative positioning pairs of objects taken from the seven classes of symmetry. They refer to the reference elements or tuples as *minimum geometric reference elements* (MGRE), emphasizing the simplicity in shape and quantity of the reference elements.

8
Dimensional Constraints

Consider a collection of n geometric objects, denoted by g_1, g_2,..., g_n. These objects can be point-sets or tuples of point-sets. Assume that each gi has been dimensioned completely. Also assume that we are interested in dimensioning their relative positions so that we can create a rigid collection, that is, an n-tuple, of these n objects. In this collection there are $_nC_2$ pairwise relative positions that can be dimensioned. It turns out that graphs, described in some detail in Appendix 3, which should be read along with this chapter, are the best abstractions to capture such geometric relationships. Figure 8.1 illustrates this using a complete graph, where each node is a geometric object and each arc between two nodes stands for the relative position between the objects represented by the two nodes. (Note that each arc can contain from one up to six dimensions.) Chapter 7 showed how to dimension the relative position of any two of these arbitrary geometric objects. But do we need to dimension all of these relative positions to define the rigid collection, that is, an n-tuple, of these n objects? A little reflection indicates that the answer is no, because even for four coplanar points we don't need $_4C_2$=6 dimensions that would fix the relative positions within each pair of the four points—as we know, just 5 dimensions will do to specify a planar quadrilateral.

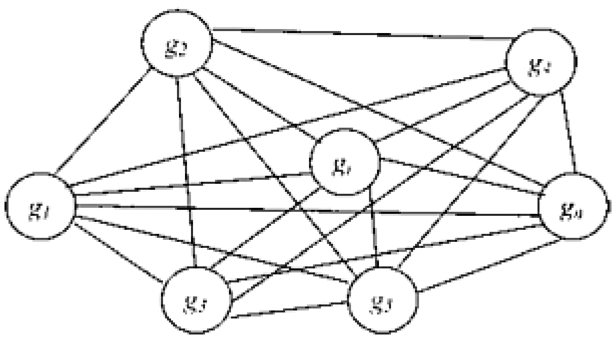


FIGURE 8.1 A complete graph whose nodes are geometric objects and whose arcs are relative positions of the nodes they connect

In general, only $O(n)$ among the $O(n^2)$ dimensions implied by the $_nC_2$ pairwise relative positions may need to be specified to define a rigid collection of n objects. (We use $O(n)$ to denote a function of n whose highest power of n is unity. Similarly, the highest power of n in $O(n^2)$ is 2.) For example, we can start with just two objects and add one object at a time to the rigid collection, thereby requiring only $O(n)$ dimensions in total for relative positioning. This possibility is illustrated as an incremental scheme in Figure 8.2(a), along with another possibility using a balanced binary tree scheme in Figure 8.2(b). Once these *O(n)* dimensions are chosen, they determine the relative position of any pair of the original *n* objects.

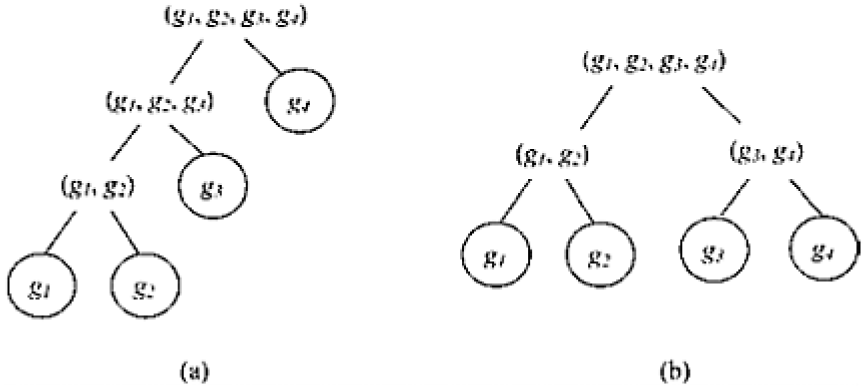


FIGURE 8.2 Relative positioning using hierarchical schemes. The leaf nodes are geometric objects (point-sets or tuples of point-sets). The interior nodes and the root node are tuples. (a) An incremental scheme. (b) A balanced binary tree scheme.

If the relative positions of objects are dimensioned hierarchically using a binary tree scheme, such as those proposed in Figure 8.2 or a combination of them, then we have no further problem to solve. The problem arises only when we have simultaneous specification of certain dimensions of pairwise relative positions among *n* objects. Such simultaneously specified dimensions are called *dimensional constraints*. We then are left wondering whether these dimensions are sufficient and realizable in a rigid object and what the effects of these dimensions would be on the remaining, unspecified pairwise relative positions. These questions arise even in dimensioning the relative positions in a collection of just three distinct points, as the following examples illustrate.

Example 8.1 Based on Theorem 2.2 (side-side-side) for a triangle, we may conclude that a tuple of three points, denoted by (p_1, p_2, p_3), can be dimensioned by specifying the sides of the triangle formed by these points as vertices, as shown in Figure 8.3(a). Each dimension indicated can be viewed as the relative position dimension of two points. Together, they form simultaneous specification of three pairwise relative positions.

But these three dimensions cannot be chosen totally arbitrarily, because triangle inequality imposes the condition, or constraint, that the sum of any two side dimensions of this triangle must not be less than the third side dimension. Violation of this constraint will result in a dimensional specification that cannot be realized.

In addition, in the planar case we have to indicate the chirality of the triangle. This may be captured as a constraint that traversing the points p_1, p_2, p_3 and back to p_1, in that order, must form a counterclockwise loop.

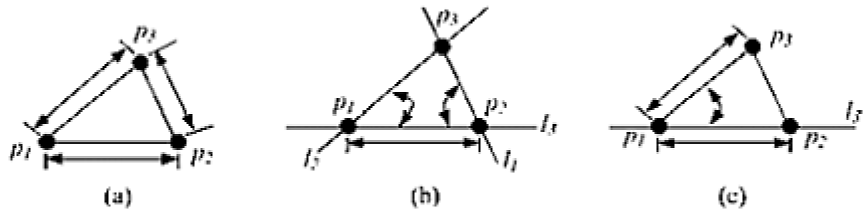


FIGURE 8.3 Relative positioning scheme for three points. (a) Simultaneous specification of three pairwise relative positions of points. (b) Simultaneous specification of a relative position of two points and two relative positions of a pair of lines. (c) Specification of a relative position between two points and the position of a point relative to a tuple of the first two points. This specification has a hierarchy.

Example 8.2 Next, based on Theorem 2.3 (angle-side-angle) for a triangle, we may decide that a tuple of three points can be dimensioned by specifying one side and its two adjacent angles in the triangle formed by these points as vertices. Figure 8.3(b) illustrates this case. The side dimension fixes the relative position of two of the points, and each of the two adjacent angle dimensions fixes the relative position of a pair of straight lines that contain the edges of the triangle.

A closer look reveals that, in fact, we have dimensioned a tuple of six objects: three points, p_1, p_2, and p_3, and three lines, l_1, l_2, and l_3, as shown in Figure 8.3 (b). We have implicitly specified six incidence constraints:

p_1, p_2 are on l_3
p_2, p_3 are on l_1
p_1, p_3 are on l_2

We then explicitly specified three dimensions $d(p_1, p_2)$, $\theta(l_2, l_3)$ and $\theta(l_1, l_3)$. Together they constitute simultaneous specification of relative positions among the six objects. These are the dimensional constraints. Other relative position dimensions, such as $d(p_1, p_3)$ and $\theta(l_1, l_2)$, are not specified, but they are determined by the dimensional constraints.

The two angle dimensions in Figure 8.3(b) cannot be chosen totally arbitrarily, because their sum must be less than 180°. In addition, in the planar case, we have to indicate the chirality of the triangle, as we did in Example 8.1.

Example 8.3 Finally, Figure 8.3(c) shows how Theorem 2.1 (side-angle-side) may lead to a dimensioning scheme for a tuple of three points. We can view this as a dimensioning scheme that uses a hierarchy. For first we can take $d(p_1, p_2)$ to be the dimension of the relative position of p_1 and p_2 to form the tuple (p_1, p_2) This tuple belongs to the general class in Table 6.2, and its reference tuple consists of a line l_3 that contains these points and any point, say, p_1, on this line. We then position the point p_3 relative to the tuple (p_1, p_2) by positioning p_3

relative to the reference tuple (l_3, p_1). This is accomplished by two polar coordinates: radial dimension $d(p_1, p_3)$ and the indicated angular dimension.

All three indicated dimensions can be chosen arbitrarily, subject only to a chirality constraint, which may be indicated by requiring that p_3 lie to the left of the directed line from p_1 to p_2.

Simultaneous specification of all ${}_nC_2$ pairwise relative positions among n geometric objects may not guarantee that the resulting collection is rigid, and these specifications may not even be consistent, as the following example illustrates.

Example 8.4 Figure 8.4(a) shows three nonparallel, coplanar lines. Their pairwise relative positioning requires only angle dimensioning, as shown. But these three angles are not independent, for they should sum to 180°. In addition, simultaneous specification of these three pairwise relationships does not yield a rigid collection of three lines. To get a rigid collection, we should impose a hierarchy by first placing l_2 relative to l_1 (using just an angle dimension between them) and then placing l_3 relative to the tuple (l_1, l_2) using two dimensions. Recall that the continuous symmetry of the 2-tuple (l_1, l_2) in this case belongs to the general class in the plane.

In Figure 8.4(b), two of the three lines are parallel. Here the two angle dimensions indicated between nonparallel lines cannot be independent. Specifying different angle values in this case leads

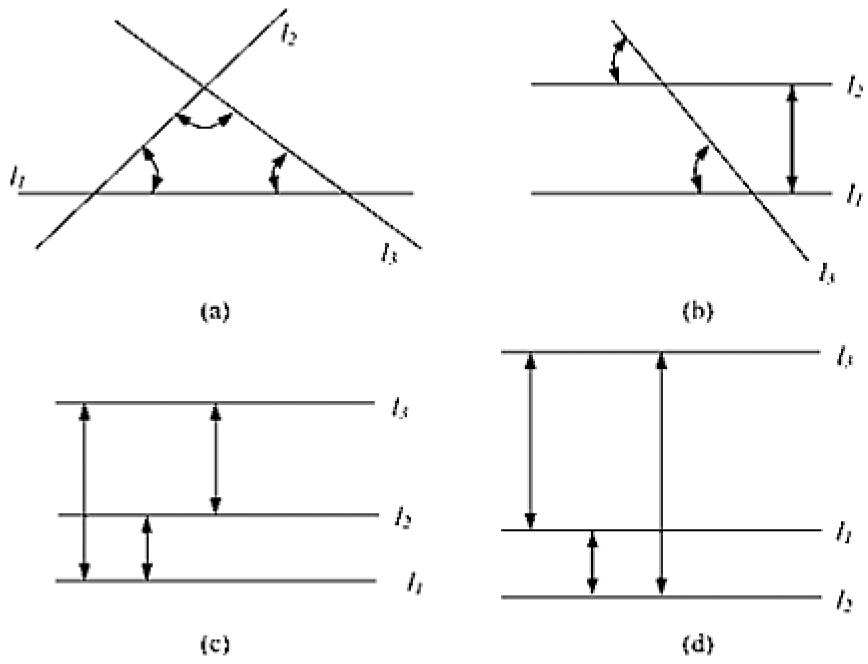


FIGURE 8.4 Simultaneous relative positioning of three coplanar lines.

to inconsistency. If we had used a hierarchy, we would have first formed the tuple (l_1, l_2) that belongs to the linear class and then placed the line l3 relative to the tuple using only one angle dimension.

If all three lines are parallel, as in Figures 8.4(c) and (d), then the three indicated dimensions cannot be independent either. A hierarchical dimensioning scheme would avoid inconsistency in this case.

These examples provide some insight into what can be expected when more than two objects are positioned simultaneously. The general problem of resolving simultaneous specification of constraints (that is, determining whether they have any solution and, if so, how many and what they are) is quite difficult. If we restrict the problem to some two-dimensional cases or simple three-dimensional cases, then they become tractable. These are the types of constraint resolutions tackled in modern CAD systems. Before we discuss these cases, let's examine some basic geometric constraints commonly used in engineering.

8.1 BASIC GEOMETRIC CONSTRAINTS

There are four major types of constraints occurring so frequently that they deserve some special treatment. These are *incidence, parallelism, perpendicularity,* and *chirality*. These constraints can be applied irrespective of whether we are using a hierarchical dimensioning scheme or a simultaneous dimensioning scheme. In fact, in many cases, such constraint specification is a prerequisite to any dimensioning or parameterization scheme. For example, we need to impose the constraint that the two planes are parallel before we can consider dimensioning or parameterizing the distance between them.

There is a curious connection between these constraints and invariance structure of some important geometric transformations. Figure 8.5 shows a Venn diagram of four geometric transformations and their invariant structures, which include the four constraints just mentioned. Let's look at each of the constraints in some detail.

8.1.1 Incidence Constraint

Incidence is a general term used to indicate that one object is completely overlapped by another object. Some examples of incidence involving just two objects are point on point, point on line, point on plane, line on line, line on plane, and plane on plane. One may argue that these are just special cases when

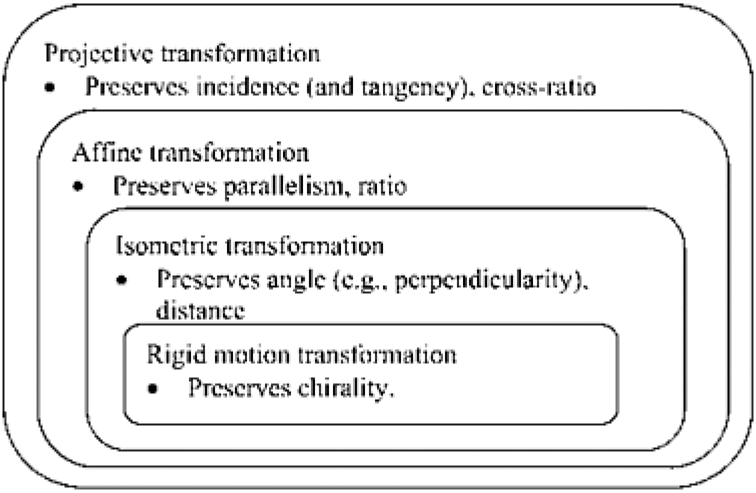


FIGURE 8.5 Hierarchy of geometric transformations and the corresponding invariances.

some distances and angles involved in relative positioning these objects assume zero values. This is indeed true, as, for example, "line on line" is a special case of two skew lines whose shortest distance and the twist angle both vanish. However, we call this out as a major constraint because it plays a crucial role in the symmetry group classification of a tuple of two objects in Table 6.4.

Incidence is preserved under a projective transformation that can be defined by ratios of affine linear functions as

$$x' = \frac{a_{11}x + a_{12}y + a_{13}z + a_{14}}{a_{41}x + a_{42}y + a_{43}z + a_{44}} \qquad (8.1)$$

$$y' = \frac{a_{21}x + a_{22}y + a_{23}z + a_{24}}{a_{41}x + a_{42}y + a_{43}z + a_{44}}$$

$$z' = \frac{a_{31}x + a_{32}y + a_{33}z + a_{34}}{a_{41}x + a_{42}y + a_{43}z + a_{44}}$$

where the coefficients a_{ij} are all real and independent. Using homogeneous coordinates, Eqs. (8.1) can be rewritten as

$$\begin{Bmatrix} \rho x' \\ \rho y' \\ \rho z' \\ \rho \end{Bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \\ 1 \end{Bmatrix} \quad (8.2)$$

or, more compactly, as

$$\rho X' = A_P X \quad (8.3)$$

The projective transformation, as represented by the invertible matrix A_P, forms a group. Under projective transformation, points map to points, straight lines map to straight lines, and planes map to planes. Projective transformation also preserves incidence relationships. For example, a 2-tuple of a plane and a line in that plane transforms to a 2-tuple of a plane and a line in that plane.

Tangency (or cotangency) is often used as a constraint, particularly when circular elements are involved. We can treat it as a particular case of incidence constraint. It is preserved under projective transformation. For example, consider a 2-tuple consisting of a circle and a line tangent to this circle. Under projective transformation, the circle may be deformed to an ellipse, but the transformed line will remain tangential to this ellipse.

Projective transformation also preserves cross ratios. But this is not of interest to us right now. Parallelism is not preserved under projective transformation. This is the next constraint in the hierarchy we will now consider.

8.1.2 Parallelism Constraint

There are three cases of parallelism constraint: line vs. line, line vs. plane, and plane vs. plane. These cases occur when the angle between them becomes zero. Parallelism can influence the symmetry classification of the resulting tuple.

Parallelism is preserved under affine transformation, which can be defined by affine linear functions as

$$\begin{Bmatrix} x' \\ y' \\ z' \\ 1 \end{Bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ y \\ z \\ 1 \end{Bmatrix} \quad (8.4)$$

or, more compactly, as

$$X' = A_A X \quad (8.5)$$

The affine transformation, as represented by the invertible matrix A_A, is a subgroup of the group of projective transformations. It maps points to points, straight lines to straight lines, and planes to planes. It also preserves incidence and tangency constraints. In addition, it preserves parallelism. For example, if two lines are parallel, then their affine transformations are also lines that are parallel.

Affine transformation also preserves ratios of segments in a line. But it does not preserve distances or angles (other than parallelism). In particular, it does not preserve perpendicularity.

8.1.3 Perpendicularity Constraint

There are three cases of perpendicularity constraint: line vs. line, line vs. plane, and plane vs. plane. These cases occur when the angle between them is a right angle. Perpendicularity can also change the symmetry classification of the resulting tuple.

Perpendicularity is preserved under isometric transformation. In fact, all distances and angles are preserved under isometric transformation. Incidence and parallelism are also preserved. Isometry is a subgroup of the group of affine transformations, where the upper left 3×3 submatrix of A_A in Eq. (8.4) is restricted to be orthogonal. That is, the matrix

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{8.6}$$

is restricted to be orthogonal. Recall that we studied orthogonal matrices in some detail in Section 2.2.

The determinant of an orthogonal matrix can be either +1 or −1. Because of this possibility, isometry cannot preserve chirality. An important consequence of this property is the fact that *if a geometric object S satisfies certain specified, unsigned distance and angle dimensions, then the mirror image of S also satisfies these dimensional constraints*. Note that this fact holds even for a subset of the geometric object. Figure 8.6 shows some simple examples involving just triangles and quadrilaterals in the plane. The triangles shown in Figures 8.6(a) and 8.6(b) are two mirror images that satisfy the side-angle-side dimensional specifications for a triangle.

If a planar quadrilateral is specified using four side dimensions and one angle dimension, then these dimensional constraints can be satisfied by the four solutions shown in Figures 8.6(c) through 8.6(f). While Figures 8.6(c) and 8.6(d) are mirror images of each other, Figures 8.6(c) and 8.6(e) are not. Only a subset of Figure 8.6(c) is a mirror image of a subset of Figure 8.6(e); this shows that we can modify a geometric object locally by mirror reflection and still satisfy the dimensional specifications.

The examples of Figure 8.6 can be generalized to any polygon. The general lesson we learn is that if we can find one solution, that is, a geometric object, that satisfies certain distance and angle dimension specifications, then

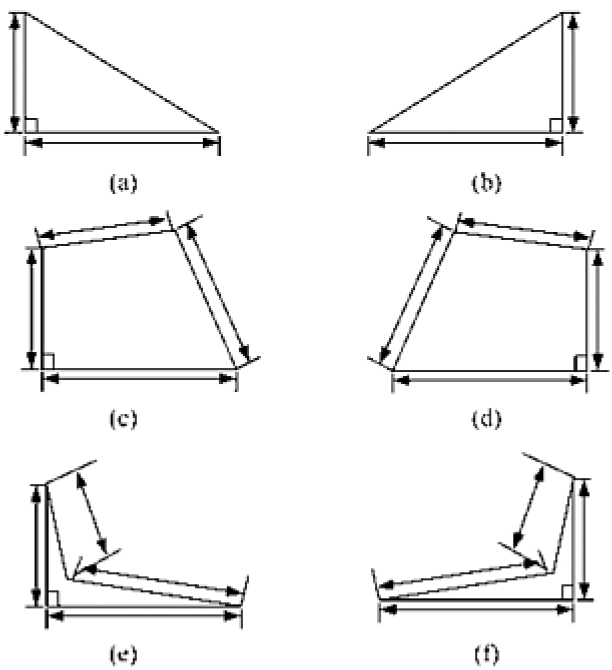


FIGURE 8.6 Multiple solutions under isometry: (a) and (b) are for triangles; (c), (d), (e), and (f) are for quadrilaterals.

we can find (finitely) many solutions by local or global mirror reflections of this solution.

8.1.4 Chirality Constraint

Chirality is a binary constraint. It specifies whether the geometric object is left-handed or right-handed. It is important for the interchangeability of parts in industry. We have seen several instances where this is an issue.

Chirality is preserved under rigid motion transformation. The group of rigid motions is a subgroup of the group of isometries, where the determinant of the orthogonal matrix in Eq. (8.6) is restricted to be +1. In addition, rigid motion preserves distances, angles (including parallelism and perpendicularity), and incidence relationships. Since distances and angles alone do not indicate chirality, it is important to impose this constraint by other means. In engineering drawings, the visual image encodes the chirality. Drawings that are more or less to scale help us distinguish the case in Figure 8.6(c) from that in Figure 8.6(e). In CAD models, a boundary representation can encode the local and global chirality. We will say more about boundary representations in the next chapter.

8.2 RIGIDITY THEORY

There is a body of literature called *rigidity theory* that deals with the question of whether a given set of dimensional constraints defines a rigid object. The rigidity theory shows that all one-dimensional and some two-dimensional problems can be solved satisfactorily using a graph structure of the constraints. It has also shown that the problem can get very hard in two and three dimensions. We will review some of the results from rigidity theory that are useful for our purpose of resolving dimensional constraints. More on rigidity theory can be found in Appendix 3.

8.2.1 Dimensional Constraint Graph

Consider a graph whose vertices, which are also called *nodes,* are geometric objects (point-sets or tuples). The edges, which are also called *arcs,* of the graph correspond to specified relative positioning dimensions between the nodes that are connected by the edges. Figure 8.1 shows a complete graph in which every node is connected to every other node. This is a special case. Generally, in a dimensional constraint graph, only some of the edges shown in Figure 8.1 will be present. From our study of the general theory of relative positioning in Chapter 7, we know that we can replace the geometric objects in the nodes of the graph by their reference elements or tuples consisting of just points, lines, planes, and helices.

From Chapter 7 we know that some of these reference elements in the nodes may be oriented. We also know that some of the indicated dimensions represented by the graph edges may be signed. Note that an edge in the graph may contain more than one dimension. For example, the edge between two nodes that stand for lines that are skew carries the shortest distance as well as the (signed) twist angle. A reference tuple may be further split into its elements, each occupying a distinct node; in this case their incidence relationship (such as point on line, line on plane) can be indicated by joining these nodes by edges that stand for incidence. Appendix 3 contains more information on dimensional constraint graphs along with some basic graph theory.

In a graph G we will indicate the vertex (or node) set by V and the edge set by E. The number of nodes, or vertices, in V will be denoted by $|V|$ and the number of edges in E will be denoted by $|E|$.

8.2.2 One-Dimensional Rigidity

Consider a set of distinct points that are collinear. If these are the reference elements under consideration, then their dimensional constraint graph takes on a simple form. Figure 8.7 shows an example along with a dimensional constraint graph. Each edge in the graph is an unsigned distance between the points connected by the edge.

For such one-dimensional problems it is clear that if the graph is a tree, then the points can be positioned using the hierarchy implied by the tree structure, and the problem is solved. The same argument holds if we replace the collinear points by parallel lines in a plane or by parallel planes in space. In fact, Figures 1.2, 1.4, and 1.9 illustrate how parallel lines and parallel planes in actual parts are dimensioned.

If the dimensional constraint graph is not a tree, then it is not connected or it contains cycles. Then we have the following problems:

1. If the graph is not connected, then we have an *underdimensioning* problem. So the specified dimensions do not define a rigid collection of objects. This is also referred to as an *underconstrained* problem.
2. If the graph contains one or more cycles, then we have an *over-dimensioning* problem. This is also referred to as an *overconstrained* problem. Overdimensioning need not mean that these dimensions are inconsistent; it just means that there is redundancy that may lead to inconsistency. See Figure 8.8.

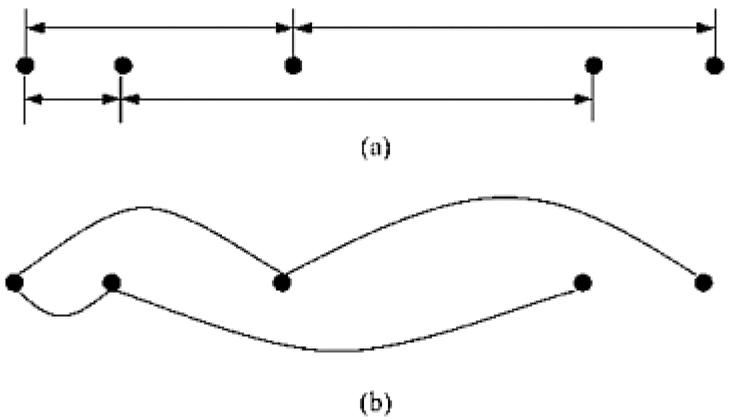


FIGURE 8.7 (a) Simultaneous dimensioning of relative positions of collinear points. (b) The corresponding dimensional constraint graph.

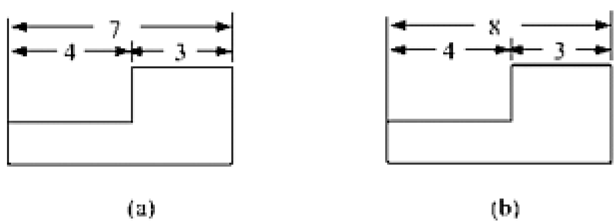


FIGURE 8.8 (a) Overdimensioning that is consistent. (b) Overdimensioning that is inconsistent.

The necessary and sufficient condition for a connected graph to be a tree is that

$$|E|=|V|-1 \tag{8.7}$$

This is a simple combinatorial condition that can easily be checked.

The dimensional constraint tree may yield as many as $2^{|V|-1}$ different solutions, only one of which the designer may want. This is due to the fact alluded to in Section 8.1.3. Since only unsigned distances are specified, the mirror image of any solution or subsolution is also a solution to the problem. The multiplicity can be reduced if some of the dimensions are signed values.

8.2.3 Two-Dimensional Rigidity

Consider a set of *N* distinct points in a plane and specify some pairwise distances between them to form a dimensional constraint graph. See Figure 8.9 for an example. With these distance dimensions, do we have a rigid collection of these *N* points? It is not easy to answer this general question. A partial answer that ensures the so-called generic rigidity (see Appendix 3 for a description of various mathematical notions of rigidity) is provided by the following theorem.

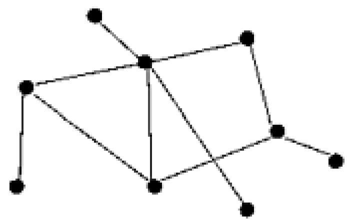


FIGURE 8.9 A dimensional constraint graph with only distance dimensions for a set of points in a plane.

Theorem 8.1: Laman's Theorem *A graph G with edge set E and vertex set V is generically rigid in the plane if only if there is a subset F of E that satisfies*

1. $|F|=2|V|-3$*, and*
2. *For all* $F' \subseteq F,\ F' \neq \varnothing,\ |F'| \leq 2k-3$ *where k is the number of vertices that are endpoints of edges in F′.*

This is only a combinatorial characterization of the rigidity problem. Obviously, we need to add other conditions, such as triangle inequality, to ensure that we can actually realize an embedding of this rigid collection in a two-dimensional plane. Even if it is realized, there can be as many as $2^{|V|-2}$ different solutions to this problem. This is again for the reasons mentioned in Section 8.1.3.

A mechanical model of the foregoing problem is provided by a framework in which the nodes are revolute joints and the edges are rigid bars of the given lengths. The revolute joints permit relative rotational motion of the bars that are joined. If such a framework can be realized, it is either a rigid structure or a flexible mechanism. Laman's first condition ensures that there are just enough edges to locate the nodes. His second condition ensures that no subgraph is overbraced and that the edges are distributed "wisely" throughout the graph. Applying his theorem will indicate, quite correctly, that the planar framework represented by Figure 8.9 is not generically rigid. It will also indicate that the planar frameworks in upcoming Figures 8.11 and 8.13 are generically rigid. This doesn't mean that the frameworks are rigid; see Appendix 3 for a counterexample.

Laman's theorem is applicable only when the simultaneously specified dimensions are all distances. If angles are also involved, then even a similar combinatorial characterization of rigidity is not yet available, in spite of intense search.

8.2.4 Three-Dimensional Rigidity

An obvious analogue of Laman's theorem for a set of points in three-dimensional space fails to provide sufficient conditions for rigidity. Here are Laman's conditions in three-dimensional space.

1. $|F|=3|V|-6$
2. For all $F' \subseteq F$, $F' \neq \emptyset$, $|F'| \leq 3k-6$ where k is the number of vertices that are endpoints of edges in F'.

These are only necessary conditions, but they are not sufficient. The "double banana" graph of Figure 8.10 satisfies Laman's conditions, but it is not rigid.

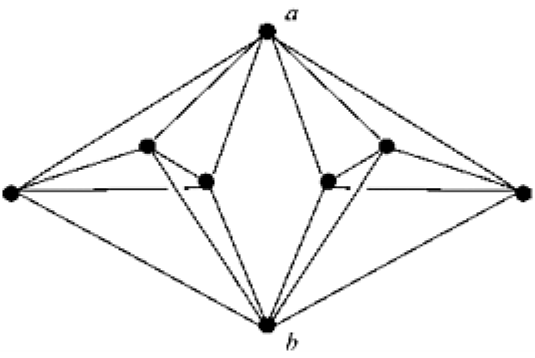


FIGURE 8.10 A "double banana" framework in three-dimensional space that is not rigid. It can twist about an axis through the vertices labeled *a* and *b*.

These results show that there is no simple way to simultaneously specify some distance dimensions on a wireframe and be sure that these dimensions define a rigid structure. The problem gets more complicated if angle dimensions are also involved.

8.3 INDUCING HIERARCHY IN SIMULTANEITY

The most successful attempt to date in solving dimensional constraint problems involves inducing a hierarchy in the flat structure inherent in simultaneous dimensional specifications. It is based on the following ideas.

1. Breaking down the problem into smaller ones, which are sometimes called *clusters*
2. Solving these small problems by analytical or numerical methods
3. Combining these clusters to form bigger clusters, again by analytical or numerical methods, till the whole problem is solved

In general, there is no guarantee that this divide-and-conquer strategy will always work. But for many practical problems that are two-dimensional and are not too complicated, it has worked quite well. We will illustrate this idea in the following simple but representative examples.

Example 8.5 Figure 8.11 dimensions a planar pentagon using seven distances simultaneously We can also view this as dimensioning a rigid tuple of five points *A, B, C, D,* and *E.* Laman's theorem can be used to verify that such a specification is

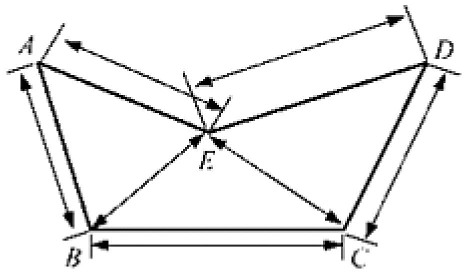


FIGURE 8.11 Dimensioning a pentagon using seven distance dimensions.

generically rigid. We can construct the pentagon by first constructing the three triangles *ABE, BCE,* and *CDE* because all the sides are known in these triangles. Then we just join them together by relative positioning to get the pentagon. The entire construction can be accomplished using "ruler and compass." We have thus induced a hierarchy in constructing the pentagon.

E**xample 8.6** Figure 8.12 shows another dimensioning of a simple pentagon in a plane. This involves five distance dimensions and two angle dimensions, all specified simultaneously. We can induce a hierarchy by doing the following:

1. Construct triangles *ABE* and *CDE,* because they are well defined by the side-angle-side theorem. In this process, edges *BE* and *CE* will also be constructed.
2. Then construct triangle *BCE* using the specified side *BC* and the sides *BE* and *CE* constructed in the previous step.

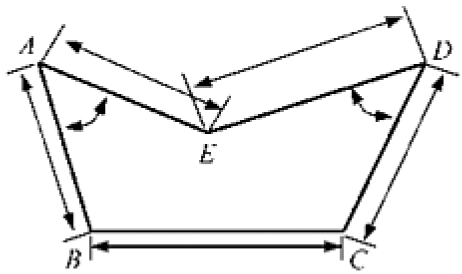


FIGURE 8.12 Dimensioning a pentagon using five distance dimensions and two angle dimensions.

This completes the construction of the pentagon. This required only a knowledge of constructing triangles. With that, we were able to come up with a hierarchy of constructions to define the pentagon.

Example 8.7 A planar hexagon *ABCDEF* has been dimensioned in Figure 8.13 using nine distance dimensions. It can also be viewed as a dimensioning of a rigid tuple of six points. We can again apply Laman's theorem to make sure that this specification is generically rigid. We can induce a hierarchy in this flat dimensional specification using the following construction procedure.

1. Construct triangles *AEF* and *BDC,* because we know all the three sides in these triangles.
2. Combine these two triangles; that is, position them relative to each other, using the distance specifications of *AB, CF,* and *DE.* This is a little tricky, but it can be done.

The first step can be accomplished using simple "ruler and compass" construction. But the second step involves some detailed calculations.

The trickiness alluded to in the second step in Example 8.7 will become clearer when one studies Figures A3.9 and A3.10 in Appendix 3 and the explanation contained therein.

It is possible to introduce intrinsic dimensions also as part of the constraints, as the following example illustrates.

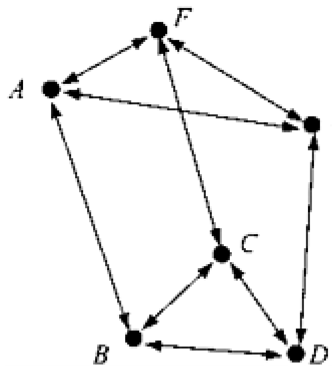


FIGURE 8.13 Dimensioning a hexagon *ABCDEF* using nine distance dimensions.

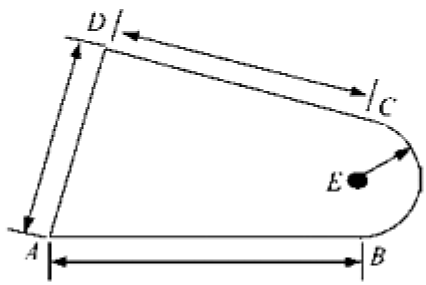


FIGURE 8.14 A dimensioned planar sketch involving one circular arc.

Example 8.8 Figure 8.14 shows a planar sketch in which an intrinsic dimension (radius of a circular arc) and relational dimensions (distances between points) simultaneously constrain the whole object. *E* is the center of the circular arc. Tangency constraint is implicit at *B* and *C*. The following construction procedure induces a hierarchy on this dimensioning.

1. Construct the right triangles *ABE* and *DCE*. All dimensions are available for their construction. In this process, the edges *AE* and *DE* will be determined.
2. Construct the triangle *AED*. All sides of this triangle are now known, due to the previous step.
3. Now put the three triangles together to form the final object.

Here again, we can have a finite number of multiple solutions, of which only one the designer may want. The general idea of inducing hierarchy in dimensional constraints can be automated using some clever algorithms and heuristics.

8.4 SUMMARY

Dimensional constraints arise when certain dimensions are specified simultaneously without any hierarchy. We have to deal with such specifications in engineering practice, especially in planar (that is, two-dimensional) sketches that are often used in preliminary steps while defining part geometry. In general, it is difficult to decide whether these specifications lead to valid geometric objects. In particular cases, as illustrated in simple Examples 8.5–8.8, these constraints can be resolved and valid objects that obey these constraints can be created. Interestingly, the most successful approach to resolve the constraints relies on geometric construction procedures that induce a hierarchy in the flat structure inherent in the constraint specification. Modern CAD systems have adopted such techniques to handle dimensional constraints in their software.

Specification of dimensional constraints is also a means of parameterizing a geometric object and indicating an instance in the parameter family. If some constraints involve inequalities (such as an interval within which a geometric parameter should lie), then the specification restricts the choice to a subset of the parameterized family of objects. In this light, dimensions can be viewed as equality constraints on some geometric parameters. Some ISO data exchange standards take this point of view.

8.5 EXERCISES

1. Construct dimensional constraint graphs for the problems in Examples 8.2, 8.3, 8.6, and 8.8. (*Hint:* See Appendix 3 for an example of a dimensional constraint graph.)
2. Construct possible multiple solutions to each of the problems in Examples 8.5, 8.6, 8.7, and 8.8.
3. Figure 8.15 shows a planar sketch of a pentagon. Is it valid? If so, give a construction procedure that builds this object hierarchically.
4. A dimensioned planar sketch is shown in Figure 8.16. What are the implied tangency constraints? Is this dimensioning valid? If so, give a construction procedure to build this object.

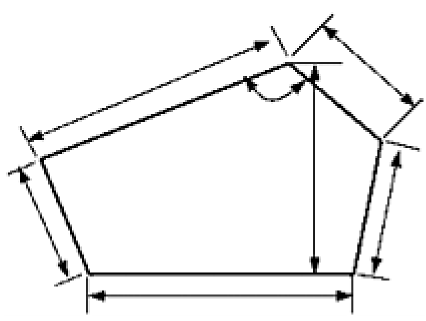


FIGURE 8.15 A dimensioned planar pentagon.

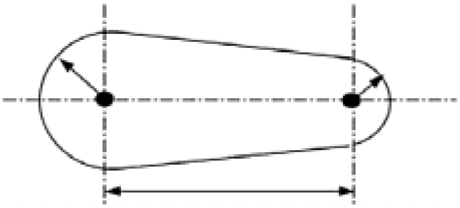


FIGURE 8.16 A dimensioned planar sketch involving two circular arcs.

5. Does Laman's Theorem 8.1 help us in determining whether a dimensioned two-dimensional sketch is over- or underdimensioned?

8.6 NOTES AND REFERENCES

Rigidity theory has many applications, including biochemistry and ceramics, and it is an active field of research (Thorpe and Duxbury, 1999). An important paper by Laman (1970) started much of the current development in rigidity graph theory.

Simultaneous specification of dimensional constraints is also called a *variational* constraint scheme. Variational geometry deals with resolving such constraints to build the underlying geometry. Early attempts in this regard depended on numerical solutions to the set of constraint equations. Later, more robust methods based on graph theoretic ideas were proposed by Owen (1991, 1996), Bouma et al. (1995), and Fudos and Hoffmann (1997). These methods have also been successfully implemented in commercial CAD systems. Research on a better understanding of the dimensional constraint problem and on developing smarter algorithms and heuristics is still ongoing.

ISO 10303–108 (2003) deals with data exchange issues involving parameterization and constraints for explicit geometric models.

9
Dimensioning Solids

All manufactured objects have solidity. Even thin parts have a certain thickness. Although in our study thus far we discussed dimensioning curves and surfaces and then described how to dimension the geometric relationship between these objects, our ultimate goal is to dimension solids. A brief mathematical theory of solids is reviewed in Appendix 4, which should be read along with this chapter. What we have studied so far is necessary to dimension solids, but it is not sufficient, as the following case illustrates.

Figure 9.1 shows four different solids with the same dimensions. Even the surfaces involved in each solid are the same, as shown in Figure 9.2. These surfaces consist of an unbounded cylinder C of diameter d and two unbounded parallel planes, P_1 and P_2, separated by a distance h and perpendicular to the axis of the cylinder. We can define the constraints and dimensions (or parameters) of this collection of surfaces as follows.

Constraints:

$P_2 \| P_1$

Axis of $C \perp P_1$

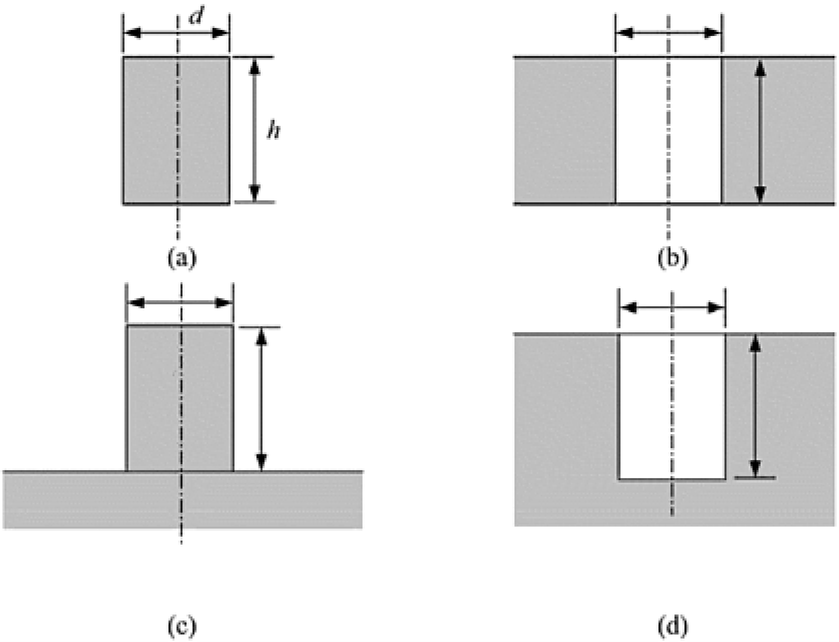


FIGURE 9.1 Four different solid objects with the same dimensions.

Parameters:
Distance h between P_1 and P_2 (relational dimension)
Diameter d of C (intrinsic dimension)

But this is not enough to define a solid object unambiguously. As Figure 9.1 illustrates, these constraints and dimensions may correspond to (a) a finite, solid cylindrical pin, (b) a cylindrical through-hole, (c) a cylindrical stud

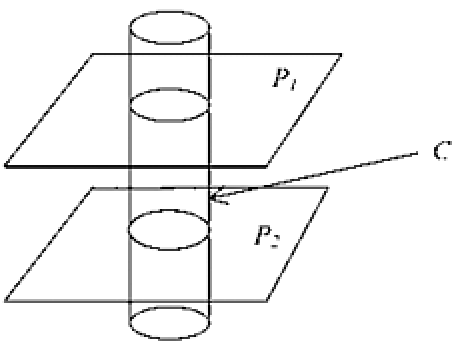


FIGURE 9.2 Surfaces involved in the four solids of Figure 9.1.

protruding from a planar half-space, or (d) a cylindrical blind hole. How can we distinguish among these choices?

A key to resolving such ambiguities is to realize that we are not dimensioning just a rigid collection of surfaces as indicated by the given constraints and parameters for Figure 9.2, but we are defining a solid. A brief introduction to a theory of solids is given in Appendix 4, which should be read along with this chapter. There are many ways to define solids. Let's start with a simple case of a solid tetrahedron and see how it can be modeled and dimensioned.

9.1 DIMENSIONING A SOLID POLYHEDRON

In three-dimensional space, a tetrahedron is the simplest polyhedron (that is, a solid bounded by only flat faces). For this reason it is also called a three-dimensional simplex. One way to dimension (or parameterize) it is to specify all of its six edges, as shown in Figure 9.3(a). But it can be ambiguous because its mirror image, shown in Figure 9.3(b), also satisfies these dimensional specifications. So we need to indicate the chirality of the tetrahedron if we want to avoid this ambiguity. This can be accomplished in several ways. One popular option is to define a boundary representation for the tetrahedron as follows.

The boundary of a solid tetrahedron consists of four triangular faces. Each triangular face can be oriented in space by specifying an outward normal that points away from the solid. This orientation can also be encoded by specifying an order for the three vertices of the triangle so that a right-hand rule can be used to obtain the outward normal direction. The tetrahedron can be embedded in three-dimensional space with a right-handed Cartesian coordinate system, as shown in Figure 9.4. With these preliminaries, a very

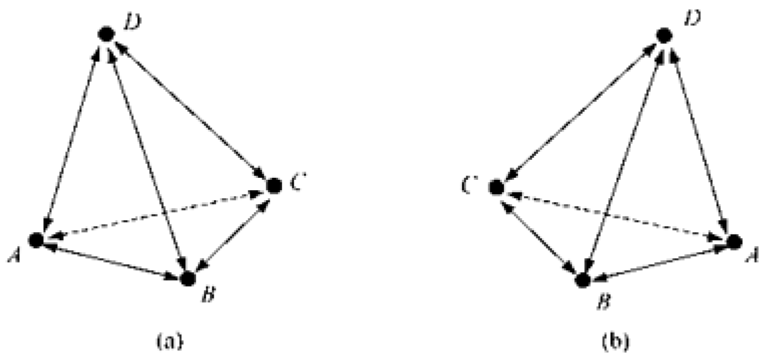


FIGURE 9.3 Dimensioning a solid tetrahedron by specifying its six edges. (a) and (b) are mirror reflections of each other.

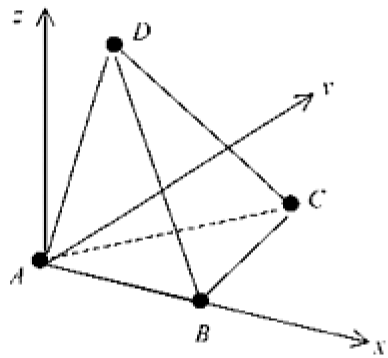


FIGURE 9.4 Embedding a tetrahedron in three-dimensional space, and an associated Cartesian coordinate system.

simple boundary representation for the tetrahedron can be given in the form of a vertex table and a face table, as shown in Table 9.1 and Table 9.2, respectively.

The vertex table contains the *x*-, *y*-, and *z*-coordinates of the four vertices of the tetrahedron. Six of the 12 coordinates can be assigned zero values without any loss of generality because, as Figure 9.4 shows, a Cartesian coordinate system can be assigned to an arbitrary tetrahedron by choosing the origin as one of the vertices, the *x*-axis along one of the edges, and the *xy*-plane containing one of the faces. The remaining six coordinates form the unfilled cells in Table 9.1. These can be filled in directly to define a tetrahedron, or they can be derived from the specification of the six edge dimensions shown in Figure 9.3(a). The latter method involves some calculations, but they are not difficult.

The face table, shown in Table 9.2, lists the vertices of each of the four triangular faces, numbered 1, 2, 3, and 4. The three vertices for each face are arranged in an order so that applying the right hand-rule for each list of vertices will yield the direction of the outward normal for that face. Together the vertex table and the face table form a boundary representation for the tetrahedron and define it unambiguously.

TABLE 9.1 Vertex Table for the Tetrahedron Shown in Figure 9.4

	x	*y*	*z*
A	0	0	0
B		0	0
C			0
D			

TABLE 9.2 Face Table for the Tetrahedron Shown in Figure 9.4

1	*A*	*C*	*B*
2	*A*	*B*	*D*
3	*B*	*C*	*D*
4	*C*	*A*	*D*

Using this type of boundary representation has two benefits. First, it encodes chirality so that the two cases shown in Figure 9.3 can be distinguished. Second, it can also distinguish a solid tetrahedron from a tetrahedral void (that is, the complement of the solid tetrahedron). Thus, *a boundary representation of a tetrahedron provides an embedded instance of a parameterized solid model of a tetrahedron*.

Even the simple case of a tetrahedron reveals some subtle issues related to dimensioning, parameterization, and boundary representation. We first note that the boundary representation itself is not parameterized, except when the six coordinate entries in the unfilled cells of Table 9.1 are treated as the parameters of the tetrahedron. But a tetrahedron is not always dimensioned by coordinate dimensioning of its vertices. The designer may have chosen, for some functional reason, to dimension a solid tetrahedron as in Figure 9.3(a). These dimensions determine, through some calculations, the coordinates of the unfilled cells in the vertex table of the boundary representation. This mapping between dimensions of a solid and its boundary representation is easily established for simple objects like a tetrahedron. But the mapping quickly gets quite complex for general polyhedra, as described next.

It turns out that the simple vertex and face tables used in boundary representation of a tetrahedron can be extended to any polyhedron. To do this, any polygonal face of a polyhedron that is not a triangle has to be triangulated, because the face table assumes that the faces are all triangles. This can always be done, and let's assume, without any loss of generality, that all faces of our polyhedron are triangles.

Now let's assume that this polyhedron is dimensioned by some distance and angle dimensions. Deriving a boundary representation for this polyhedron involves the following two steps.

1. *Creating a vertex table.* For this, we need to determine the coordinates of all the vertices in the polyhedron. Sometimes this is referred to as determining an embedding of the model in Euclidean space. As we saw in Chapter 8, this may involve solving a dimensional constraint problem in three-dimensional space. Unless the polyhedron is simple, this can be a difficult problem. If, for example, all the edges shown in Figure 8.10 are dimensioned, we will not be able to create a vertex table for it because it is not rigid.
2. *Creating a face table.* It is the oriented triangles on the boundary that give solidity to the polyhedron. Dimensioning (or parameterizing) alone will not create this face table. It has to be created by other means. Care must be exercised to ensure that the

oriented triangles completely cover the boundary of the solid polyhedron.

Before we leave this section, let's make some observations about boundary representations. The vertex and face tables are the simplest, but not the most efficient, way to capture the boundary of a polyhedron. It is advantageous to store the incidence relationships (e.g., what edges are incident at a vertex, what faces are incident at an edge) and the adjacency relationships (e.g., what edges are adjacent in a face) in a polyhedral boundary data structure. For solids with curved boundaries, we need to define, build, and maintain more sophisticated data structures for boundary representation. This happens to be one of the major tasks performed inside CAD systems. From a theoretical viewpoint, it is sufficient to know that these boundary representations are capable of informing us of what bounds what and the incidence and adjacency relationship for any geometric entity they have.

Finally, note that it is not easy to modify a boundary representation directly, because even a simple change in one element of the boundary may lead to a large number of changes to be made to keep the boundary representation valid. Many connectivities, in the form of incidence and adjacency relationships, may need to be reset, and many point, curve, and surface representations may need to be recomputed. This is why, in modern CAD systems, the current boundary representation is partially or wholly discarded when a model change is made and a new one is computed. This fact argues against dimensioning just the boundary representation of a solid model.

9.2 DIMENSIONING PROCEDURALLY DEFINED SOLIDS

Polyhedral boundary representation discussed in the last section is an example of what is called an *explicit representation* of solids. It has a flat structure. It describes the solid in its final form without any reference to how it might have been constructed. An alternative way to represent a solid is to give the procedure used in constructing it. We will examine some of the powerful procedural representations of solids next.

9.2.1 Constructive Solid Geometry

Sets can be manipulated by Boolean operations, such as complementation, union, intersection, and subtraction (difference). Since solids are modeled as point-sets, we can modify them using Boolean operations. (In fact, we should use regularized Boolean operations, as described in Appendix 4.) Let's illustrate this idea for the four solid objects discussed earlier and shown in Figure 9.1. First, the three surfaces in Figure 9.2 are associated with three half-spaces, denoted H_1, H_2, and H_3, shown in Figure 9.5. Here H_1 and H_2 are half-spaces bounded by the planes P_1 and P_2, respectively, and H_3 is a solid cylindrical half-space bounded by the cylinder C. Then the following hold:

1. $(H_1 \cap H_2) \cap H_3$ defines a finite, solid cylindrical pin, as in Figure 9.1(a).
2. $(H_1 \cap H_2) - H_3$ defines a cylindrical through-hole, as in Figure 9.1(b).
3. $(\bar{H}_2 \cup H_3) \cap H_1$ defines a protruding cylindrical stud, as in Figure 9.1(c).
4. $H_1 - (H_2 \cap H_3)$ defines a cylindrical blind hole, as in Figure 9.1(d).

Several Boolean operations on point-sets have been used in these expressions. Here, overbar indicates complementation, ∪ stands for union, ∩ stands for intersection, and − stands for subtraction. These Boolean expressions give a tree structure to the construction procedure. They unambiguously define the indicated solid objects. Figure 9.6 shows the Boolean expressions for the construction of the solids in Figure 9.1, each in the

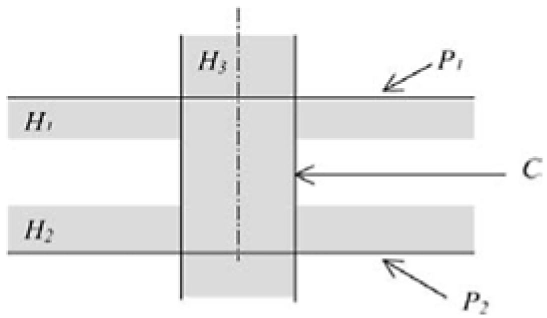


FIGURE 9.5 Half-spaces associated with the simple surfaces of Figure 9.2.

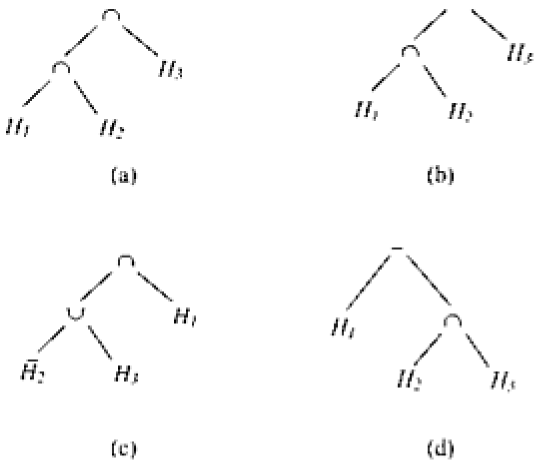


FIGURE 9.6 CSG tree hierarchy for the construction of each of the solids in Figure 9.1. The half-spaces H_1, H_2, and H_3 are shown in Figure 9.5.

form of a tree. It is called a *constructive solid geometry* (CSG) tree for that particular solid. Its leaf nodes are half-spaces, such as H_1, H_2, and H_3, shown in Figure 9.5 and explained earlier. They may also be fully dimensioned solids. Its interior nodes are binary Boolean operations (that is, union, intersection, and difference) on point-sets. Sometimes complementation, which is a unary operation, may also be specified at an interior node. The result of each Boolean operation is a subsolid. The root operation defines the final solid. Note that the hierarchical nature of the CSG tree makes it convenient to build solids from simpler solids, and the process can be continued recursively as long as we want.

Each primitive half-space at the leaf nodes may have intrinsic dimensions (or parameters). More on dimensioning primitive half-spaces can be found in Sections 3.1.6 and 4.1.9. By convention, at any interior node that stands for a binary Boolean operation, the solid on the right is positioned relative to the solid on the left before the Boolean operation is performed. Here we introduce the relational dimensions (or parameters) depending on the symmetry group classification of the left and the right solid.

It is instructive to examine each of the CSG trees in Figure 9.6 and the solids they construct in some detail.

Example 9.1 In Figure 9.6(a) we start with the half-spaces H_1 and H_2, which have no intrinsic dimensions. Half-space H_2 is first positioned relative to half-space H_1 by their bounding planes, because these are their reference elements. These bounding planes are constrained to be parallel, and their outward normals are antiparallel facing away from each other. We then require only one dimension for the relative position, and it is the distance between the parallel planes. The result of the intersection operation is an unbounded slab, which belongs to the planar class. The center plane of the slab can be chosen as the reference element for the slab.

We then bring in the solid cylindrical half-space H_3, which has an intrinsic dimension, namely, its diameter. Its symmetry belongs to the cylindrical class, and its reference element is its axis. We position the cylinder relative to the slab that had been constructed so far by positioning the axis of H_3 relative to the center plane of the slab. Here we require that these reference elements be perpendicular, and then perform the intersection operation. It results in the cylindrical pin shown in Figure 9.1(a), whose symmetry belongs to the revolute class.

Example 9.2 Next consider Figure 9.6(b). We construct the unbounded slab using the half-spaces H_1 and H_2, as we did in Example 9.1. We then position the solid cylindrical half-space H_3 relative to the slab, as in that example, but now perform a subtraction of H_3 from the slab. It results in the cylindrical through-hole shown in Figure 9.1(b), whose symmetry also belongs to the revolute class.

Example 9.3 Now consider Figure 9.6(c). Here we first complement the half-space H_2 and position the solid cylindrical half-space H_3 relative to it so that the axis of H_3 is perpendicular to the plane bounding the complement of H_2. A union operation is performed, which results in a solid of revolution. We then bring in the half-space H_1 and position it relative to the solid of revolution so that the bounding plane of H_1 is perpendicular to the axis of revolution and is at a specified distance from the bounding plane of the complement of H_2. (See Section 7.4—in particular, Fig. 7.9—for details on relative positioning a revolute class and a planar class.) An intersection operation then yields the pin protruding from a planar half-space, as shown in Figure 9.1(c), whose symmetry also belongs to the revolute class.

Example 9.4 Finally, consider Figure 9.6(d). We first position the solid cylindrical half-space H_3 relative to half-space H_2 so that the axis of H_3 is perpendicular to the bounding plane of H_2 and perform the intersection operation. This results in an unbounded cylinder that is cut in half, whose symmetry belongs to the revolute class. We then position this half-cylinder relative to half-space H_1 so that the axis of the half-cylinder is perpendicular to the bounding plane of H_1 and the end face of the half-cylinder is at a specified distance from this plane, and we perform the Boolean subtraction. This results in the cylindrical blind hole shown in Figure 9.1(d), whose symmetry also belongs to the revolute class.

Thus a CSG tree representation is a procedural representation of a solid because it gives a procedure to construct the solid. A hierarchical dimensioning (or parameterizing) scheme can be woven quite naturally in this tree structure, as the examples demonstrate. The leaf nodes carry intrinsic dimensions. Interior nodes that stand for binary Boolean operations also carry relational dimensions. A boundary representation for each solid in Examples 9.1 through 9.4 can be derived from the CSG tree. This process of derivation is called *boundary evaluation,* because the boundary is deemed as the final result of various relative positioning and Boolean operations.

Boolean operations can also be performed on two-dimensional regions. So it is possible to start with primitive half-spaces in a plane and to construct dimensioned planar regions using a CSG tree structure. A planar polygonal region, for example, can be constructed using a CSG tree in which the leaf nodes are all half-planes (bounded by straight lines). Figure 9.7 illustrates five half-planes, h_1, h_2, h_3, h_4, and h_5, that can be used to define polygonal regions.

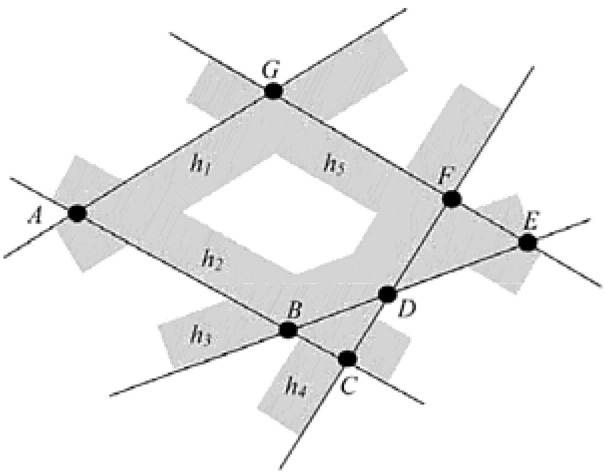
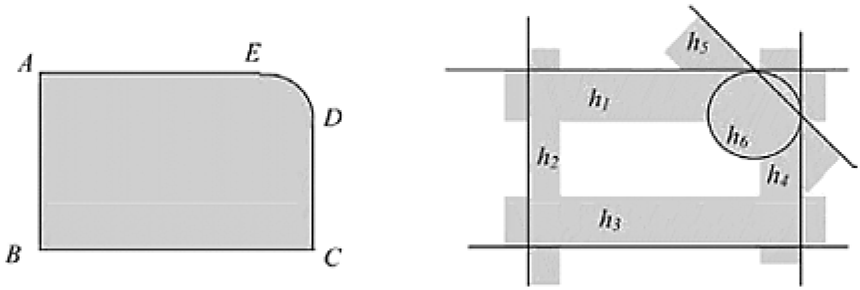


FIGURE 9.7 Half-planes used in the construction of a planar polygonal region.

A convex polygonal region bounded by a convex *polygon ABDFG* is constructed by the Boolean expression $h_1 \cap h_2 \cap h_3 \cap h_4 \cap h_5$ or the associated CSG tree. A nonconvex polygonal region bounded by the polygon *ACDEG* is constructed by the Boolean expression



(a) (b)

or the associated CSG tree. In either case, the half-planes and their Boolean combinations have to be positioned relative to each other using some relational dimensions.

It turns out that given any planar polygonal region (convex or otherwise), it is possible to come up with a CSG tree representation for it. If the planar region is not bounded by polygons, then more work is required to find a CSG representation for it, as Figure 9.8 illustrates. In this simple example, which involves a circular arc, tangency is implied at D and *E* in Figure 9.8(a). In addition to the four half-planes h_1, h_2, h_3, and h_4 and the circular disk h_6, we need to introduce a half-plane h_5, as shown in Figure 9.8(b), so that the planar region can be created by the Boolean expression $(h_1 \cap h_2 \cap h_3 \cap h_4 \cap h_5) \cup h_6$

Constructive solid geometry representation is not unique. That is, the same solid (in two or three dimensions) may be represented by more than one CSG tree. This shouldn't be a surprise to us because we know that the same object may be dimensioned in multiple ways.

9.2.2 Sweeps

The CSG tree is not the only procedural representation for solids. In Section 4.3 we saw how useful surfaces can be generated by sweeping curves. Sweeps are also a popular means of constructing solids. For this, we first start with a dimensioned two-dimensional region. The two-dimensional region may be created using a CSG approach, as explained toward the end of the last section.

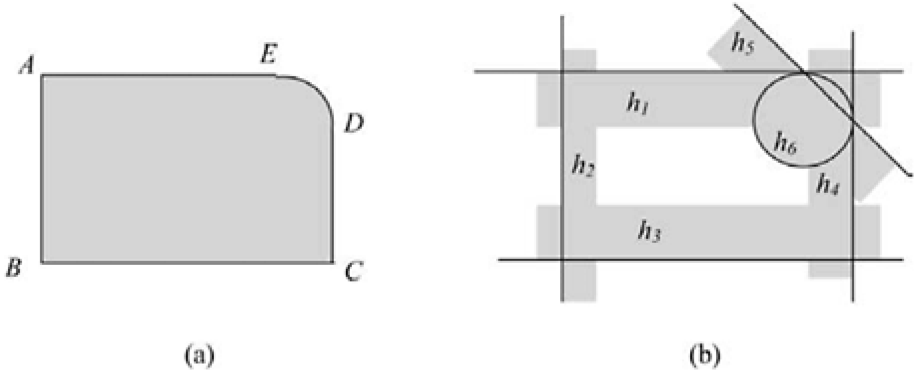


FIGURE 9.8 (a) A planar region. (b) Half-spaces needed to create the planar region using a CSG tree.

But more often, in mechanical CAD systems, one starts with a planar sketch with some dimensional constraints, as discussed in Chapter 8. A dimensioned planar sketch defines one or more simple closed curves. Figure 9.9 illustrates what we mean by a simple closed curve. It closes on itself, as in Figure 9.9(b), but it does not intersect itself as do the closed curves in Figures 9.9(c) and 9.9(d). The closed curve can consist of several pieces, such as straight-line segments and circular arcs. The power of simple closed curves lies in the following theorem, which would seem obvious but defies a simple proof.

Theorem 9.1: Jordan Curve Theorem *Every simple closed plane curve divides the plane into two, and only two, components.*

One of these components is the bounded planar region inside the curve and the other is the unbounded region outside the curve.

Based on this simple idea, a dimensioned planar sketch that forms a simple closed curve can be used to define a dimensioned two-dimensional (2D) region. The 2D region thus created is then swept (translationally, rotationally, or in some more involved way) to create a dimensioned solid. This is how several geometric "features" are created in a CAD system. A solid created by a sweep operation can then, for example, be used as a leaf node in a CSG tree.

The most general translational sweep can be defined as a *Minkowski addition*. This is defined for two point-sets embedded in Euclidean space with a coordinate reference system. Using the position vector of points, Minkowski addition of two point-sets *A* and *B* is defined as follows.

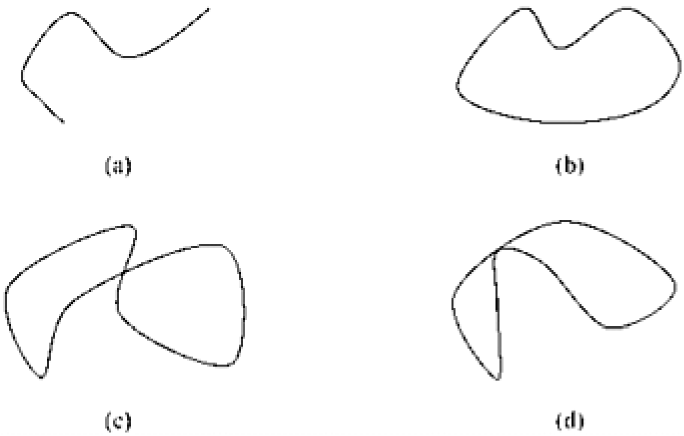


FIGURE 9.9 Different types of planar curves. (a) An open curve. (b) A closed curve that is also simple. (c) and (d) show closed curves that are not simple.

Definition 9.1: Minkowski Addition

The point-sets *A* and *B* can be of any dimensionality. Note that Minkowski addition is commutative; that is, $A \oplus B = \{a + b : a \in A, b \in B\}$

Figure 9.10 shows a simple Minkowski addition of a line segment and a circular disk. Figure 9.11 shows the Minkowski addition of a circle and a circular disk. In both cases, the Minkowski addition can be viewed as a sweep of the circular disk *B* so that its center lies on the line segment *A,* as in Figure 9.10, or on the circle *A,* as in Figure 9.11. If the set *A* is chosen to be a two-dimensional region, as illustrated in Figure 9.12, its Minkowski addition with a circular disk *B* is obtained by sweeping disk *B* so that the center of the disk is positioned at every point of *A*. Note in Figure 9.12(b), that after the sweep, all convex corners have been rounded by an amount determined by the radius of *B*. This fact can be exploited in a blending operation, such as rounding and filleting, discussed in the next section.

In general, Minkowski addition can be viewed as a translational sweep of one of the point-sets, which is positioned so that its origin point lies on every point of the other point-set. The sweep is translational in the sense that the point-set that is being swept is not subjected to any rotation during the sweep; it is merely translated. A dimensioning scheme for Minkowski addition can be derived thanks to the following theorem.

Theorem 9.2: Minkowski Addition Congruence Theorem $A \oplus B$ *is congruent to* $A' \oplus B'$ *if A is congruent to A′, B is congruent to B′, and the relative orientation of A to B is the same as the relative orientation of A′ to B′.*

Note that only the orientation part of the relative positioning matters in Theorem 9.2. See Figure 9.13 for an illustration of this fact. From this theorem we infer that dimensioning (parameterizing) *A* and *B* and dimensioning (parameterizing) their relative orientation will also dimension (parameterize) $A \oplus B$

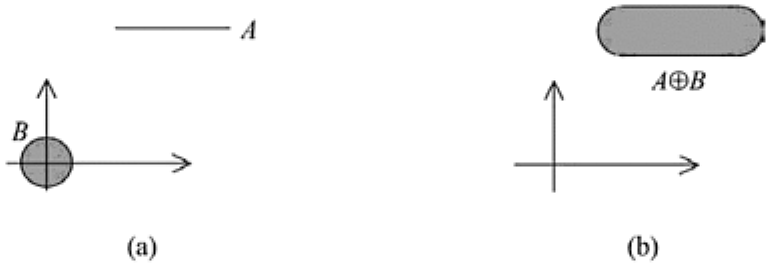


FIGURE 9.10 Minkowski addition of a line segment and a circular disk.

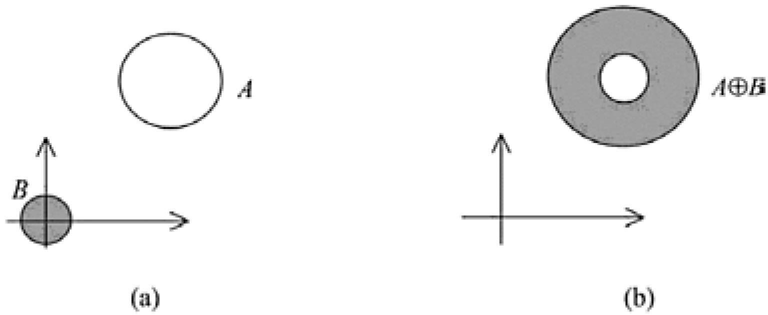


FIGURE 9.11 Minkowski addition of a circle and a circular disk.

If one of the point-sets, say, *B,* happens to be a circular disk (in two-dimensional applications) or a spherical ball, then the relative orientation of *B* with respect to any arbitrary *A* is irrelevant owing to the symmetry of *B*. Therefore, we need not worry about dimensioning this relative orientation at all. In addition, the only intrinsic dimension for such *B* is its radius (or diameter). Figures 9.10, 9.11, and 9.12 exploit this special property of a circular disk. The resulting Minkowski sum $A \oplus B$ in these figures is called the constant-radius offset of the point-set *A*. Sometimes, the boundary of $A \oplus B$ is also called the constant-radius offset of the boundary of *A*.

9.2.3 Blends

Blending is a means of smoothly joining two or more objects. Figure 9.14 illustrates a blending of a sphere and a cylinder using a surface patch that joins

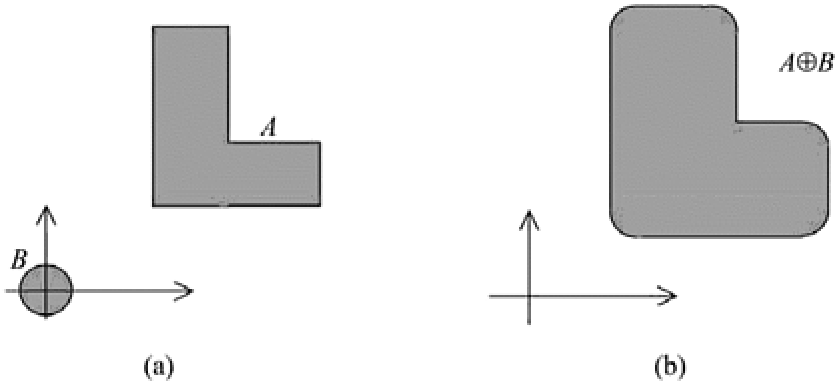


FIGURE 9.12 Minkowski addition of a two-dimensional region with a circular disk.

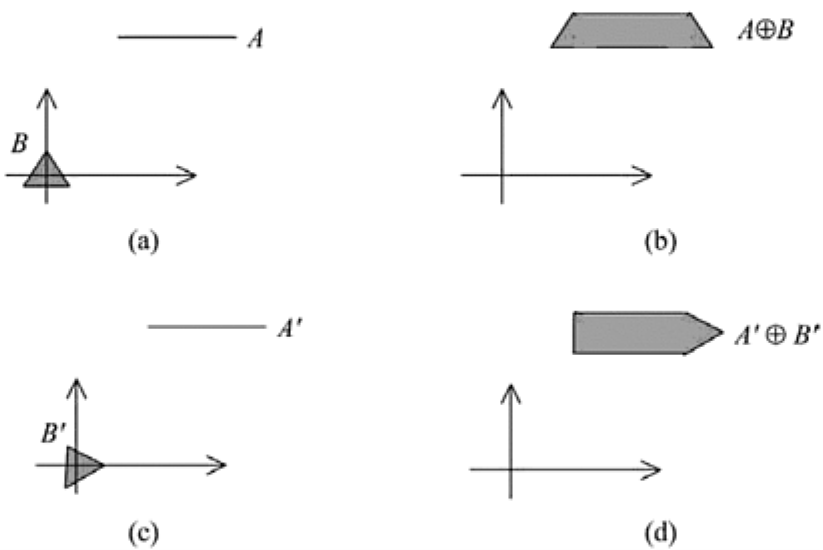


FIGURE 9.13 Relative orientation dependence of Minkowski additions. *A* and *A*′ are congruent, and so are *B* and *B*′. But the relative orientation of *A* and *B* is different from the relative orientation of *A*′ and *B*′.

these two surfaces smoothly. Such blending can be achieved in many ways. Different CAD systems choose different ways to achieve blending.

Rounding and filleting are two of the most popular blend operations. Figure 9.15 shows constant-radius rounding and filleting using a two-dimensional example. Note that rounding and filleting sharp corners essentially involve smooth blending of the edges incident at those corners, as

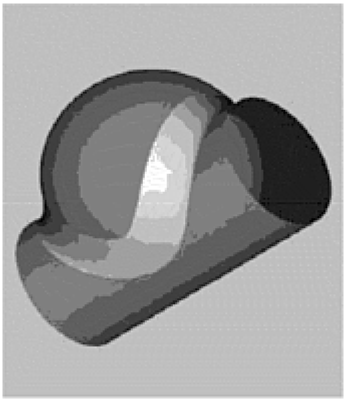


FIGURE 9.14 Smooth blending of a sphere and a cylinder.

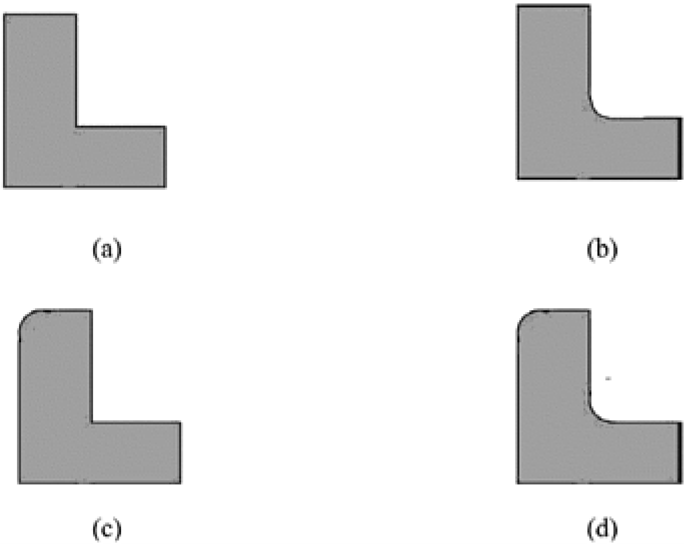


FIGURE 9.15 (a) A two-dimensional region with sharp corners. (b) Filleting the reentrant (nonconvex) corner. (c) Rounding a convex corner. (d) Applying both rounding and filleting.

shown in this figure. In three-dimensional cases, rounding and filleting of edges and vertices are essentially smooth blending of faces that are incident at these edges and vertices. If only constant-radius rounding and filleting are involved, then the radius is the only additional dimension (or parameter) involved in such operations.

A simple theory of constant-radius rounding and filleting of solids can be developed using Minkowski sums and certain morphological operations involving spherical balls. We have already seen Minkowski addition in Definition 9.1. Using this addition, Minkowski subtraction is defined as follows.

Definition 9.2: Minkowski Subtraction $A \ominus B = \overline{\overline{A} \oplus B}$

In this definition, overbar indicates complementation. Figure 9.16 illustrates Minkowski subtraction for two simple point-sets. Comparison of Figures 9.12 and 9.16 shows that Minkowski addition results in an expanded (dilated) object whereas Minkowski subtraction results in a shrunken (eroded) object.

Minkowski addition and subtraction are all we need to define the following *morphological operations*. In all these morphological operations, we assume that *B* indicates a spherical ball. (In two-dimensional applications, *B* will be considered a circular disk.)

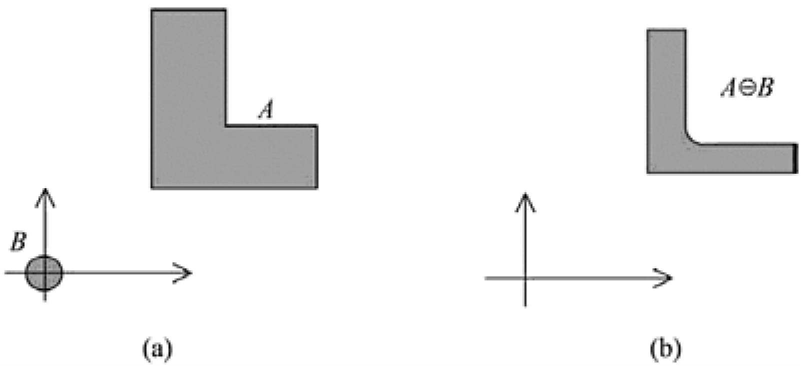


FIGURE 9.16 Minkowski subtraction of a two-dimensional region with a circular disk.

Definition 9.3: Dilation $D(A, B) = A \oplus B$
Definition 9.4: Erosion $E(A, B) = A \ominus B$
Definition 9.5: Opening $O(A, B)=D(E(A, B), B)$.
Definition 9.6: Closing $C(A, B)=E(D(A, B), B)$.

Applying the morphological operation of *opening* rounds all corners of *A* by a constant radius given by the radius of *B*. See Figure 9.17 for an example of applying erosion followed by dilation to achieve opening. Applying the *closing* operation fillets all corners of *A* by the same radius. Figure 9.18 shows the same example, but applying dilation first and erosion next, for closing. Roughly speaking, opening “knocks down” peaks and closing “fills up” valleys. If both rounding and filleting are desired, then applying closing followed by opening, that is, $O(C(A, B), B)$, will yield the result.

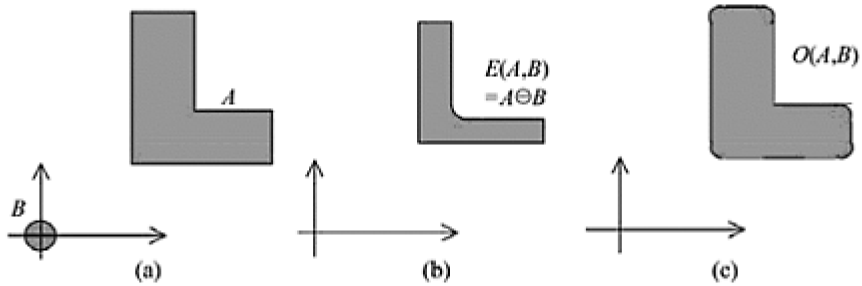


FIGURE 9.17 Opening *A* with *B* rounds all sharp (convex) corners. It doesn't affect reentrant (nonconvex) corners.

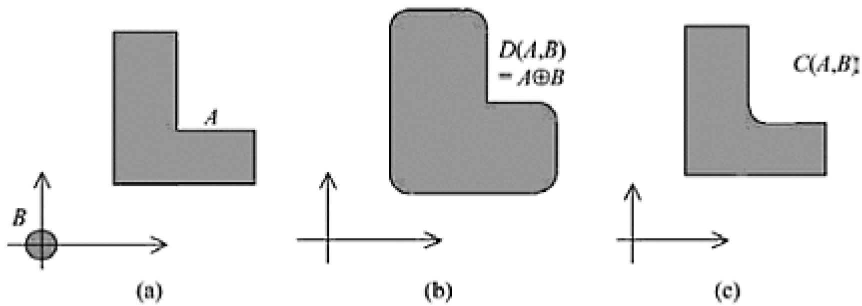


FIGURE 9.18 Closing *A* with *B* fillets the reentrant (nonconvex) corner. It doesn't affect convex corners.

So the smooth blending of a sphere and a cylinder shown in Figure 9.14 can be achieved by a simple closing operation: Take the union of the solid sphere and solid (finite) cylinder and to the result apply the closing operation with a constant-radius spherical ball. The only additional dimension (or parameter) involved in the closing operation is the radius of filleting (that is, the radius of *B*).

Morphological operations are global in the sense that they apply to the whole of point-set *A*. Often, we may want to apply rounding or filleting only to a portion of a solid object. In some cases this may be accomplished by selecting a subsolid in a procedural definition of the solid and applying the morphological operation to that subsolid. See Figure 9.19 for an example of this operation. Let's assume that only the top vertex of the triangular region in Figure 9.19(a) should be rounded so that we obtain the region shown in Figure 9.19(b). This cannot be accomplished by applying an opening operation for the whole region. Instead, we first represent the triangular region as an intersection of three half-planes, as shown in Figure 9.19(c). Then we round $h_1 \cap h_2$ by applying the opening operation $O((h_1 \cap h_2), B)$, and intersect it with h_3 to get the desired result.

9.2.4 Summary

A procedurally defined solid is constructed using a series of geometric operations on several geometric objects. Therefore, it is supposed to possess a *construction history*. Formally, construction history is a hierarchy of construction operations, such as Booleans, sweeps, and blends, performed on well-defined geometric objects. Due to the hierarchical nature, a construction history of a solid can contain other construction histories. Intrinsic and relational dimensions (and parameters) can be specified as part of the construction history.

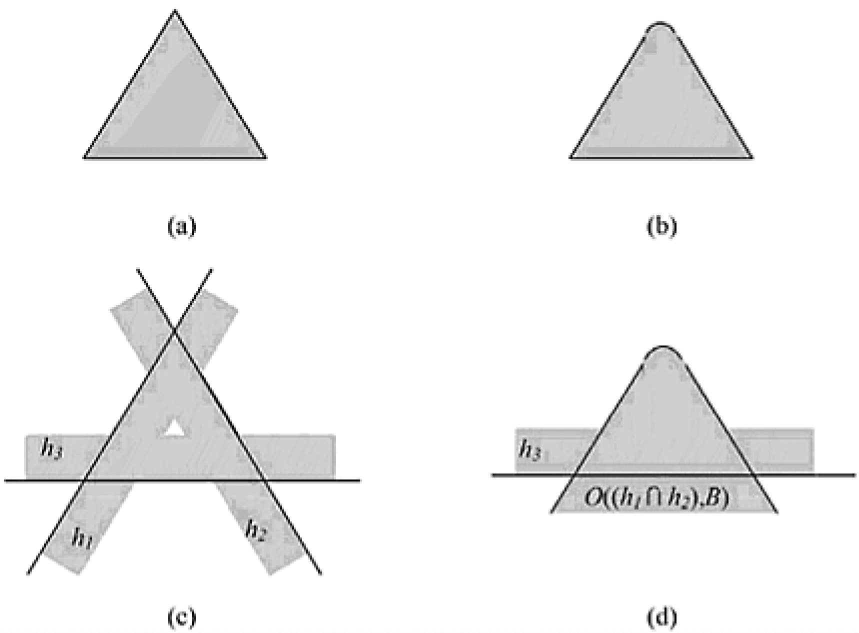


FIGURE 9.19 Selective rounding.

A procedurally defined solid is generally considered to be attractive because it can be modified by just editing its construction history and rerunning it. As we saw earlier, this is not the case with explicit representation of solids. However, editing a construction history is not without its perils, as even the following simple case illustrates. Consider a construction history embodied in the CSG tree of Figure 9.6(a) and explained in detail in Example 9.1. Let's assume that we decide to edit this history by just locally changing the relative positioning of H_1 and H_2, as shown in Figure 9.20. Where would the half-space H_3 be placed relative to the intersection of H_1 and H_2 shown in Figure 9.20(b)? Since the edit has changed the symmetry classification of the tuple (H_1, H_2) from planar to prismatic, the original relative positioning of H_3 with respect to (H_1, H_2) is no longer valid, and this also needs to be changed. So the effect of a local edit can propagate quite far down the construction history and render the original dimensional and parametric specifications invalid. It is up to the creator of the model to ensure that the edited version of a construction history remains consistent.

An explicit representation, such as a boundary representation, can be derived for an instance of a procedurally defined solid. In fact, most modern CAD systems keep a dual representation of a solid—one procedural and the other an explicit instance.

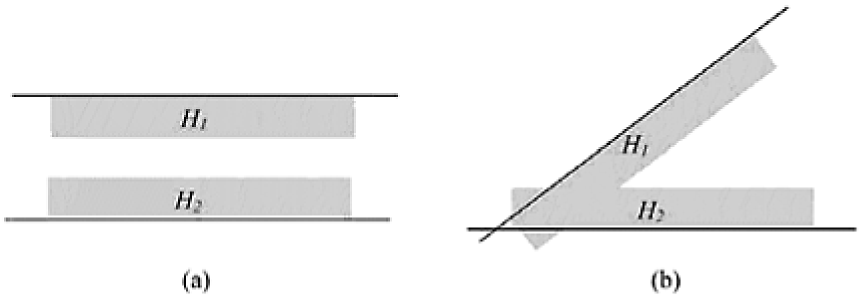


FIGURE 9.20 Changing the relative positioning of two half-spaces from (a) to (b) in a construction history embodied in the CSG tree of Figure 9.6 (a).

9.3 DIMENSIONING FEATURES

Features are geometric objects whose complexity lies somewhere between elementary objects, such as curves and surfaces discussed in Chapters 3 and 4, and solids, which have been studied thus far in this chapter. The four geometric objects shown in Figure 9.1 can be considered as features.

Features are often parameterized. Thus we can have parameterized macros for constructing them. If certain features (for example, blind threaded holes) occur frequently in a design, then it is convenient to have such parameterized macros for these features so that they can be quickly instantiated and incorporated in the geometric model of a product.

Features can be named using the vocabulary of a particular trade (e.g., “I-beam” in structural engineering, “dovetail slot” in mechanical engineering, “via hole” in electrical engineering). This enhances the user friendliness of CAD systems. More importantly, these names may have semantic implication in downstream applications. For example, a manufacturing application may want to know how many “I-beams” or “via holes” are present in a particular design.

For each feature, its constraints, dimensions, or parameters are predefined along with its explicit or procedural representation. A typical feature is a relatively simple object. So if it contains some dimensional constraints, then they can be preanalyzed and the solution can be stored for use in a dimensional scheme. Figure 9.21 shows two examples of parameterized features. In each case certain constraints are implicit. In Figure 9.21(a) the cylindrical and conical surfaces are coaxial. In Figure 9.21(b), tangency is implied in several places.

The features shown in Figure 9.21 could be created in many ways. To create the blind hole with conical bottom in Figure 9.21(a), one can start

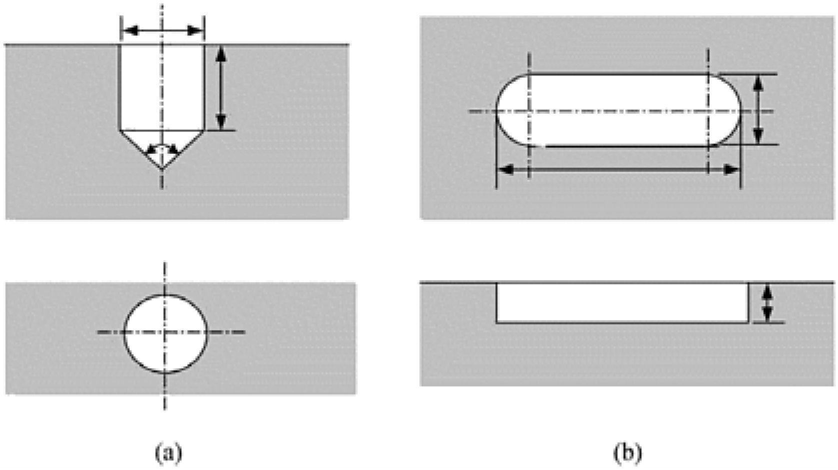


FIGURE 9.21 Examples of parameterized features. (a) A blind hole with a conical bottom. (b) An elongated slot.

with a planar sketch, as shown in the top projected view. When proper constraints are imposed, it becomes a well-dimensioned (or parameterized) sketch. The region outside of this simple closed curve can be limited by intersection with a half-plane, and the result can be swept rotationally about the axis to give us the desired feature. Similarly, to create the elongated slot in Figure 9.21(b), one can start with a planar sketch, shown in the top view. With proper constraints, it defines a planar region. (The reader is encouraged to list these constraints explicitly.) It can then be swept translationally in a direction perpendicular to the plane of the sketch by a finite amount, and the resulting solid can be subtracted from a half-space to produce the desired feature.

These examples illustrate how features can be predefined by a combination of dimensional constraints described in Chapter 8 and construction procedures discussed in Section 9.2. In a typical application, such a feature can be selected by its name and instantiated by specifying numerical values for its parameters (that is, specifying its dimensions). It can then be used in a construction procedure to build a solid.

9.4 SUMMARY

Dimensions and constraints are not sufficient to define a solid. They should be a part of a solid representation. Ideally, a procedural representation of a solid is preferable because its hierarchical representation can also be used for a dimensioning (or parameterizing) scheme. Many modern CAD systems use a hybrid of explicit and procedural representation schemes. Typically, a subsolid in a procedural representation of a solid may have an explicit representation. Even if a solid is defined procedurally, its boundary representation for an instance of the solid is also computed and stored along with it. Such dual representations are common in modern CAD systems.

Features can be used as parameterized macros in constructing a solid model. Dimensions, constraints, and parameters for the features are more easily defined than for a general solid. This explains their popularity in modern CAD systems.

9.5 EXERCISES

1. Prove Theorem 9.2. Why do you think that only the orientation component of relative positioning of *A* and *B* matters?
2. Use morphological operations to apply the selective rounding and filleting shown in Figure 9.15(d).
3. Minkowski sums and morphological operations can be regularized (see Appendix 4 for regularization of sets). Examine the need for such regularization and how they affect sweep operations.

9.6 NOTES AND REFERENCES

Mantyla (1988) describes boundary representations of polyhedral solids. Requicha and Voelcker (1985) describe Boolean operations in solid modeling, with particular attention to boundary evaluation. Hoffmann (1989) gives a good introduction to geometric and solid modeling. Shah and Mantyla (1995) deal with features and their parameterization in CAD/CAM.

Matheron (1975) and Dougherty (1992) are good references for morphological operations. Shapiro and Vossler (1995) describe problems that arise when solid models are parameterized.

Appendix 1
Matrices

Matrices are used in many places in this book, where some knowledge of matrices is assumed. In this appendix we give a brief review of matrices of real numbers. In the past, the tedium of computing determinant, rank, eigenvalues, and so on of matrices was a major impediment to the liberal use of matrices in engineering applications. Lately, this tedium has been considerably lessened by the wide availability of high-quality software such as MATLAB®. Therefore we can focus on the basic properties and applications of matrices and leave detailed computation to such software.

A1.1 SCALARS, VECTORS, AND MATRICES

A *scalar* is a real number. The set of all real numbers is denoted by $\mathbb{R}$. So the fact that a is a scalar is expressed compactly by $a = a_1\hat{i} + a_2\hat{j} + a_3\hat{k}$ A vector of real numbers is an ordered collection of scalars. For example, $a=[a_1\ a_2\ a_3]$ is a vector of three real numbers, and we will say that a is a three-dimensional vector of real numbers and express this compactly by $\hat{i}, \hat{j},$ This vector can also be indicated by the familiar notation $a = a_1\hat{i} + a_2\hat{j} + a_3\hat{k}$, where $\hat{i}, \hat{j}$ and $\hat{k}$ are the unit (basis) vectors. The components of a vector have a single subscript. The ith component of a general n-dimensional vector a is denoted by a_i, and if all

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \quad \text{or} \quad \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

then we say that the vector

$$\begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix} \quad \text{or} \quad \begin{pmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{pmatrix} \quad \text{or} \quad \begin{Bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{Bmatrix}$$

A *tensor* is a generalization of a vector. This generalization is best achieved using the subscript notation. Thus $[a_i]$ will denote a first-order tensor, $[a_{ij}]$ will denote a second-order tensor, $[a_{ijk}]$ will denote a third-order tensor, and so on. So vectors are first-order tensors. By convention, scalars are zeroth-order tensors. Matrices are second-order tensors. A component, or element, of a matrix A will be denoted by a_{ij}. The subscript integer j may vary from 1 to m, and the subscript integer j may vary from 1 to n. If all $a_{ij} \in \mathbb{R}$ then we say compactly that the matrix $A \in \mathbb{R}^{m\times n}$

Matrix A itself can be written out explicitly in rows and columns as

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \quad \text{or} \quad \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

Each row in this matrix can be viewed as a row vector. Similarly, each column can be viewed as a column vector. Therefore, the ith row vector of A can be written as

$[a_{i1}\ a_{i2} \ldots a_{in}]$ or $(a_{i1}\ a_{i2}\ a_{in})$

and the jth column vector of A can be written as

$$\begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix} \quad \text{or} \quad \begin{pmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{pmatrix} \quad \text{or} \quad \begin{Bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{Bmatrix}$$

A zero matrix is one in which all elements are zeros. An element aii for which the row index and the column index are the same is called a *diagonal element* of A. There are min (m, n) diagonal elements in A. An element of A that is not a diagonal element is called an *off-diagonal element*.

A1.2 BASIC MATRIX OPERATIONS

The negative of a matrix is obtained by taking the negative of every element of the matrix. That is, $-A=[-a_{ij}]$. The transpose of a matrix, denoted A^T, is obtained by interchanging the rows and columns of A. Multiplication of a matrix A by a scalar s produces a matrix in which every element is a multiple of the corresponding element of A by s. That is, $sA=[sa_{ij}]$.

Example A1.1 Let

$$A = \begin{bmatrix} 2 & -5.5 & 8.1 \\ -4.6 & -22.3 & 9.4 \end{bmatrix}$$

Then, we have:

$$-A = \begin{bmatrix} -2 & 5.5 & -8.1 \\ 4.6 & 22.3 & -9.4 \end{bmatrix}, \qquad A^{\mathrm{T}} = \begin{bmatrix} 2 & -4.6 \\ -5.5 & -22.3 \\ 8.1 & 9.4 \end{bmatrix}$$

$$3A = \begin{bmatrix} 6 & -16.5 & 24.3 \\ -13.8 & -66.9 & 28.2 \end{bmatrix}$$

The operation of matrix addition, written as $C=A+B$, produces a matrix C with $cij=a_{ij}+b_{ij}$. Only matrices with the same number of rows and the same number of columns can be added. Matrix subtraction is defined similarly; that is, in $C=A-B$, $c_{ij}=a_{ij}-b_{ij}$.

Example A1.2 Let

$$A = \begin{bmatrix} 2 & -5.5 & 8.1 \\ -4.6 & -22.3 & 9.4 \end{bmatrix} \quad \text{and}$$

$$B = \begin{bmatrix} -5.2 & 12.2 & 2.5 \\ 6.6 & -17 & -4.3 \end{bmatrix}$$

Then we have

$$A + B = \begin{bmatrix} -3.2 & 6.7 & 10.6 \\ 2 & -39.3 & 5.1 \end{bmatrix} \quad \text{and}$$

$$A - B = \begin{bmatrix} 7.2 & -17.7 & 5.6 \\ -11.2 & -5.3 & 13.7 \end{bmatrix}$$

Matrix addition is commutative and associative. That is, $A+B=B+A$ and $(A+B)+C=A+(B+C)$. Note that the order is important in matrix subtraction. That is, $A-B$ need not be the same as $B-A$. It is easy to see that $(A+B)^{\mathrm{T}}=A^{\mathrm{T}}+B^{\mathrm{T}}$.

The product (or multiplication) of two matrices A and B is denoted AB and is defined only if the number of columns of A equals the number of rows of B. If

$$c_{ij} = \sum_{k=1}^{p} a_{ik} b_{kj}, \qquad i = 1, \ldots, m; j = 1, \ldots, n$$

and

$$A = \begin{bmatrix} 2 & -5.5 & 8.1 \\ -4.6 & -22.3 & 9.4 \end{bmatrix} \quad \text{and}$$

$$B = \begin{bmatrix} -5.2 & 12.2 & 2.5 \\ 6.6 & -17 & -4.3 \end{bmatrix}$$

then the product

$$A^{\mathrm{T}}B=\begin{bmatrix} -40.76 & 102.60 & 24.78 \\ -118.58 & 312.00 & 82.14 \\ 19.92 & -60.98 & -20.17 \end{bmatrix},$$

$$BA^{\mathrm{T}}=\begin{bmatrix} -57.25 & -224.64 \\ 71.87 & 308.32 \end{bmatrix}$$

is defined by its elements as

$$c_{ij}=\sum_{k=1}^{p} a_{ik}b_{kj}, \qquad i=1,\ldots,m; j=1,\ldots,n$$

An important observation here is that matrix multiplication is not commutative. That is, *AB* need not be the same as *BA,* even if both are well defined.

Example A1.3 As in Example A1.2, let

$$A=\begin{bmatrix} 2 & -5.5 & 8.1 \\ -4.6 & -22.3 & 9.4 \end{bmatrix} \qquad \text{and}$$

$$B=\begin{bmatrix} -5.2 & 12.2 & 2.5 \\ 6.6 & -17 & -4.3 \end{bmatrix}$$

The matrix product *AB* is not defined because the number of columns of *A* does not match the number of rows of *B*. However, A^TB and BA^T are well defined and are given by

$$A^{\mathrm{T}}B=\begin{bmatrix} -40.76 & 102.60 & 24.78 \\ -118.58 & 312.00 & 82.14 \\ 19.92 & -60.98 & -20.17 \end{bmatrix},$$

$$BA^{\mathrm{T}}=\begin{bmatrix} -57.25 & -224.64 \\ 71.87 & 308.32 \end{bmatrix}$$

Matrix multiplication is associative; that is, $(AB)C=A(BC)$. It is also distributive over addition. That is, $A(B+C)=AB+AC$. What is interesting is the reverse order in which transpose applies over multiplication; that is, $(AB)^{\mathrm{T}}=B^{\mathrm{T}}A^{\mathrm{T}}$.

A special case of matrix multiplication is the scalar (or inner) product of two vectors of the same dimension. If $A, B \in \mathbb{R}^{m\times 1}$ then $A^{\mathrm{T}}B=B^{\mathrm{T}}A$ is a scalar and is the scalar (or inner) product of the vectors *A* and *B*. If *A* and *B* are row vectors, that is, $A, B \in \mathbb{R}^{1\times n}$ then $AB^{\mathrm{T}}=BA^{\mathrm{T}}$ is the scalar (or inner) product.

A1.3 SQUARE MATRICES

In a square matrix, the number of rows equals the number of columns. A square matrix in which all diagonal elements are 1 and all off-diagonal elements are 0 is called an *identity* (or unit) *matrix*. It is often denoted by the letter I. If A and I are square matrices of the same size (that is, the same number of rows and the same number of columns), then $AI=IA=A$. A square matrix A is symmetric if $A=A^T$. The identity matrix is symmetric; so is a zero square matrix.

Example A1.4

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

is the $\mathbb{R}^{2\times 2}$ identity matrix. It is symmetric.

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

is the $\mathbb{R}^{3\times 3}$ identity matrix. It is also symmetric.

$$\begin{bmatrix} 3 & -4.5 & 6.1 \\ -4.5 & 0 & 5.3 \\ 6.1 & 5.3 & 1.2 \end{bmatrix}$$

is symmetric.

$$\begin{bmatrix} 3 & -4.5 & 6.1 \\ -0.5 & 0 & 5.3 \\ 2.1 & 3.3 & 1.2 \end{bmatrix}$$

is not symmetric.

Every square matrix has a determinant. The determinant of square matrix $A \in \mathbb{R}^{n\times n}$ is a scalar and is denoted by $|A|$ or det(A). It can be defined recursively as follows. First, we define the determinant of a 1×1 matrix to be equal to the scalar value of the sole element in that matrix. Next, for any square matrix A that has more than one row (or column) we call A_{ij} the *cofactor* of the element a_{ij} in A. This cofactor A_{ij} is a scalar and is defined to be $(-1)^{i+j}$ times the determinant of the $(n-1)\times(n-1)$ square matrix formed by deleting the ith row and the jth column of A. Finally, with these notations, the determinant of A can be obtained as

$$|A|=\det(A)=a_{11}A_{11}+a_{12}A_{12}+\ldots+a_{1n}A_{1n}$$

Example A1.5 Let's find the determinant of the following 2×2 square matrix using the preceding formula:

$$A=\begin{bmatrix} -57.25 & -224.64 \\ 71.87 & 308.32 \end{bmatrix}$$

We have $|A|=-57.25\times 308.32-224.64\times(-1)\times 71.87=-1.5064\times 10^3$. Now let's find the determinant of the following 3×3 square matrix

$$B=\begin{bmatrix} -40.76 & 102.60 & 24.78 \\ 18.58 & 30.00 & 82.14 \\ 19.92 & -60.98 & -20.17 \end{bmatrix}$$

It is given by

$$|B|=-40.76\times\begin{vmatrix} 30.00 & 82.14 \\ -60.98 & -20.17 \end{vmatrix}+102.60\times(-1)$$

$$\times\begin{vmatrix} 18.58 & 82.14 \\ 19.92 & -20.17 \end{vmatrix}$$

$$+24.78\times\begin{vmatrix} 18.58 & 30.00 \\ 19.92 & -60.98 \end{vmatrix}$$

It turns out to be -1.6056×10^4.

As mentioned in the beginning, computing determinants of matrices of even modest size can be tedious and error prone. But luckily we can use reliable software to compute these quantities.

It can be shown that the determinant of the product of two square matrices is equal to the product of their determinants. That is, $|AB|=|A||B|$. This result can be extended to the product of several square matrices. It can also be shown that the determinant of a square matrix is equal to the determinant of its transpose, that is, $|A|=|A^T|$.

If the determinant of a square matrix is zero, then it is called a *singular* matrix; otherwise it is called a *nonsingular* matrix. A nonsingular square matrix A has an inverse matrix, denoted by A^{-1}, that is of the same size as A. It has the property that $AA^{-1}=A^{-1}A=I$. So a nonsingular matrix is also called an *invertible* matrix.

Example A1.6 For matrices A and B in Example A1.5,

$$A^{-1} = \begin{bmatrix} -0.2047 & -0.1491 \\ 0.0477 & 0.0380 \end{bmatrix} \quad \text{and}$$

$$B^{-1} = \begin{bmatrix} -0.2743 & -0.0348 & -0.4786 \\ -0.1252 & -0.0205 & -0.2372 \\ 0.1078 & 0.0275 & 0.1949 \end{bmatrix}$$

It can be shown that the inverse of the product of two nonsingular matrices is the product of their inverses, taken in the reverse order. That is, $(AB)^{-1}=(B^{-1})(A^{-1})$. This result also can be extended to the product of several nonsingular matrices.

A1.4 RANK

Let $\{v_1, v_2,..., v_n\}$ be a set of n vector of the same dimension. A linear combination of these vectors can be expressed as $\alpha_1v_1+\alpha_2v_2+\ldots+ \alpha_nv_n$, where α_i are scalars. These n vectors are said to be linearly dependent if $\alpha_1v_1+\alpha_2v_2+\ldots+\alpha_nv_n=0$ (that is, the zero vector), for $\alpha_i\neq 0$ for at least one i. Otherwise, the vectors are linearly independent.

The column rank of a matrix is the maximum number of linearly independent columns in that matrix. Similarly, the row rank of a matrix is the maximum number of linearly independent rows in that matrix. It can be shown that for any matrix (not necessarily square), the column and row ranks are the same, and it is called the *rank* of the matrix. Simply put, the rank of a matrix is the dimension of the space spanned by its rows (or columns); this geometric interpretation will be illustrated in the following examples. The rank of an $m\times n$ matrix cannot exceed min(m, n). For $m\neq 0$, $n\neq 0$, only the zero matrix has a rank of 0.

Example A1.7 Consider the following matrix from Example 4.1.

$$\begin{bmatrix} 0 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{bmatrix}$$

Its rank is 3. It is a nonsingular matrix; that is, it has a nonzero determinant. None of its rows (or columns) can be expressed as a linear combination of the other two rows (or columns). If we take the three row vectors as basis vectors, then any three-dimensional vector can be expressed as a linear combination of these three basis vectors. The same is true for the three column vectors. This is what we mean by the row (or column) vectors spanning a three-dimensional space.

Example A1.8 Next, consider the following matrix from Example 4.2:

$$\begin{bmatrix} 1 & 1 & -1 \\ 1 & 2 & 0 \\ -1 & 0 & 2 \end{bmatrix}$$

Its rank is 2. It is a singular matrix because its determinant is zero. Its third row can be obtained by subtracting twice the first row from the second row. So the rows are not linearly independent. The three row (or column) vectors span a two-dimensional space. Geometrically, this means that the origin and the three points whose coordinates are obtained from the rows (or columns) of this matrix are coplanar.

Example A1.9 Finally, consider the following matrix from Example 4.3:

$$\begin{bmatrix} 9 & -3 & 3 \\ -3 & 1 & -1 \\ 3 & -1 & 1 \end{bmatrix}$$

Its rank is 1. It is a singular matrix. It turns out that its first row is three times its third row and that its second row is just the negative of its third row. The row (or column) vectors span only a one-dimensional space. Geometrically, this means that the origin and the three points whose coordinates are obtained from the rows (or columns) of this matrix are collinear.

It can be shown that an $n \times n$ matrix A is of rank n if and only if it is not singular, that is, if $|A| \neq 0$. Determining the rank of an arbitrary matrix is a difficult task. As remarked earlier, we are lucky to have widely available software to do this for us.

A1.5 TRANSFORMATIONS AND HOMOGENEOUS COORDINATES

Multiplying a matrix by a vector of the right size yields another vector. This gives us a simple means of transforming vectors. Since points can be represented by vectors of their coordinates, we see that points and point-sets can also be transformed by matrices. This geometric transformation is expressed compactly by the matrix notation $x'=Ax$, where a vector x of coordinates is transformed to a vector of coordinates x' by the matrix A.

There is an even simpler transformation of vectors obtained by vector addition, because addition of two vectors of the same size yields another vector. We can then combine these two types of transformations in the notation $x'=Ax+x_0$. This can be reduced to a single matrix multiplication operation if we extend the size of the vectors by 1, as in

$$\begin{Bmatrix} x' \\ 1 \end{Bmatrix} = \begin{bmatrix} A & x_0 \\ 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ 1 \end{Bmatrix}$$

The extended vector, whose last element always has a value of unity, is supposed to correspond to the homogeneous coordinates of a point. Matrix transformations of vectors of both point coordinates and its homogeneous coordinates are used extensively in this book.

A1.6 ORTHOGONAL MATRICES

Orthogonal matrices are square matrices with the property that $A^TA=I$; that is, A^T is the inverse of A. From this it follows that $AA^T=I$. Hence the rows of an orthogonal matrix are orthonormal. That is, the inner product (also called dot product) of a row by itself is unity, but the inner product of any two distinct rows is zero. The columns of an orthogonal matrix are also orthonormal.

Recall that $|A^T|=|A|$ for any square matrix A and $|I|=1$. Therefore, if A is orthogonal, we have $|A^TA|=|I|=1$ and $|A^TA|=|A^T||A|=|A|^2$. Hence, $|A|^2=1$, leading to the important result that $|A|$ can only be ±1 if A is orthogonal.

The inner product of a vector with itself is always a non-negative scalar, and its positive square root is called the Euclidean norm, or 2-norm, of the vector. If x is a column vector, then the positive square root of x^Tx becomes the 2-norm of x. Geometrically we can interpret this value as the length of the vector x. That is, it is the Euclidean distance between the origin and the point whose coordinates are represented by the vector x.

Orthogonal transformation is the only matrix transformation that preserves the 2-norm of a vector. This is because $(Ax)^T(Ax)=x^TA^TAx=x^Tx$. We interpret this result geometrically by saying that only orthogonal transformations preserve distances. As described in Chapter 2, an orthogonal transformation is a rotation if $|A|=+1$.

A1.7 EIGENVALUES AND EIGENVECTORS

Let A be an $n\times n$ square matrix and x an $n\times 1$ vector. The matrix multiplication Ax transforms an arbitrary vector x into another $n\times 1$ vector, x'. In general, x' will be different from x, but sometimes x' can turn out to be parallel to x, that is, x' is just a scalar multiple of x. We capture this condition by the equation $Ax=\lambda x$, where λ is a scalar. Given a matrix A, we are interested in finding out those vectors that are transformed by A into parallel vectors—these are called *eigenvectors* of A. In addition, we are also interested in finding the corresponding values for the scalar multiplier λ—these are called *eigenvalues* of A. Eigenvectors are known only within a multiplicative constant; that is, if x^* is an eigenvector, then any constant multiple of x^* is also an eigenvector for the same eigenvalue.

The problem of finding the eigenvalues of A can be posed as a polynomial root-finding problem by setting $|A-\lambda I|=0$. There are better ways to find not only the eigenvalues, but also the associated eigenvectors, and these are implemented in widely available software.

Example A1.10 For the matrix in Example A1.7, the eigenvalues are −0.5, −0.5, and 1.0. The corresponding eigenvectors are

$$\begin{Bmatrix} -0.5870 \\ 0.7850 \\ -0.1980 \end{Bmatrix}, \quad \begin{Bmatrix} 0.5676 \\ 0.2246 \\ -0.7921 \end{Bmatrix}, \quad \text{and} \quad \begin{Bmatrix} 0.5774 \\ 0.5774 \\ 0.5774 \end{Bmatrix}$$

Note how identical eigenvalues can have distinct eigenvectors. Also recall that the matrix had a rank of 3.

Example A1.11 Next, for the matrix in Example A1.8, the eigenvalues are 3, 2, and 0. The corresponding eigenvectors are

$$\begin{Bmatrix} -0.5774 \\ -0.5774 \\ 0.5774 \end{Bmatrix}, \quad \begin{Bmatrix} 0.0000 \\ -0.7071 \\ -0.7071 \end{Bmatrix}, \quad \text{and} \quad \begin{Bmatrix} -0.8165 \\ 0.4082 \\ -0.4082 \end{Bmatrix}$$

The matrix has a rank of 2. So one of the eigenvalues is zero. Note that the corresponding eigenvector is nonzero.

Example A1.12 Finally, for the matrix in Example A1.9, the eigenvalues are 0, 0, and 11. The corresponding eigenvectors are

$$\begin{Bmatrix} 0.4253 \\ 0.6891 \\ -0.5868 \end{Bmatrix}, \quad \begin{Bmatrix} 0.0308 \\ -0.6590 \\ -0.7515 \end{Bmatrix}, \quad \text{and} \quad \begin{Bmatrix} 0.9045 \\ -0.3015 \\ 0.3015 \end{Bmatrix}$$

The matrix has a rank of 1. So two of the eigenvalues are zero.

If A is a square, symmetric matrix of real numbers, then it can be shown that all of its eigenvalues are real and its eigenvectors are orthogonal. This property is quite useful in the study of conic curves and quadric surfaces. Matrices considered in Examples A1.10 through A1.12 are of this type. They arise from the quadratic form of the governing equations of conics and quadrics.

It can be shown that the sum of the diagonal elements of any square matrix A (called the *trace* of A) is equal to the sum of the eigenvalues of A. Also, its can be shown that the product of eigenvalues of A is equal to the determinant of A.

A1.8 EXERCISES

1. Use the recursive formula to find the determinant of matrices in Eqs. (2.5) and (2.6).
2. Show that for a nonsingular matrix A,

$$|A^{-1}| = \frac{1}{|A|}$$

3. Prove that the product of two invertible real matrices is invertible.

4. Prove that the product of two orthogonal matrices is orthogonal.

A1.9 NOTES AND REFERENCES

A basic and elegant introduction to matrices and, in particular, eigenproblems is provided by Lanczos (1957). Bellman (1970) gives a good introduction to matrix analysis. Golub and Van Loan (1996) discuss matrix computations quite comprehensively.

Appendix 2
Groups

Study of geometry is incomplete without some understanding of groups. Chapter 6 covered some aspects of groups needed for our study of symmetry. In this appendix we give an independent, but brief, introduction to groups. We will, however, refer to Chapter 6 for several examples.

A2.1 AXIOMS

An abstract group is a collection G of elements with a binary operation, called the *group operation,* that will be denoted by •. This strong connection between the collection of elements and the group operation is captured by denoting the entire group as (G, •). The group must satisfy the following four axioms.

Axiom G1: Closure *For any* $g_1, g_2 \in G$, $g_1 \bullet g_2 \in G$.

Axiom G2: Associativity $g_1 \bullet (g_2 \bullet g_3) = (g_1 \bullet g_2) \bullet g_3$.

Axiom G3: Identity *There exists an identity element* $e \in G$ *such that* $g \bullet e = e \bullet g = g$ *for all* $g_2 \in G$.

Axiom G4: Inverse *For each* $g \in G$ *there is an inverse element* $g^{-1} \in G$ *G such that* $g \bullet g^{-l} = g^{-l} \bullet g = e$.

We can show that the identity element is unique. To see this, assume that there are two identity elements e and f in the group (G, •). Then, by definition, for any $a \in G$ we have $a \bullet e = e \bullet a = a$ and $a \bullet f = f \bullet a = a$. Therefore, by substituting f for a first and e for a next, we have $f \bullet e = e \bullet f$ and $e \bullet f = f \bullet e = e$. This shows that $e = f$, proving the uniqueness of the identity element.

We can also show that there is a unique inverse for each element in the group (G, •). To show this, for any $a \in G$ we define a left inverse l and a right inverse r so that $l \bullet a = e$ and $a \bullet r = e$. Therefore, $l = l \bullet (a \bullet r) = (l \bullet a) \bullet r = r$. This shows that the left and right inverses are the same and that the inverse is unique.

The group operation need not be commutative. That is, $g_1 \bullet g_2$ need not be equal to $g_2 \bullet g_1$. If it is, then the group is called a *commutative,* or *Abelian,* group.

Example A2.1 ($\mathbb{R}$, +) is a group, where $\mathbb{R}$ is the set of all real numbers and + is the usual arithmetic addition operation. Closure and associativity are easy to see. Zero is the identity element. Negation gives us the inverse. ($\mathbb{R}$, +) is a commutative group, because + is a commutative operation.

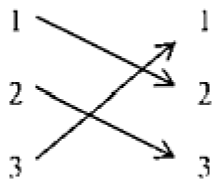
Similarly, ($\mathbb{Z}$, +) is a group, where $\mathbb{Z}$ denotes the set of all integers (including positive integers, negative integers, and zero). ($\mathbb{Z}$, +) is also a commutative group.

Example A2.2 ($\mathbb{R}\backslash 0$, ×) is a group, where $\mathbb{R}\backslash 0$ is the set of all real numbers excluding zero and × is the usual arithmetic multiplication operation. Closure and associativity are easy to see. Unity, that is, 1, is the identity element. The usual real inverse gives us the inverse. The last statement shows why it was necessary to exclude 0 from the set of numbers in the group. ($\mathbb{R}\backslash 0$, ×) is also a commutative group.

($\mathbb{Z}\backslash 0$, ×) is not a group.

Sometimes, certain "operations" themselves form a group under a group operation that tells us how to compose these "operations." Such groups are studied with great interest in geometry. Let's start with a simple exercise in permutation operation.

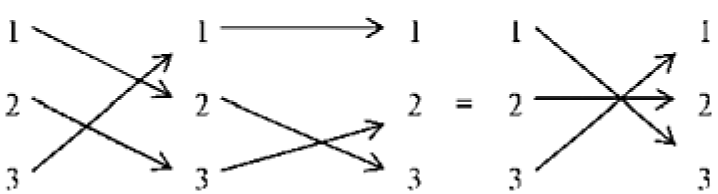
Example A2.3 Consider permutations of the three numbers (1, 2, 3). A particular permutation, represented by (2, 3, 1), can be shown pictorially as the following mapping.



We will represent this mapping, or operation, more compactly by

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{pmatrix}$$

There are only ${}_3P_3$=6 such permutations, and all six operations can be given symbolic names, α_1 through α_6, as enumerated in the first two columns of Table A2.1. If we take two operations, say, α_2 and α_4, then their *composition* $\alpha_2 \bullet \alpha_4$ can be depicted pictorially as follows:

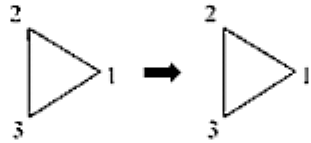
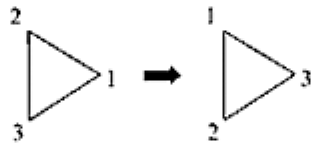
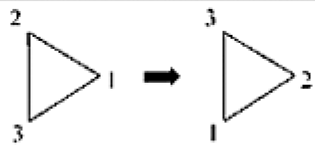
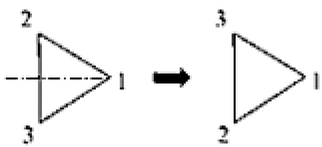
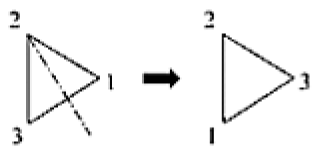
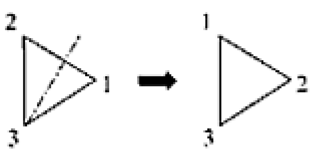


So the composition of α_2 and α_4, taken in that order, results in α_5. Any number of such compositions can be chained together to yield a permutation.

With these preliminaries, we can see that ($\{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6\}$, •) is a group. This is the permutation group of order 3 and is usually denoted by S_3. The composition table (also called Cayley table or, more loosely, multiplication table) for elements of this group is shown in Table A2.2. Clearly, the identity element is α_1. The table helps us to easily verify that the closure, associativity, and inverse axioms are satisfied by this group. S_3 is not a commutative group, because Table A2.2 is not symmetric.

We will now turn to an example from geometry.

TABLE A2.1 Permutation Group S3 and Symmetry Group of Transformations of an Equilateral Triangle

	Permutations	Transformations	
α_1	$\begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \end{pmatrix}$		No motion. $m_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
α_2	$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{pmatrix}$		CCW rotation by 120°. $m_2 = \begin{bmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} & -\frac{1}{2} \end{bmatrix}$
α_3	$\begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix}$		CCW rotation by 240°. $m_3 = \begin{bmatrix} -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{\sqrt{3}}{2} & -\frac{1}{2} \end{bmatrix}$
α_4	$\begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 2 \end{pmatrix}$		Reflection about axis through 1. $m_4 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
α_5	$\begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{pmatrix}$		Reflection about axis through 2. $m_5 = \begin{bmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} \\ -\frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix}$
α_6	$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix}$		Reflection about axis through 3. $m_6 = \begin{bmatrix} -\frac{1}{2} & \frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix}$

These two groups are isomorphic.

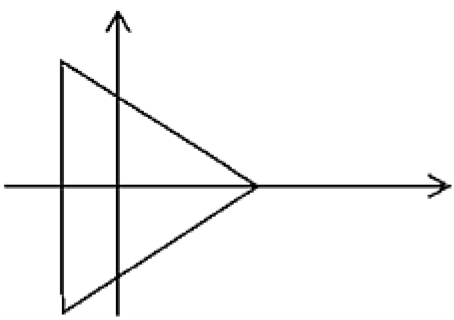
Example A2.4 Consider the set of six transformations shown in the last two columns of Table A2.1. Each of the transformations maps an equilateral triangle to itself by rotation or reflection. (The vertices of the triangle are labeled 1, 2, and 3 to establish a connection to the permutation group in Example A2.3.) The

TABLE A2.2 Composition (Cayley, or Multiplication) Table for Elements of the Permutation Group S_3.

	α_1	α_2	α_3	α_4	α_5	α_6
α_1	α_1	α_2	α_3	α_4	α_5	α_6
α_2	α_2	α_3	α_1	α_5	α_6	α_4
α_3	α_3	α_1	α_2	α_6	α_4	α_5
α_4	α_4	α_6	α_5	α_1	α_3	α_2
α_5	α_5	α_4	α_6	α_2	α_1	α_3
α_6	α_6	α_5	α_4	α_3	α_2	α_1

Each interior cell is the result of $\alpha_i \bullet \alpha_j$, where α_j is the row element and α_j is the column element.

transformation matrices shown in the last column of Table A2.1 should be interpreted as those applied with respect to the *x*- and *y*-coordinates in the following diagram:



These transformations can be composed by applying the matrix multiplication operation, which may be denoted by ×. We can see that $(\{m_1, m_2, m_3, m_4, m_5, m_6\}, \times)$ is a group. (Prove it.) An important fact we should remember is that the matrix multiplication $m_1 \times m_2$ stands for a composition in which the transformation m_2 is applied first and then followed by the application of the transformation m_1.

Some groups may contain subgroups, which are groups by themselves. More formally, if $(G, \bullet)$ is a group, H is a subset of G, and $(H, \bullet)$ satisfies all the four group axioms, then $(H, \bullet)$ is a subgroup of $(G, \bullet)$. In Example A2.3, $(\{\alpha_1, \alpha_2, \alpha_3\}, \bullet)$ can be shown to be a subgroup of S_3. (Prove it.) This subgroup is usually denoted by A_3 and is called the alternating group of order 3. A_3 is a commutative group.

A2.2 GROUP-ISMS

When are two groups equivalent? Intuitively, if we take the same group and give each of its elements two different symbols, we should not claim to have generated two different groups. This notion can be generalized and captured more formally by the notion of *isomorphism*. By isomorphism between two groups $(G, \bullet)$ and $(G', \circ)$ is meant a one-to-one correspondence $g \leftrightarrow g'$ (where $g \in G$ and $g' \in G'$) between their elements that preserves the group multiplication structure; that is, if $g_i \leftrightarrow g_i'$ and $g_j \leftrightarrow g_j'$, then $g_i \bullet g_j \leftrightarrow g_i' \circ g_j'$. For group theoretic purposes, isomorphic groups are equivalent.

Example A2.5 Consider the set A of real numbers under addition, which was seen to form a group in Example A2.1. Similarly, the set of all positive real numbers B under multiplication form a group. These two groups can be seen to be isomorphic by the exponential relationship $b=e^a$, where $a \in A$ and $b \in B$

Example A2.6 In Table A2.1, the permutation group $(\{\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6\}, \bullet)$ explained in Example A2.3 and the transformation group $(\{m_1, m_2, m_3, m_4, m_5, m_6\}, \times)$ explained in Example A2.4 are isomorphic. Keep in mind the fact that $\alpha_i \bullet a_j$ corresponds to $m_j \times m_i$ due to the order in which matrix multiplication is carried out.

Example A2.7 The group of rotations about the z-axis, represented by the matrices

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and the group of rotations about the x-axis, represented by the matrices

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $\phi \in \mathbb{R}$, are isomorphic. (Prove it.)

Any isomorphism of a group onto itself is called a group *automorphism*.

Example A2.8 The one-to-one correspondence $x \leftrightarrow x^3$, where $x,\ x^3 \in \mathbb{R}$, maps the group $(\mathbb{R}\backslash 0, \times)$ onto itself and is an isomorphism because $(x_i \times x_j)^3 = x_i^3 \times x_j^3$. Therefore, it is a group automorphism.

Isomorphism between two groups is a relationship that obeys some very simple, but important, properties. For example, the following can be shown:

1. *Reflexive:* The group G is isomorphic to itself.
2. *Symmetric:* If G_1 is isomorphic to G_2, then G_2 is isomorphic to G_1.
3. *Transitive:* If G_1 is isomorphic to G_2 and G_2 is isomorphic to G_3, then G_1 is isomorphic to G_3.

Therefore, isomorphism is an *equivalence* relation. This formalizes the notion expressed earlier that we treat two groups that are isomorphic as equivalent for group theoretic purposes. This equivalence is exploited in the classification results of Chapter 6.

A2.3 NORMAL SUBGROUPS

Of all subgroups of a given group, there are subgroups called *normal* subgroups that have some interesting properties. Let $(N, \bullet)$ be a subgroup of $(G, \bullet)$, and let N be written as the set $\{n_1, n_2, n_3, \ldots\}$. If $g \in G$, then we will write $gN=\{g \bullet n_1, g \bullet n_2, g \bullet n_3, \ldots\}$, called a *left coset* of N, and $Ng=\{n_1 \bullet g, n_2 \bullet g, n_3 \bullet g, \ldots\}$, called a *right coset* of N. The element g will be called the *coset leader* in these cosets.

The subgroup N is called a normal subgroup of G if for any element $g \in G$, $gN=Ng$, that is, if the left and right cosets are the same in the set theoretic sense. Note that we are not requiring $g \bullet n1$ to be equal to $n_1 \bullet g$. We have encountered several normal subgroups in the main body of this book, as recounted in the following examples.

E**xample A2.9** The translation group $\mathbb{R}^3$ (see Sec. 6.1.2) is a normal subgroup of the rigid motion group $SE(3)$ (see Sec. 6.1.3).
$\mathbb{R}^3$, which can be represented by the matrix notation

$$\begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $d_x, d_y, \in \mathbb{R}$, is also a normal subgroup of the group of motions represented by the matrices of Eq. (6.18), which are

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & d_x \\ \sin\phi & \cos\phi & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $\phi, d_x, d_y, \in \mathbb{R}$

Prove these results.

Example A2.10 The group of translations along the *z*-axis represented by the matrices of Eq. (6.15), which are

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $d \in \mathbb{R}$, is a normal subgroup of the group of motions represented by the matrices of Eq. (6.11), which are

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where ϕ, $d \in \mathbb{R}$

It turns out that the group of rotations about the *z*-axis, which can be represented by the matrices

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $\phi \in \mathbb{R}$ is also a normal subgroup of the group of motions represented by the matrices of Eq. (6.11), just shown.

Prove these results.

Example A2.11 The group of translations in the *xy*-plane, that is, $\mathbb{R}^2$, which can be represented by the matrices

$$\begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where d_x, $d_y \in \mathbb{R}$, is a normal subgroup of the planar rigid motion group *SE*(2) represented by the matrices of Eq. (6.13), which are

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & d_x \\ \sin\phi & \cos\phi & 0 & d_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $\phi, d_x, d_y \in \mathbb{R}$.

$\mathbb{R}^2$ is also a normal subgroup of the group of motions represented by the matrices of Eq. (6.19), which are

$$\begin{bmatrix} \cos\phi & -\sin\phi & 0 & d_x \\ \sin\phi & \cos\phi & 0 & d_y \\ 0 & 0 & 1 & \mu\phi \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where $\phi, d_x, d_y \in \mathbb{R}$ and μ is a constant pitch.

Prove these results.

From the basic relationship $gN=Ng$, it can be seen that a normal subgroup has the important property that $gNg^{-1}=N$. The operation $g \bullet x \bullet g^{-1}$ is called the *conjugation* of x by g. So we see that a normal subgroup N is invariant under conjugation by any member of its parent group G. We also see that the conjugation of N by any member of G is a group automorphism of N.

A2.4 PRODUCTS OF GROUPS

It is possible to construct new groups from two given groups. The idea here is to construct the new group using the notion of products.

Given two sets M and N, the Cartesian product $M \times N$ is the set of ordered pairs (m, n) where $m \in M$ and $n \in N$ If $(M, \bullet)$ and $(N, \circ)$ are groups, then the group formed by the Cartesian product $M \times N$ under the group operation defined by $(m_1, n_1)(m_2, n_2)=(m_1 \bullet m_2, n_1 \circ n_2)$ is called the *direct product* of the groups M and N.

> **Example A2.12** Recall from Chapter 6 that $\mathbb{R}^3$ is the group of translations and $SO(3)$ is the group of rotations in three-dimensional space. Their direct product $\mathbb{R}^3 \times SO(3)$ is a group, but it is not the rigid motion group $SE(3)$. The reason is that the composition of two members from this direct product does not obey the rules of composition of two rigid motions. (Prove it.)

This last example provides a motivation for defining a more general product, called a *semidirect product,* as follows. First, we extend the notation of cosets and say that if A and B are subsets of a group G, then $AB=\{a \bullet b: a \in A, b \in B\}$. Now, let N and H be two subgroups of G. Then G is called a semidirect product of N and H and is denoted by $N \circledS H$, if the following hold:

1. N is a normal subgroup of G.
2. $N \cap H=\{e\}$.
3. $NH=G$.

We can now apply these conditions to see if the rigid motion can be expressed as a semidirect product.

1. $\mathbb{R}^3$ is a normal subgroup of the rigid motion group $SE(3)$ (see Example A2.9). $SO(3)$ is another, but not normal, subgroup of $SE(3)$.
2. The only common element between $\mathbb{R}^3$ and $SO(3)$ is the identity element.
3. The right coset $\mathbb{R}^3 SO(3)$, which is also equal to the left coset $SO(3)\mathbb{R}^3$ owing to the normality of $\mathbb{R}^3$, equals $SE(3)$.

Therefore, we can say that the rigid motion group is the semidirect product of the translation group and the rotation group; that is, $SE(3) = \mathbb{R}^3 \circledS SO(3)$

Example A2.13 Consider the cylindrical group represented by matrices of the form

$$C(\phi, d) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where ϕ and d are two independent parameters for rotation about the z-axis and translation along the z-axis, respectively It can be constructed from two simpler groups as follows. First, consider the rotation group represented by the matrices

$$R(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi & 0 & 0 \\ \sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Next, consider the translation group represented by the matrices

$$T(d) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

It is possible to construct the cylindrical group using the direct product $C(\phi, d) = T(d) \times R(\phi)$ (Prove it.) It is also possible to construct it using the semidirect product; that is, $C(\phi, d) = T(d) \circledS R(\phi)$ (Prove it.)

There are instances where products of groups can be defined unambiguously without specifying whether they are direct or semidirect products. If N is a normal subgroup of G and M is another subgroup, not necessarily normal, of G, then it is easy to see that $NM=MN$ and that it is a subgroup of G. We will then simply call MN or NM the product of M and N. All products of groups encountered in Chapter 6 are of this type.

A2.5 LIE GROUPS

Example A2.4 presented a group of six transformations, out of which three form a subgroup of rotations. That is, the rotations represented by the matrices m_1, m_2, and m_3 form a group. This is a *discrete* group because the members in this set are finite in number. In contrast, the rotations represented by matrices in Example A2.7 form *continuous* groups because the members in these sets are not finite in number.

Lie groups formalize the notion of continuous groups. A Lie group G is a smooth manifold for which the group operation and the group inverse are also smooth. Here *smoothness* refers to the condition that the function and its derivatives of all order are continuous.

Example A2.14 $\mathbb{R}^n$ is a Lie group. $\mathbb{R}^n$ is a smooth manifold. The vector addition is smooth. The inverse is the negation and so is smooth as well.

Example A2.15 The general linear group $GL(n, \mathbb{R})$ is the set of all real, nonsingular square matrices of order n. It is a Lie group because it can be seen from the formulas that the matrix multiplication (which is the group operation) and the matrix inversion are smooth.

Example A2.16 The special orthogonal group $SO(n, \mathbb{R})$ is a subgroup of $GL(n, \mathbb{R})$ In $SO(n, \mathbb{R})$ we require additionally that the row (column) vectors be orthogonal, which is the same as saying that the inverse is the same as the transpose, and that the determinant of each matrix is +1. Both $SO(n, \mathbb{R})$ and its specialization $SO(3, \mathbb{R})$ which is the group of rotations in three-dimensional space, are Lie groups.

Example A2.17 The special Euclidean group $SE(3, \mathbb{R})$ is the rigid motion group and is a Lie group. It has several Lie subgroups, that is, subgroups that also qualify as Lie groups. Some of the Lie subgroups have only one connected component (as opposed to several disjoint components); all such connected Lie subgroups of the rigid motion group are enumerated in Chapter 6.

A2.6 EXERCISES

1. Why is $(\mathbb{Z}\backslash 0, \times)$ not a group?
2. Following Example A2.3, enumerate all elements of the permutation group S_4. Can you identify a subgroup A_4 in S_4 that is similar to A_3? Can you relate A_4 to the symmetry group of a regular tetrahedron, under rigid motion?
3. Prove that the groups $(A, +)$ and $(B, \times)$ in Example 2.5 are isomorphic under exponentiation.
4. Relate the permutation group S_3 to the dihedral group D_3 of Chapter 6. Similarly, relate the alternating group A_3 to the cyclic group C_3 of Chapter 6.

A2.7 NOTES AND REFERENCES

Group theory started with the work of Galois in 1830 and was developed by people like Cayley in the 19th century. It acquired a central role in Klein’s Erlanger program, which essentially defined geometry as the study of transformation groups. This paved the way to look at geometries other than Euclidean.

There are several good books on group theory, such as Birkhoff and MacLane (1997) and Burn (1994). More recently, Leyton (2001) used group theory in a constructive definition of geometric shapes.

Appendix 3
Graphs

A graph is arguably the most useful abstraction for studying relationships among objects. This is definitely true in the case of the relationship among geometric objects. This appendix gives a brief summary of graph theory, which is used in the main body of the book, especially in Chapter 8.

A3.1 BASIC DEFINITIONS

A graph $G=(V, E, \psi)$ is a finite set V of elements called *vertices,* a finite set E of elements called *edges,* and an *incidence function* ψ that associates with each edge in E an unordered pair of (not necessarily distinct) vertices in V. The vertices are sometimes called *nodes* and the edges are sometimes called *arcs.* We will use $|E|$ and $|V|$ to denote the number of edges in E and the number of vertices in V, respectively It is also a general practice to use m to denote the number of edges and n to denote the number of vertices.

Example A3.1 Figure A3.1 shows a simple pictorial example of a graph with a vertex set $V=\{v_1, v_2, v_3\}$ and an edge set $E=\{e_1, e_2, e_3\}$. The incidence function is the mapping $\psi(e_1)=(v_1, v_2)$, $\psi(e_2)=(v_2, v_3)$, and $\psi(e_3)=(v_1, v_3)$. Note that

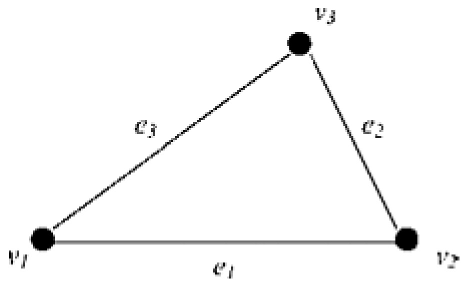


FIGURE A3.1 A graph with three vertices and three edges.

such pictorial representations are merely visual aids. The graph is completely defined by the vertex set, the edge set, and the incidence function.

A graph with no edges is an *empty graph*. A graph with no vertices (and, of course, no edges) is a *null graph*. If the incidence function ψ associates with each edge in E an ordered pair of (not necessarily distinct) vertices in V, then G is a *directed graph*. Unless otherwise specified, a graph is assumed to be undirected.

If two or more edges are associated with the same pair of vertices, then they are called *parallel edges*. If an edge is associated with two identical vertices, it is called a *self-loop*. A graph is *simple* if it has no parallel edges and no self-loops.

Example A3.2 Figure A3.2 shows an example of a graph with parallel edges and a self-loop. Its incidence function is the mapping $\psi(e_1)=(v_1, v_2)$, $\psi(e_2)=(v_2, v_3)$, $\psi(e_3)=(v_1, v_3)$, $\psi(e_4)=(v_1, v_3)$, and $\psi(e_5)=(v_2, v_2)$. Here, e_3 and e_4 are the parallel edges and e_5 is the self-loop.

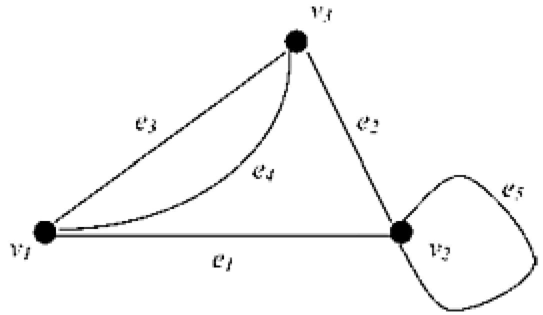


FIGURE A3.2 A graph with parallel edges and a self-loop.

There is a simple way to capture the incidence function using an *incidence matrix*. Consider a graph G with m edges and n vertices. Assume, for simplicity, that G has no self-loops. Then the incidence matrix $M(G)$ is an $n \times m$ matrix, where each row corresponds to a vertex and each column corresponds to an edge. An element m_{ij} in the matrix $M(G)$ is assigned the value 1 if the jth edge is incident on the ith vertex, and it is taken to be 0 otherwise. For example, the incidence matrix of the graph shown in Figure A3.1 is

$$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

Such matrices make it easy to represent graphs in computer software. The incidence matrix can be embellished to represent directed graphs by assigning +1 to m_{ij} if the jth edge is directed away from the ith vertex, and − 1 if the incident edge is directed toward the vertex. In a similar embellishment for a self-loop, m_{ij} may be assigned the value of 2.

It is possible to capture the incidence function more compactly using an *adjacency matrix*. Two vertices in a graph are called *adjacent* if there is an edge between them. Now, consider a graph G with m edges and n vertices. Also assume, for simplicity, that G has no parallel edges. The adjacency matrix $A(G)$ is an $n \times n$ matrix, where each row corresponds to a vertex and each column also corresponds to a vertex. An element a_{ij} in the matrix $A(G)$ is assigned the value 1 if there is an edge between the ith and the jth vertices, and is taken to be 0 otherwise. In this scheme, the adjacency matrix of the graph shown in Figure A3.1 is given by

$$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

A nonzero diagonal entry means that there are self-loops in the graph. Parallel edges can be accommodated by embellishing the adjacency matrix by setting the value of a_{ij} to be the number of such parallel edges between the *i*th and the *j*th vertices. Directed graphs can be represented by embellishing the adjacency matrix to have positive and negative elements, rendering the adjacency matrix asymmetric.

By relabeling the edges and vertices of a graph G but not changing its incidence function, a seemingly new graph G' may be created. Obviously, this is not a different graph. We capture this "sameness" of two graphs using the notion of *graph isomorphism*. (Compare this with the notion of group isomorphism presented in Appendix 2.) Two graphs G and G' are isomorphic if there is a one-to-one correspondence between the edge of G and G', a one-to-one correspondence between the vertices of G and G', and the corresponding edges of G and G' are incident on corresponding vertices of G and G'.

Consider a graph $G=(V, E, \psi)$. A graph $G'=(V', E', \psi')$ is a subgraph of G if V' is a subset of V, E' is a subset of E and ψ' is a restriction of ψ to E'. Two types of subgraphs of $G=(V, E, \psi)$ are of some special interest.

1. Suppose that V' is a nonempty subset of V. The subgraph of G whose vertex set is V' and whose edge set is the set of all those edges in G that have both vertices in V' is called a *vertex-induced subgraph of* G.
2. Now suppose that E' is a nonempty subset of E. The subgraph of G whose vertex set is all the endpoints of the edges in E' and whose edge set is E' is called an *edge-induced subgraph* of G.

For example, the graph shown in Figure A3.1 can be obtained as an edge-induced subgraph of the graph shown in Figure A3.2 by choosing the edge set $\{e_1, e_2, e_3\}$.

A3.2 GRAPH NAVIGATION

There are many ways to navigate a graph. By this we mean that we can traverse continuously through vertices and edges of a graph in many ways, which are referred to as walks, trails, paths, and circuits, as described next.

A *walk* in a graph is a finite alternating sequence of vertices and edges that can be denoted as $v_0, e_1, v_1, e_2, v_2, e_3, \ldots, v_{k-1}, e_k, v_k$, where v_{i-1} and v_i denote the end vertices of the edge e_i. In this walk, v_0 and v_k are the two *end,* or *terminal,* vertices of the walk. All other vertices are *internal* vertices of the walk. Such a walk is also referred to as a (v_0-v_k) walk. In a walk, edges and vertices can appear more than once. A walk is *open* if the end vertices are distinct; if the end vertices are the same the walk is *closed*.

A walk is a *trail* if all its edges are distinct. A trail can be open or closed. An open trail is a *path* if all its vertices are distinct. A closed trail is a *circuit* if all its vertices, except the end vertices, are distinct. A circuit is also called a *cycle*.

In a graph *G,* two vertices v_i and v_j are *connected* if there is a (v_i-v_j) path in *G*. A graph itself is *connected* if every pair of vertices in it is connected. A graph that is not connected contains two or more *connected components*. To see this, observe that vertex connectedness is an equivalence relation on the vertex set *V*. So it is possible to partition *V* into nonempty subsets $V_1, V_2, \ldots, V_k$ such that two vertices *u* and *v* are connected if only if both belong to the same set V_i. The subgraph of *G* induced by V_i is a connected component of *G*.

A3.3 SPECIAL GRAPHS

There are some special graphs that are of interest to us. A graph in which there is an edge between every pair of vertices is a *complete* graph. A complete graph with *n* vertices is unique up to isomorphism and is denoted by K_n. Figure A3.1 shows a complete graph K_3 with three vertices.

A tree is a special type of graph. To define a tree, we first observe that a graph is said to be *acyclic* if it has no circuits (or cycles). A *tree* is a connected, acyclic graph. A connected subgraph of a tree T is called a subtree of *T*. It can be shown that the following statements are equivalent for a graph *G* with *n* vertices and *m* edges:

1. *G* is a tree.
2. There is exactly one path between any two vertices of *G*.
3. *G* is connected and $m=n-1$.
4. *G* is acyclic and $m=n-1$.

These results are exploited in Section 8.2.2. An acyclic graph is also called a *forest* in which each connected component is a tree.

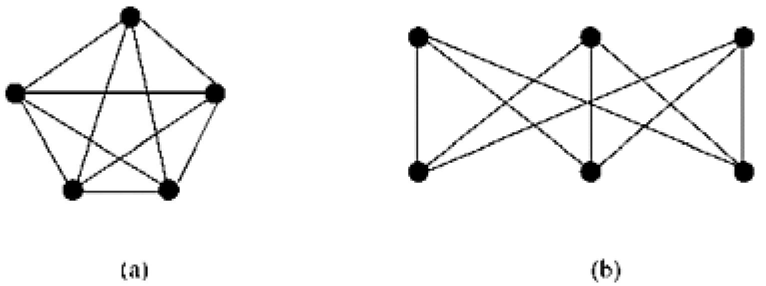


FIGURE A3.3 Two basic nonplanar graphs. (a) The complete graph K_5. (b) A graph normally denoted as $K_{3,3}$.

We have seen that graphs are pictorially represented by drawing them in a plane of paper. A graph is said to be embeddable in a plane, or *planar,* if it can be drawn in the plane so that its edges intersect only at their endpoints. Such a drawing is called a *planar embedding* of the graph. It is possible to draw every simple planar graph with each edge as a straight-line segment. (Recall that a simple graph has no parallel edges or self-loops.)

Two basic nonplanar graphs are shown in Figure A3.3. These graphs are considered basic because every nonplanar graph contains at least one of these two as a subgraph.

There is a remarkably simple theorem, due to Euler, that shows that the number of edges and vertices of a planar graph cannot be arbitrary; that is, they are related to each other. To see this, first observe that an embedding of a planar graph divides the plane into regions, one of which is the infinite region lying outside the graph. Euler showed that if a connected, planar graph has m edges, n vertices, and r regions, then $n-m+r=2$. This is known as the *Euler's formula*. It is a combinatorial result in the sense that it involves only the number of vertices, edges, and regions.

A3.4 GRAPH RIGIDITY

A four-bar linkage, shown in Figure A3.4 in the form of a graph, is a planar mechanism that consists of four rigid bars, shown as edges, and four revolute joints, shown as vertices. Although the bars are rigid, their assembly as a mechanism is not. It can flex, or deform, in many configurations, and three of these are shown in Figure A3.4. In contrast, Figure A3.5 shows a graph that can be interpreted as the assembly of five rigid bars, again using revolute joints, and it is a rigid structure.

It is easy to extend the notion of linkages to build mechanisms in space. Figure A3.6 shows a nine-bar linkage in three-dimensional space. The edges in the graph are rigid bars, and the vertices are spherical (ball-and-socket) joints. The assembled mechanism is not rigid, and three of its deformed configurations are shown in Figure A3.6.

These examples provide some intuition behind the problem of graph rigidity. Roughly speaking, we start with a connected graph and assume that the edges are rigid bars (that is, the edge lengths are fixed) and the vertices are revolute joints if we are considering a problem in the plane, or spherical joints if we are considering a problem in space. Such an assembly will be called a *framework*. We then ask if this framework is a mechanism that can flex or a structure that is rigid.

More formally, in a connected graph G with edge set E and vertex set V, we denote the vertex v_i by a point $p_i(t)$ with the understanding that it has two

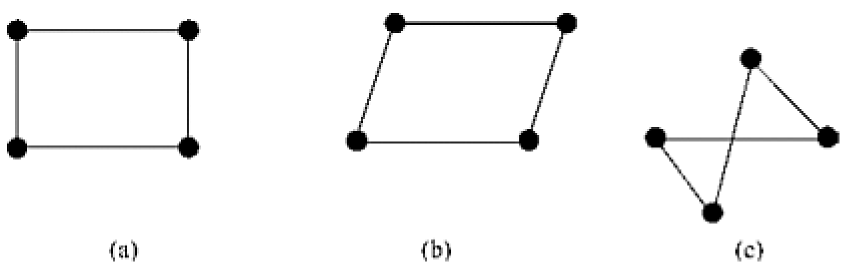


FIGURE A3.4 A four-bar linkage in the plane in three different configurations.

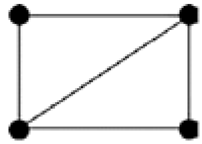


FIGURE A3.5 A planar five-bar framework that is rigid.

coordinates in a planar problem and three coordinates in a space problem, and t is a continuous parameter. Thus we have an embedding of the graph G in a plane or in space, and the edges in G are allowed to cross each other. As before, let m be the number of edges in E and n be the number of vertices in V. The edge length constraints are captured by the set of equations

$$(p_i(t) - p_j(t)) \cdot ((p_i(t) - p_j(t)) = d_{ij}^2 \tag{A3.1}$$

where • indicates vector dot product, for each edge in E with endpoints $p_i(t)$ and $p_j(t)$. The framework represented by the graph G is said to be rigid if all solutions to Eq. (A3.1) are locally trivial, that is, if the tuple $(p_1(t), p_2(t), \ldots, p_n(t))$ is congruent under rigid motion to the tuple $(p_1(0), p_2(0), \ldots, p_n(0))$ for all t near 0. An equivalent statement is that the graph G is rigid if the distance between any pair of vertices in V is preserved whether they are adjacent or not.

Solving the set of equations in Eq. (A3.1) is difficult in general. There are several delicate special cases that need to be handled. For example, if the nine-bar framework of Figure A3.6 is flattened to lie in a plane, as shown in Figure A3.7, it becomes rigid. This rigid configuration is unstable because if all the six vertices are not perfectly coplanar, some flexibility of the framework is possible.

When faced with such a difficult problem, the usual mathematical practice is to simplify the problem by redefining it. This is precisely what is done in the following definitions of infinitesimal rigidity and generic rigidity.

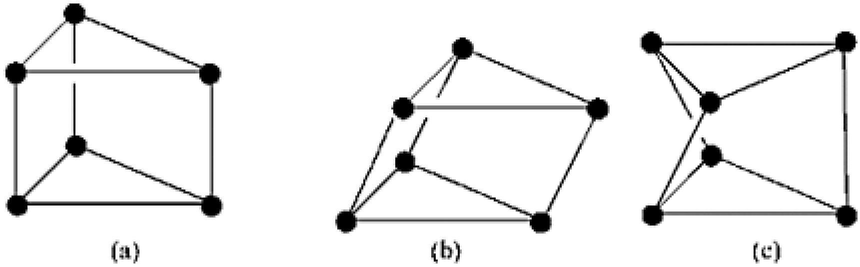


FIGURE A3.6 A nine-bar linkage in space in three different configurations.

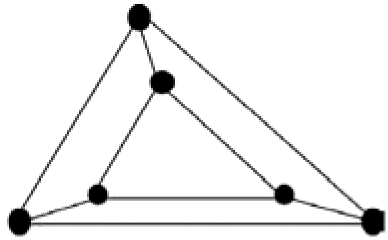


FIGURE A3.7 A planar nine-bar framework that is rigid.

A3.4.1 Infinitesimal Rigidity

The expression in Eq. (A3.1) can be differentiated with respect to *t,* yielding

$$(p_i(t) - p_j(t)) \cdot ((p_i'(t) - p_j'(t)) = 0 \tag{A3.2}$$

where $p_i'(t)$ can be considered to be the velocity of ith vertex. A geometrical interpretation of this equation can be given by rewriting it as

$$(p_i(t) - p_j(t)) \cdot p_i'(t) = (p_i(t) - p_j(t)) \cdot p_j'(t) \tag{A3.3}$$

This means that in the rigid bar connecting p_i and *pj,* the component of the velocities at the endpoints along the length of the bar are equal. This makes sense because the bar is rigid and its length should remain invariant, which would not be the case if the two endpoints move at different velocities along the length of the bar.

Note that Eq. (A3.2) does not contain the distance d_{ij}^2. It gives one equation per edge in the graph, and therefore there are *m* equations in total. In this set of equations we will treat the positions of the vertices, that is, $p_i(t)$, $i=1,\ldots, n$, as known quantities and the velocities of the vertices, that is, $p_i'(t)$, $i=1,\ldots, n$, as unknowns. Since each velocity has two or three components, depending on whether we are dealing with a two-dimensional or a three-dimensional framework, there are *nd* unknowns, where *d* is the dimension of the space in which the framework is embedded.

We thus have a system of *m* linear equations in *nd* unknowns and we can bring in all the powerful machinery of matrices and linear algebra to analyze it. The $m\times nd$ matrix in this set of linear equations is called the *rigidity matrix*.

Example A3.3 Consider the planar four-bar framework shown in Figure A3.8 (a). Assume that the coordinates of the four vertices are $p_1=(0, 0)$, $p_2=(2, 0)$, $p_3=(2, 1)$, $p_4=(0, 1)$. If we denote

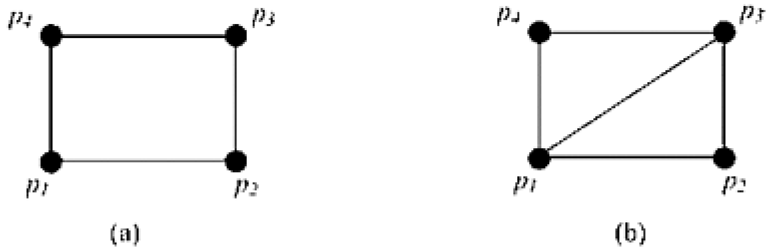


FIGURE A3.8 Two planar frameworks.

$p_i' = (x_i', y_i')$, the set of equations given by Eq. (A3.2) can be written explicitly as

$$\begin{bmatrix} -2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & -2 & 0 \end{bmatrix} \begin{Bmatrix} x_1' \\ y_1' \\ x_2' \\ y_2' \\ x_3' \\ y_3' \\ x_4' \\ y_4' \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \quad \text{(A3.4)}$$

The 4×8 matrix on the left-hand side of Eq. (A3.4) is the rigidity matrix for this framework.

If a framework has at least d+1 vertices in general position, as we will assume from now on, there are always $d(d+1)/2$ solutions to the set of linear equations in Eq. (A3.2) that correspond to the rigid motion of the entire framework. To be specific, for two-dimensional frameworks there are always three solutions to the velocities (two translational components and one rotational component), and for three-dimensional frameworks there are always six solutions to the velocities (three translational components and three rotational components), which correspond to the rigid motion. If there are no other solutions, then we claim that the framework has infinitesimal rigidity (also called *first-order* rigidity). We thus have the following theorem.

Theorem A3.1 *A framework with n vertices is infinitesimally rigid in d-dimensional space if and only if its rigidity matrix has a rank of* $nd-(d(d+1))/2$.

Example A3.4 The rigidity matrix in the matrix equation (A3.4) has a rank of 4. To be infinitesimally rigid, Theorem A3.1 demands that it should have a rank of 5. Therefore, the framework of Figure A3.8(a) is not infinitesimally rigid. We also know from prior engineering knowledge that this framework is a four-bar linkage and is not rigid.

Example A3.5 Consider the framework of Figure A3.8(b). It is a five-bar framework whose rigidity matrix is given by

$$\begin{bmatrix} -2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & -2 & 0 \\ -2 & -1 & 0 & 0 & 2 & 1 & 0 & 0 \end{bmatrix} \tag{A3.5}$$

The rank of this matrix is 5, which is the same as demanded by Theorem A3.1. Therefore the framework of Figure A3.8(b) is infinitesimally rigid. From our prior engineering knowledge we know that this is a rigid structure.

What is the relationship between a rigid framework and an infinitesimally rigid framework? There is a general theorem regarding this.

Theorem A3.2 *Infinitesimal rigidity implies rigidity.*

The converse of this theorem need not be true, as the following example illustrates.

Example A3.6 Consider the planar nine-bar framework shown in Figure A3.9. Assume that its six vertices have coordinates p_1=(0, 0), p_2=(3, 0), p_3=(1, 1), p_4=(0, 3), p_5=(3, 3), p_6=(1, 2). The rigidity matrix for this framework is given by

$$\begin{bmatrix} -3 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -3 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -3 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -3 & 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & -2 & -1 \end{bmatrix} \tag{A3.6}$$

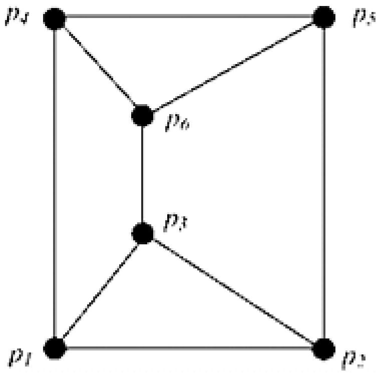


FIGURE A3.9 A planar nine-bar framework.

Its rank is 8. From Theorem A3.1, we would then infer that the framework shown in Figure A3.9 is not infinitesimally rigid because its rank is not 9. It is, however, a rigid structure.

Before we leave the section on infinitesimal rigidity, let's consider what it does and does not accomplish. Note that the rigid framework problem, which depends on solving equations defined by Eq. (A3.1), is quite different from the infinitesimally rigid framework problem, which depends on solutions to the equations of Eq. (A3.2). In the rigid framework problem, only the lengths of the edges of the graph that represents the framework are specified. On the other hand, in the infinitesimally rigid framework problem, we completely ignore the edge lengths and assume that the coordinates of the vertices of the graph representing the framework are known; we then merely determine whether such a framework can flex or be a rigid structure. This leads to a simple characterization of infinitesimal rigidity in terms of the rank of a rigidity matrix.

If we can formulate an infinitesimal rigidity problem from a rigidity problem—and this is by no means a simple step—then we can easily determine if the framework is infinitesimally rigid. If the framework is infinitesimally rigid, then we can be sure that the framework is rigid. If not, the framework may or may not be rigid.

A3.4.2 Generic Rigidity

Determining rigidity or infinitesimal rigidity of a framework requires metrical considerations such as edge lengths or vertex coordinates of the graph that represents the framework. In generic rigidity we consider only combinatorial rigidity; that is, we concentrate on the graph structure of the framework and ignore any metric information. By this we mean that only the number of edges, the number of vertices, and the incidence relationship of the edge and vertex sets of the graph are considered in generic rigidity.

It is important to recognize at the outset that we should not expect too much to come out of this type of approach. Observe that the graphs of Figures A3.9, A3.10(a), and A3.10(b) are all isomorphic; but the frameworks in Figures A3.9 and A3.10(a) are rigid, whereas the framework of Figure A3.10(b) is a pantograph and is flexible. Nevertheless, it is theoretically attractive to see what we can infer from a purely combinatorial analysis of the graph of a framework. We start with a few definitions to set the stage for such a study.

A framework embedded in *d*-dimensional space and associated with a graph *G* can be represented as *(G, p),* where *p* is the set of coordinates of the vertices in *G*. Such a framework is called generic if all frameworks sufficiently near *p* have the same infinitesimal rigidity properties as *(G, p)*. Also, the embedding given by the vertex coordinates *p* is said to be a generic embedding if each framework with any graph on the vertex set *V* is generic. A graph is *generically rigid* in *d*-dimensional space if it has a generic embedding that is rigid. So generic rigidity is a property of the underlying graph. It can be shown that infinitesimal rigidity implies generic rigidity. Figure A3.11 shows the relationship among rigidity, infinitesimal rigidity, and generic rigidity.

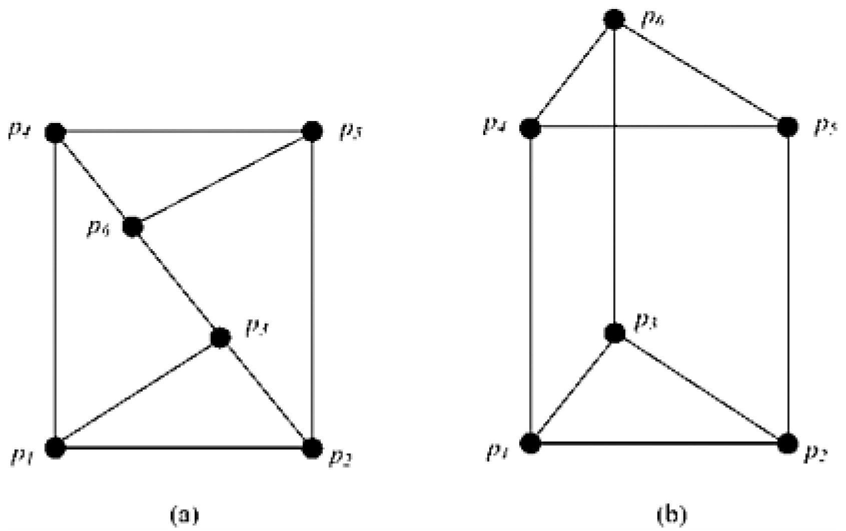


FIGURE A3.10 Two planar nine-bar frameworks.

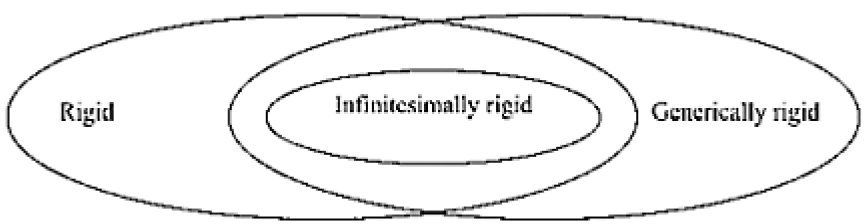


FIGURE A3.11 Relationship among rigid, infinitesimally rigid, and generically rigid frameworks.

In the one-dimensional case, rigidity, infinitesimal rigidity, and generic rigidity coincide. All these are equivalent to connectivity of the underlying graph. Note that this is purely a graph theoretic notion—we don't have to consider any metric information for this analysis. This is one of the simple, but nice, results of genetic rigidity. This result is used in Section 8.2.2 for solving one-dimensional constraint problems.

In the two-dimensional case, there is a surprisingly powerful result, due to Laman, that is captured in the following theorem.

Theorem A3.3 *A graph G with edge set E and vertex set V is generically rigid in the plane if only if there is a subset F of E that satisfies the following two conditions.*

1. $|F|=2|V|-3$.
2. *For all* $F' \subseteq F$, $F' \neq \emptyset$, $|F'| \leq 2k-3$, *where k is the number of vertices that are endpoints of edges in* F'.

Note that this theorem provides both necessary and sufficient conditions for a graph to be generically rigid in two-dimensional space. This is a purely graph-theoretic characterization and does not involve constructing a rigidity matrix and computing its rank as in infinitesimal rigidity. Applying this theorem one may conclude that the graph of all three two-dimensional frameworks in Figures A3.9 and A3.10 is generically rigid. But, as we know, the framework of Figure A3.10(b) is not rigid. This example illustrates the limitation of the notion of generic rigidity for determining the rigidity of a planar framework.

A similar characterization of generic rigidity is not yet known in the three-dimensional case. It remains an open problem. Even if a Laman-like theorem can be found for the generic rigidity of three-dimensional frameworks, it is not clear how useful that would be for determining the rigidity of frameworks.

A3.5 CONSTRAINT GRAPHS

In solving dimensional constraint problems, one often starts with constructing a *constraint graph* to represent the specified constraints. In a constraint graph the vertices correspond to geometric elements—such as points, lines, planes, and helices—and the edges correspond to specified geometric constraints between the elements they connect. These constraints may be distance or angle values. It is customary to indicate basic constraints, such as incidence and tangency, as special types of constraints in such a graph.

Example A3.7 Figure A3.12(a) shows a simple dimensioning scheme of a planar quadrilateral that is self-evident. Its dimensional constraint graph is shown in Figure A3.12(b), which can be interpreted as follows. In this constraint graph:

Nodes indicated as Ⓟ represent points and nodes indicated as Ⓛ represent straight lines.

A labeled edge between two point nodes may represent a specified distance constraint between the points, as shown.

When a line node is connected to a point node by an unlabeled edge, it indicates an incidence relationship between the line and the point. So if a line node is connected to two point nodes, then the line passes through these two points.

A labeled edge between two line nodes may represent a specified angle constraint between the lines, as shown.

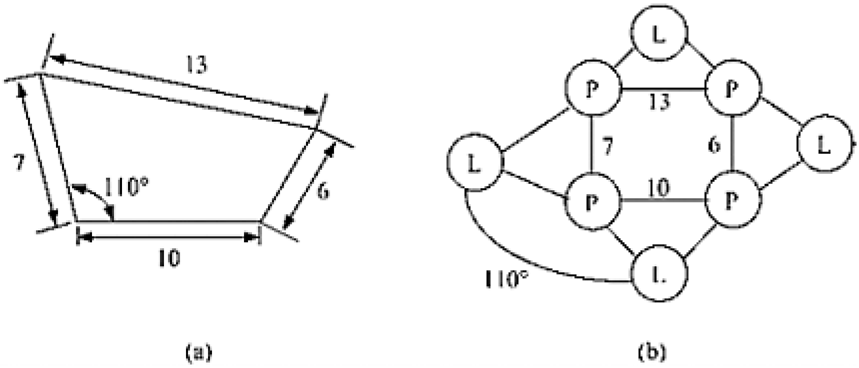


FIGURE A3.12 (a) A dimensioned quadrilateral, and (b) its dimensional constraint graph.

Such constraint graph representations are very useful in analyzing and resolving dimensional constraints described in Chapter 8. As remarked at the beginning of this appendix, graphs are most useful in capturing such relationships among geometric objects.

A3.6 EXERCISES

1. What are the incident and adjacency matrices of the graph shown in Figure A3.2?
2. Assume that the coordinates of the six vertices of the nine-bar framework of Figure A3.10(a) are p_1=(0, 0), p_2=(3, 0), p_3=(2, 1), p_4=(0, 3), p_5=(3, 3), and p_6=(1, 2). What is its rigidity matrix? Is it infinitesimally rigid? Is it a rigid framework?
3. Assume that the coordinates of the six vertices of the nine-bar framework of Figure A3.10(b) are p_1=(0, 0), p_2=(3, 0), p_3=(1, 1), p_4=(0, 3), p_5=(3, 3), and p_6=(1, 4). What is its rigidity matrix? Is it infinitesimally rigid? Is it a rigid framework?

A3.7 NOTES AND REFERENCES

Graph theory originated in 1736 with Euler, who solved the problem of crossing the bridges of Königsberg using the notion of graphs. There are several excellent textbooks on graph theory, such as Bondy and Murty (1976) and Thulasiraman and Swamy (1992). For a treatment of rigidity theory and its applications, see Thorpe and Duxbury (1999).

Appendix 4
Solids

Even though engineers tend to dimension various geometric elements and objects, ultimately their goal is to dimension solids. This is the topic of Chapter 9. It is well-known that projected views and wireframes are ambiguous, in the sense that they may represent multiple solids. Therefore, it is important to define solids unambiguously so that we may then represent and manipulate them in a computer. A cornerstone of modern CAD is a rigorous theory of solids that is decoupled from their representations. In this appendix we will briefly review this mathematical theory of solids, which is based on set theory.

A4.1 A PRIMER ON SET THEORY

A *set* is an unordered collection of objects called *elements*. If x is an element of a set X, we can denote this symbolically as $x \in X$ This may be read as "x is an element of X" or "A: is a member of X." The elements may be defined in many ways. A set of students in a particular class can be defined by listing their names or their university identification numbers; this is an example of a finite set. A set of points that lie on a circle can be defined by an equation of the circle; this is an example of an infinite set. Sets don't contain duplicate elements. If a collection contains duplicate elements, we promptly delete these duplicate elements from the collection before defining a set.

Y is a subset of X if every element of Y is also an element of X. We will denote this as $Y \subseteq X$ Two sets X and Y are equal, denoted by X=Y, if every element of Y is an element of X and every element of X is an element of Y. In other words, X=Y if X is a subset of Y and Y is a subset of X. Since sets are unordered collections, equal sets can have their elements in different order. Y is a proper subset of X, denoted by $Y \subset X$ if Y is a subset of X but not equal to X.

A universal set W is a set of all possible elements. A null set $\emptyset$ contains no element at all. With these preliminaries, we are ready to define operations that can be performed on sets. The most basic of these operations are the following Boolean operations. Assume that A and B are sets.

Union:	$A \cup B = \{x: x \in A \text{ or } x \in B\}$
Intersection:	$A \cap B = \{x: x \in A \text{ and } x \in B\}$
Complement	$\overline{A} = \{x: x \notin A\}$
Difference:	$A - B = A \cap \overline{B}$

Symmetric difference: $A \triangle B = (A - B) \cup (B - A)$

These Boolean operations possess the following important properties.

Commutativity:

$$A \cup B = B \cup A$$
$$A \cap B = B \cap A$$

Associativity:

$$(A \cup B) \cup C = A \cup (B \cup C)$$
$$(A \cap B) \cap C = A \cap (B \cap C)$$

Distributivity:

$$(A \cup B) \cap C = (A \cap C) \cup (B \cap C)$$
$$(A \cap B) \cup C = (A \cup C) \cap (B \cup C)$$

Identity sets:

$$A \cup \varnothing = A,\ A \cap W = A$$
$$A \cup \overline{A} = W,\ A \cap \overline{A} = \varnothing$$

Sets and Boolean operations that satisfy the preceding properties form a Boolean algebra. All Boolean algebras have the following useful properties.

Idempotency: $A \cup A = A,\ A \cap A = A$

Involution: $\overline{(\overline{A})} = A$

de Morgan's laws:

$$\overline{(A \cup B)} = \overline{A} \cap \overline{B}$$
$$\overline{(A \cap B)} = \overline{A} \cup \overline{B}$$

From these properties, it is clear that the Boolean operations are not independent. If, for example, we know how to do intersection and complementation, then we can construct union, difference, and symmetric difference from these two operations. This is how Boolean operations are implemented in modern CAD systems.

The Boolean operations are best illustrated using a Venn diagram, where sets are represented as simple regions, such as circular or elliptical disks, in a two-dimensional plane. So Venn diagrams provide a trivial example of how set theory can be applied to point-sets in a plane. However, the conventional set theory outlined so far is necessary but not sufficient to define solids in terms of point-sets. For this, we need some point-set topology.

A4.2 POINT-SET TOPOLOGY

Consider a set of points, or point-sets for short, in a general d-dimensional Euclidean space in the following development. The Euclidean distance between two points p and q will be denoted by $|p-q|$. An open ball $B(p, r)$ centered at the point p and having radius r can be defined as the point-set $B(p, r)=\{q:|p-q|<r\}$. The universal point-set W is the set of all points in that space. We can now define open and closed sets.

Open set: $X \subseteq W$ is open if for every point $\overline{X}$, there is a radius $r>0$ such that $B(p, r) \subseteq X$.

Closed set: $X \subseteq W$ is closed if $\overline{X}$ is open.

Example A4.1 In three-dimensional space:

1. $X=\{(x, y, z): x^2+y^2+z^2<4\}$ is open.
2. $X=\{(x, y, z): x^2+y^2>1\}$ is open.
3. $X=\{(x, y, z): x^2+y^2+z^2<4$ or $z=0\}$ is not open.
4. $X=\{(x, y, z): x^2+y^2+z^2\leq 4\}$ is closed.
5. $X=\{(x, y, z): x^2+y^2\geq 1\}$ is closed.
6. $X=\{(x, y, z): x^2+y^2+z^2\leq 4$ or $z=0\}$ is closed.

Next, we need to define the interior and boundary of a point-set.

Interior point: p is an interior point of X if there is a radius $r>0$ such that $B(p, r) \subseteq X$.

Interior. $iX=\{p: p$ is an interior point of $X\}$.

Example A4.2 In three-dimensional space:

1. If $X=\{(x, y, z): x^2+y^2+z^2\leq 4\}$, then $iX=\{(x, y, z): x^2+y^2+z^2<4\}$.
2. If $X=\{(x, y, z): x^2+y^2+z^2<4\}$, then $iX=X$.
3. If $X=\{(x, y, z): x^2+y^2\leq 1$ or $(0\leq x\leq 2, y=0, z=0)\}$, then $iX=\{(x, y, z): x^2+y^2<1\}$.

Boundary point: p is a boundary point of X if for any radius $r>0$, $B(p, r) \cap \overline{X} \neq \varnothing$ and $W = (iX) \cup \partial X \cup i(\overline{X})$. Note that a boundary point of X need not be an element of X.

Boundary: $\partial X=\{p: p$ is a boundary point of $X\}$.

Example A4.3 In three-dimensional space:

1. If $X=\{(x, y, z): x^2+y^2+z^2\leq 4\}$, then $\partial X= \{(x, y, z): x^2+y^2+z^2=4\}$. Note that X is closed.
2. If $X=\{(x, y, z): x^2+y^2+z^2<4\}$, then $\partial X= \{(x, y, z): x^2+y^2+z^2=4\}$. Note that X is open.
3. If $X=\{(x, y, z): x^2+y^2\leq 1$ or $(0\leq x\leq 2, y=0, z=0)\}$, then $\partial X=\{(x, y, z): x^2+y^2=1$ or $(1\leq x\leq 2, y=0, z=0)\}$.

We can now partition the universe as $W = (iX) \cup \partial X \cup i(\overline{X})$ for any $X \subset W$

Finally, we introduce the notion of *closure of a set,* which then leads to an important operation called *regularization of a set*.

Closure: $cl(X)=iX \cup \partial X$.

Example A4.4 In three-dimensional space:

1. If $X=\{(x, y, z): x^2+y^2+z^2\leq 4\}$, then $cl(X)=X$.
2. If $X=\{(x, y, z): x^2+y^2+z^2<4\}$, then $cl(X)= \{(x, y, z): x^2+y^2+z^2\leq 4\}$.
3. If $X=\{(x, y, z): x^2+y^2<1$ or $(0\leq x\leq 2, y=0, z=0)\}$, then $cl(X)=\{(x, y, z): x^2+y^2\leq 1$ or $(0\leq x\leq 2, y=0, z=0)\}$.

Regularization: $rX=cl(iX)$.

Regular set: X *is* a regular set if and only if $X=iX$.

Example A4.5 In three-dimensional space:

1. If $X=\{(x, y, z):x^2+y^2+z^2\leq 4\}$, then $rX=X$. Hence X is a regular set.
2. If $X=\{(x, y, z):x^2+y^2+z^2<4\}$, then $rX=\{(x, y, z):x^2+y^2+z^2\leq 4\}$. So X is not a regular set.
3. If $X=\{(x, y, z):x^2+y^2<1 \text{ or } (0\leq x\leq 2, y=0, z=0)\}$, then $rX=\{(x, y, z):x^2+y^2\leq 1\}$. So X is not a regular set.

So regularization has the effect of removing unwanted lower-dimensional entities protruding from solid objects.

A4.3 SOLIDS AND REGULAR SETS

Our aim here is to establish solids as regular point-sets. To motivate this definition, we will first consider regularized versions of Boolean operations.

Regularized union:	$A\cup^*B = cl(i(A\cup B))$
Regularized intersection:	$A\cap^*B=cl(i(A\cap B))$
Regularized complement:	$\overline{A}^* = cl(i(\overline{A}))$
Regularized difference:	$A-^*B=cl(i(A-B))$

If we start with regular sets A and B, then their regularized union is the same as their conventional union; that is, $A\cup^*B = A\cup B$ But the other regularized operations need not correspond to their conventional counterparts, as the following examples show.

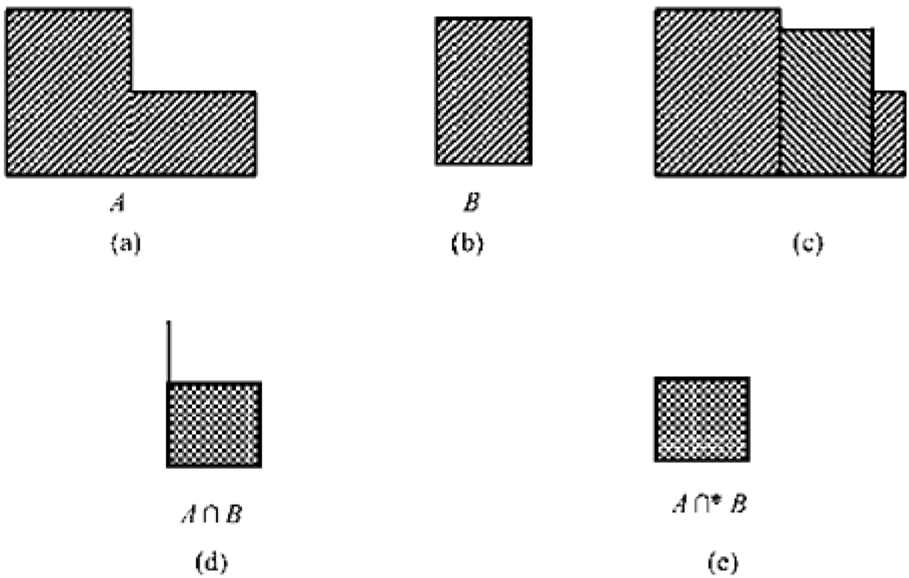


FIGURE A4.1 Distinction between conventional and regularized intersections between two regular sets *A* and *B*. The relative positioning of *A* and *B* is shown in (c).

Example A4.6 Figure A4.1 shows how regularized and conventional intersection operations can differ even if both operate on regular sets *A* and *B* in two-dimensional space. In this illustration the conventional intersection of *A* and *B* results in a line segment protruding from the rectangular region. This segment is eliminated in the regularized intersection.

E**xample A4.7** Figure A4.2 shows how regularized and conventional complementation operations differ even on a regular set *A* in two-dimensional space. The conventional complementation does not include the circular boundary, whereas the regularized complementation does.

E**xample A4.8** Figure A4.3 shows how regularized and conventional difference operations can differ even if both operate on regular sets *A* and *B* in two-dimensional space. In the conventional difference, the bottom line segment in the rectangular boundary is missing. The entire boundary is part of the regularized difference.

Regular sets and regularized Boolean operations satisfy the properties of commutativity, associativity, distributivity, and identity sets defined earlier. Therefore, they form a Boolean algebra and hence they also satisfy idempotency, involution, and de Morgan's laws, defined in Section A4.1.

We are now ready to define a solid as a regular set. In three-dimensional space, a solid is a regular subset of the three dimensional Euclidean space. Often, we also restrict solids to be bounded. (A set is bounded if there is an open ball of finite radius that contains the set.) A solid defined this way has the following characteristics:

1. Rigidity
2. Homogeneous three dimensionality (that is, no lower-dimensional entities hanging from it)

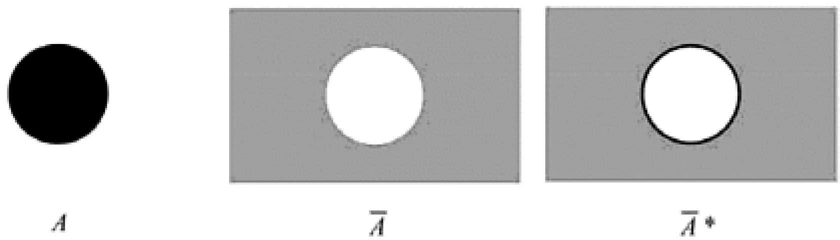


FIGURE A4.2 Distinction between conventional and regularized complementations of a regular set *A*.

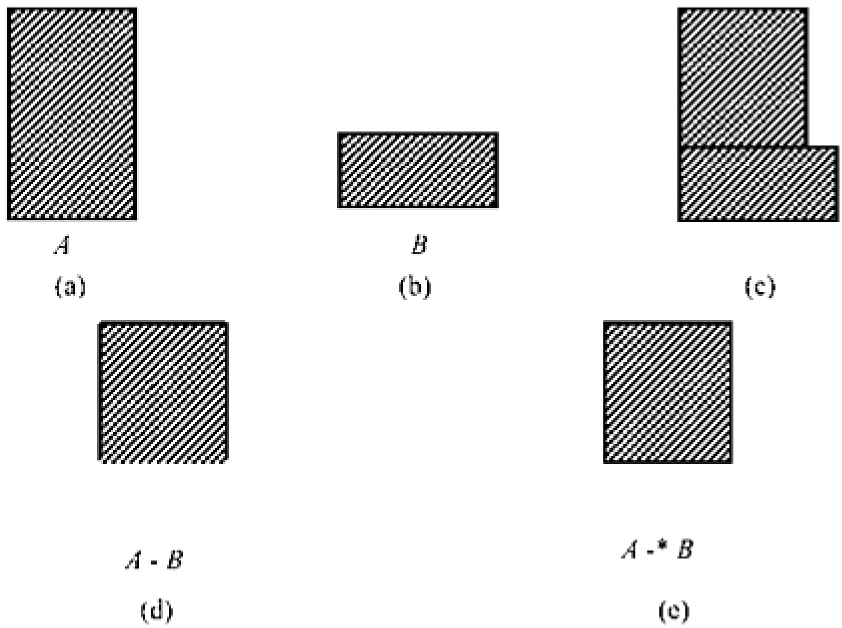


FIGURE A4.3 Distinction between conventional and regularized differences between two regular sets *A* and *B*. The relative positioning of *A* and *B* is shown in (c).

3. Finiteness (if boundedness is invoked)
4. Closure under rigid motions and regularized Boolean operations

It is possible to specialize the notion of a solid to two-dimensional space, by defining it as a regular set in two-dimensional space. This is a very useful concept that is often used in geometric modeling.

Before we leave this section, we should note that the notion of regularization is not restricted to Boolean operations. It is possible to define regularized sweep operations by taking the closure of the interior of the result of conventional sweeps.

A4.4 REPRESENTATIONS OF SOLIDS

A representation is a way of describing a particular object. It is usually a symbol structure constructed according to some well-defined syntactic rules and the meaning is given by its semantics. In Chapter 9 we saw several examples of representations of solids. One is the boundary representation, where the solid is represented by a complete description of its boundary. Another representation is the CSG tree, which is a particular case of procedurally defined solids.

Most modern CAD systems have adopted procedural representations of solids. Often, these are converted to boundary representations for various applications. The theory of solids described in this appendix is independent of the representations used to capture it in a computer. However, as we saw in Chapter 9, solid dimensioning is closely tied to the representation used to define the solid.

A4.5 EXERCISES

1. Give an example of projected views that may represent more than one solid.
2. Give an example of a three-dimensional wireframe that may represent more than one solid.

A4.6 NOTES AND REFERENCES

Tylove and Requicha (1980) describe regularized Boolean operations and their role in solid modeling. For more information on solid representations, refer to Mantyla (1988) and Hoffmann (1989).

Bibliography

ASME Y14.5M-1994. *Dimensioning and Tolerancing,* The American Society of Mechanical Engineers, New York, 1995.

ASME Y14.5.1M-1994. *Mathematical Definition of Dimensioning and Tolerancing Principles,* The American Society of Mechanical Engineers, New York, 1995.

Bellman, R. *Introduction to Matrix Analysis,* McGraw-Hill, New York, 1970.

Birkhoff, G., and MacLane, S. *A Survey of Modern Algebra,* 4th ed., Prentice Hall, Upper Saddle River, NJ, 1997.

Bondy, J.A. and Murty, U.S.R. *Graph Theory with Applications,* North-Holland, New York, 1976.

Booker, P.J. *A History of Engineering Drawing,* Chatto & Windus, London, 1963.

Bouma, W., Fudos, I., Hoffmann, C., Cai, J., and Paige, R. A geometric constraint solver, *Computer-Aided Design,* Vol. 27, 1995, pp. 487–501.

BS 308–1927. *Engineering Drawing Office Practice,* British Standard No. 308–1927, British Engineering Standards Association, London, 1927.

Burn, R.P. *Groups—A Path to Geometry,* Cambridge University Press, Cambridge, UK, 1994.

Clement, A., Riviere, A., and Temmerman, M. *Cotation Tridimensionnnelle des Systemes Mechaniques,* Ivry-sur-Seine: PYC Edition, France, 1994.

Denavit, J., and Hartenberg, R.S. A kinematic notation for lower-pair mechanisms based on matrices, *Journal of Applied Mechanics,* Transactions of the ASME, pp. 215–221, June 1955.

Dougherty, E.R. *An Introduction to Morphological Image Processing,* SPIE—The International Society for Optical Engineering, Bellingham, WA, 1992.

Farin, G. *Curves and Surfaces for Computer-Aided Geometric Design,* 3rd ed., Academic Press, San Diego, CA, 1993.

French, T.E. *A Manual of Engineering Drawing for Students and Draftsmen,* 2nd ed., McGraw-Hill, New York, 1918.

French, T.E., and Svensen, C.L. *Mechanical Drawing,* 7th ed., McGraw-Hill, New York, 1966.

Fudos, I., and Hoffmann, C. A graph-constructive approach to solving systems of geometric constraints, *ACM Transactions on Graphics,* Vol. 16, 1997, pp. 179–216.

Golub, G.H., and Van Loan, C.F. *Matrix Computations,* 3rd ed., The Johns Hopkins University Press, Baltimore, MD, 1996.

Hilbert, D., and Cohn-Vossen, S. *Geometry and the Imagination,* Chelsea, New York, 1983.

Hoffmann, C.M. *Geometric and Solid Modeling,* Morgan Kaufmann, San Mateo, CA, 1989.

ISO 128–1982. *Technical Drawings—General Principles of Presentation,* International Organization for Standardization, Geneva, 1982.

ISO/R 129–1959. *Engineering Drawing—Dimensioning,* International Organization for Standardization, Geneva, 1959.

ISO/TS 10303–108:2003. *Parameterization and Constraints for Explicit Geometric*

Product Models, International Organization for Standardization, Geneva, 2003.
ISO/TS 17450–1:2003. *Geometrical Product Specifications (GPS)—Model for Geometric Specification and Verification,* International Organization for Standardization, Geneva, 2003.
Laman, G. On graphs and rigidity of plane skeletal structures, *Journal of Engineering Mathematics,* Vol. 4, No. 4, 1970, pp. 331–340.
Lanczos, C. *Applied Analysis,* Pitman, London, 1957.
Leyton, M. *A Generative Theory of Shape,* Springer-Verlag, Berlin, 2001.
Lipschutz, M.M. *Differential Geometry,* McGraw-Hill, New York, 1969.
Mantyla, M. *An Introduction to Solid Modeling,* Computer Science Press, Rockville, MD, 1988.
Matheron, G. *Random Sets and Integral Geometry,* Wiley, New York, 1975.
O'Connor, M.A., Srinivasan, V., and Jones, A. Connected lie and symmetry subgroups of the rigid motions: foundations and classifications, *IBM Research Report RC 20512,* Yorktown Heights, NY, July 1996.
Olmsted, J.M.H. *Solid Analytic Geometry,* Appleton-Century-Crofts, New York, 1947.
Owen, J.C. Algebraic solution for geometry from dimensional constraints, *Proceedings of Symposium on Solid Modeling Foundations and CAD/CAM Applications,* ACM SIGGRAPH, Austin, TX, June 5–7, 1991.
Owen, J.C. Constraints on simple geometry in two and three dimensions, *International Journal of Computational Geometry & Applications,* Vol. 6, No. 4, pp. 421–434, 1996.
Requicha, A.A.G., and Voelcker, H.B. Boolean operations in solid modeling: boundary evaluation and merging algorithms, *Proceedings of the IEEE,* Vol. 73, pp. 30–44, 1985.
Rutter, J.W. *Geometry of Curves,* Chapman & Hall/CRC, Boca Raton, FL, 2000.
Shah, J.J., and Mantyla, M. *Parametric and Feature-Based CAD/CAM,* Wiley, New York, 1995.
Shapiro, V., and Vossler, D.L. What is a parametric family of solids? *Proceedings of the Third ACM/IEEE Symposium on Solid Modeling and Applications,* Salt Lake City, UT, May 17–19, 1995.
Sommerville, D.M.Y. *Analytical Conics,* Bell and Sons, London, 1951.
Struik, D.J. *Lectures on Analytic and Projective Geometry,* Addison-Wesley, Cambridge, MA, 1953.
Svensen, C.L. *Drafting for Engineers,* 2nd ed., Van Nostrand, New York, 1935.
Thorpe, M.F., and Duxbury, P.M., eds. *Rigidity Theory and Applications,* Kluwer Academic/Plenum, New York, 1999.
Thulasiraman, K., and Swamy, M.N.S. *Graphs: Theory and Algorithms,* Wiley, New York, 1992.
Tylove, R.B., and Requicha, A.A.G. Closure of Boolean operations on geometric entities, *Computer-Aided Design,* Vol. 12, No. 5, pp. 219–220, 1980.
Weyl, H. *Symmetry,* Princeton University Press, Princeton, NJ, 1952.

Index