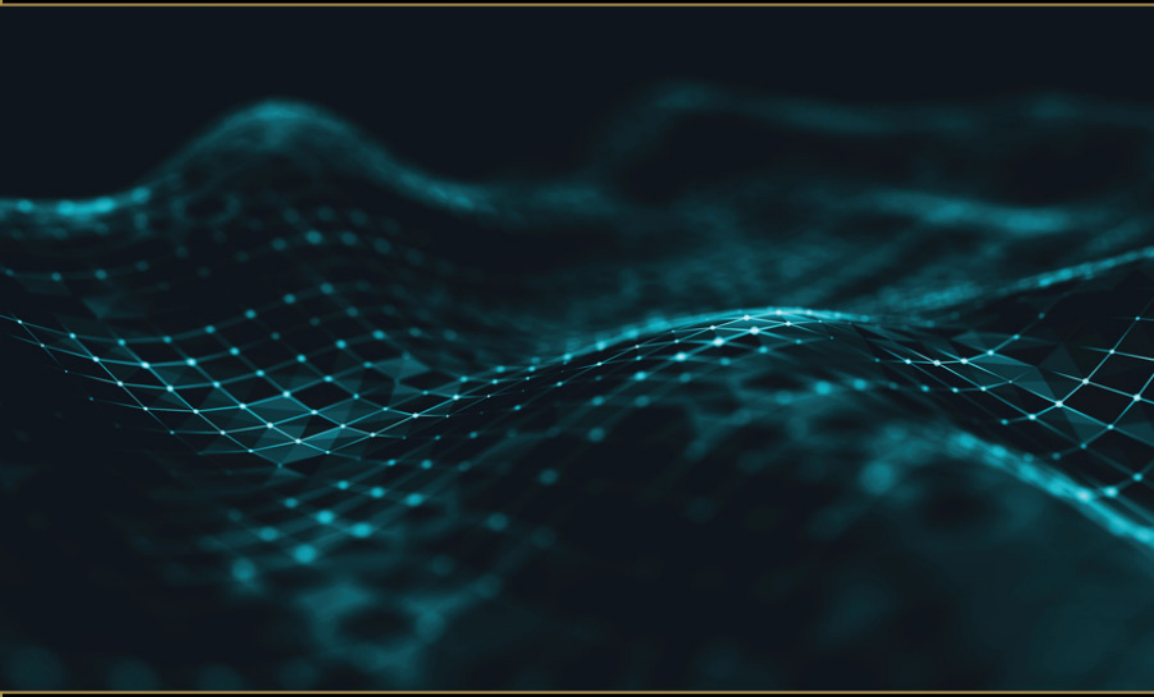
COMSOL®
MULTIPHYSICS GEOMETRY
CREATION AND IMPORT



Layla S. Mayboudi

INCLUDES DVD

Multiphysics Modeling Series

GEOMETRY CREATION AND IMPORT

WITH COMSOL MULTIPHYSICS®

LICENSE, DISCLAIMER OF LIABILITY, AND LIMITED WARRANTY

By purchasing or using this book and its companion files (the "Work"), you agree that this license grants permission to use the contents contained herein, but does not give you the right of ownership to any of the textual content in the book or ownership to any of the information, files, or products contained in it. *This license does not permit uploading of the Work onto the Internet or on a network (of any kind) without the written consent of the Publisher.* Duplication or dissemination of any text, code, simulations, images, etc. contained herein is limited to and subject to licensing terms for the respective products, and permission must be obtained from the Publisher or the owner of the content, etc., in order to reproduce or network any portion of the textual material (in any media) that is contained in the Work.

Mercury Learning and Information ("MLI" or "the Publisher") and anyone involved in the creation, writing, production, accompanying algorithms, code, or computer programs ("the software"), and any accompanying Web site or software of the Work, cannot and do not warrant the performance or results that might be obtained by using the contents of the Work. The author, developers, and the Publisher have used their best efforts to insure the accuracy and functionality of the textual material and/or programs contained in this package; we, however, make no warranty of any kind, express or implied, regarding the performance of these contents or programs. The Work is sold "as is" without warranty (except for defective materials used in manufacturing the book or due to faulty workmanship).

The author, developers, and the publisher of any accompanying content, and anyone involved in the composition, production, and manufacturing of this work will not be liable for damages of any kind arising out of the use of (or the inability to use) the algorithms, source code, computer programs, or textual material contained in this publication. This includes, but is not limited to, loss of revenue or profit, or other incidental, physical, or consequential damages arising out of the use of this Work.

The sole remedy in the event of a claim of any kind is expressly limited to replacement of the book and only at the discretion of the Publisher. The use of "implied warranty" and certain "exclusions" vary from state to state, and might not apply to the purchaser of this product.

Companion files also available for downloading from the publisher by writing to info@merclearning.com.

Geometry Creation and Import

WITH COMSOL MULTIPHYSICS®

Layla S. Mayboudi, PhD

Mercury Learning and Information

Dulles, Virginia
Boston, Massachusetts
New Delhi

Publisher: David Pallai
Mercury Learning and Information
22841 Quicksilver Drive
Dulles, VA 20166
info@merclearning.com
www.merclearning.com
800-232-0223

L. S. Mayboudi. *Geometry Creation and Import with COMSOL Multiphysics®.*
ISBN: 978-1-68392-213-1

Library of Congress Control Number: 2019940577

192021321 Printed on acid-free paper in the United States of America.

To those who believe in shining beauty in all of nature's expressions and who manifest it in geometrical harmonies of their creations.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

VARIABLES

a	constant value
a	fit parameter
a	dimension (m, ft, in)
atan	arc tangent operator
A	surface area, area (m^2, ft^2)
A'	constant value
b	constant value
b	dimension (m, ft, in)
b	fit parameter
B'	constant value
c	constant value
c	fit parameter
d	constant value
d	fit parameter
d	dimension (m, ft, in)
C'	constant value
Csc	cosecant operator
Cos	cosine operator
Cotan	cotangent operator
dx	finite difference along the x-coordinate (m, ft, in)
dy	finite difference along the y-coordinate (m, ft, in)

dz	finite difference along the z-coordinate (m, ft, in)
e	error
e	constant value
f	function
f	constant value
F'	angle (°, rad)
g	gravity acceleration (m/s^2)
h	constant value
h	fit parameter
h	dimension (m, ft, in)
i	unit vector
j	unit vector
k	unit vector
ln	natural logarithm operator
log	logarithm operator
r_i	internal radius (m, ft, in)
r_o	external radius (m, ft, in)
$Scale$	scale
Sec	secant operator
Sin	sine operator
u	dependent variable
u	response function
u	axis along the x-coordinate
u_i	dependent variable component along the x-coordinate
u_x	change of dependent variable along the x-coordinate
u_{xx}	derivative of change of dependent variable along the x-coordinate
u_ψ	angular change of dependent variable about the x-coordinate
v	dependent variable
v	response function
v	axis along the y-coordinate
v_i	dependent variable component along the y-coordinate
v_x	change of dependent variable along the y-coordinate
v_{xx}	derivative of change of dependent variable along the y-coordinate
v_φ	angular change of dependent variable about the y-coordinate

w	dependent variable
w	axis along the z-coordinate
w_i	dependent variable component along the z-coordinate
w_x	change of dependent variable along the z-coordinate
w_{xx}	derivative of change of dependent variable along the z-coordinate
w_θ	angular change of dependent variable about the z-coordinate
w_i	weight fraction for content i (kg of content/kg of the total, grains per grains)
x	coordinate along the x-axis
x	distance along the x-coordinate (m, ft, in)
x_i	dependent variable component along the x-coordinate
x_i	length along the x-coordinate (m, ft, in)
x_0	reference length (m, ft, in)
y	fit parameter
y	coordinate along the y-axis
y	distance along the y-coordinate (m, ft, in)
y_i	dependent variable component along the y-coordinate
y_i	depth along the y-coordinate (m, ft, in)
y_0	reference depth (m, ft, in)
z	coordinate along the z-axis
z	distance along the z-coordinate (m, ft, in)
z_i	dependent variable component along the z-coordinate
z_i	height along the z-coordinate (m, ft, in)
z_0	reference height (m, ft, in)

GREEK SYMBOLS

Δ	difference operator
Δt	time step (s)
Δt	time difference (s, min, hr)
Δx	step size along the x-coordinate (m, ft, in)
Δy	step size along the y-coordinate (m, ft, in)
Δz	step size along the z-coordinate (m, ft, in)
φ	rotation angle about the lateral axis, y-coordinate (°, rad)
$\dot{\varphi}$	angular velocity about the lateral axis, y-coordinate (rad/s^2)

$\ddot{\varphi}$	angular acceleration about the lateral axis, y-coordinate (rad/s^2)
θ	rotation angle about the vertical axis, z-coordinate (°, rad)
$\dot{\theta}$	angular velocity about the vertical axis, z-coordinate (rad/s^2)
$\ddot{\theta}$	angular acceleration about the vertical axis, z-coordinate (rad/s^2)
ψ	rotation angle about the lateral axis, y-coordinate (°, rad)
$\dot{\psi}$	angular velocity about the lateral axis, y-coordinate (rad/s^2)
$\ddot{\psi}$	angular acceleration about the lateral axis, y-coordinate (rad/s^2)

SUBSCRIPTS

∞	infinite

GLOSSARY

ACIS	Alan, Charles, Ian, and Spatial
AEC	Architecture Engineering Construction
AM	Additive Manufacturing
APDL	ANSYS Parametric Design Language
API	Application Programming Interface
ASM	Assembly
BIW	Body in White
BMI	Body Mass Index
BOM	Bill Of Materials
CAD	Computer Aided Design
CADAM	Computer-Augmented Design and Manufacturing
CAE	Computer Aided Engineering
CAM	Computer Aided Manufacturing
CATIA	Computer-Aided Three-Dimensional Interactive Application
CDIO	Conceive, Design, Implement, and Operate
CFD	Computational Fluid Dynamics
CLIP	Continuous Liquid Interface Production
CME	Coronal Mass Ejection
CMM	Coordinate Measuring Machine
CSG	Constructive Binary Solid Geometry
DOS	Disk Operating System

DRW	Drawing
ECAD	Electronic and Electrical Computer-Aided Design
EDM	Electronic Distance Measurement
EES	Engineering Equation Solver
FDM	Finite Difference Method
FEM	Finite Element Method
FVM	Finite Volume Method
GPS	Global Positioning System
GUI	Graphical User Interface
HG	Heat Generation
I/O	Input/Output
ISS	International Space Station
KISS	Keep It Simple, Stupid
KML	Keyhole Markup Language
LED	Light Emitting Diode
LEED	Leadership in Energy and Environmental Design
LL	Low Lead
MBSE	Model Based Systems Engineering
MCAD	Mechanical Computer Aided Design
ME	Material Extrusion
MEMS	Micro-Electro-Mechanical Systems
MIT	Massachusetts Institute of Technology
Muda	Futility, Uselessness, and Wastefulness
OAI	Object Action Interface
ODE	Ordinary Differential Equations
OpenCL	Open Computing Language
OpenMP	Open Multi-Processing
PBF	Powder Bed Fusion
PDE	Partial Differential Equation
PLM	Product Lifecycle Management
PRT	Part
Revit	Revise-Instantly
RPV	Remoted Pilot Vehicles
RTK	Real Time Kinematic
SLD	Solid
SMART	Specific, Measurable, Attainable, Relevant, and Timely

ST	Synchronous Technology
TEL	Tetraethyllead
TIMWOODS	Transportation, Inventory, Material, Waiting, Over-production, Over-processing, Defects, and Skills
UV	Ultraviolet
VP	Vat Photopolymerization
VR	Virtual Reality
WAC	World Aeronautical Charts
XML	Extensible Markup Language
4D BIM	Four-Dimensional Building Information Modeling
5M	Material, Machine, Method, Measurement, Man, and Money
5S	Sort, Simplify, Shine, Standardize, and Sustain

PREFACE

An outline of this work is presented here—a road map of the journey. The chapters are designed so that they are most comprehensible if followed in the order the material is presented. In addition to dealing with the specifics of how geometry can be created for the purpose of engineering analysis, this work takes a wider outlook on the world of geometry, both of today and yesterday. It shows how the geometry is fundamental to nature and how it is interconnected with the world around us. The historical background provided shows the progression of the geometrical science from the dawn of civilization to the multiple specialized disciplines that the geometry has evolved into today. The world of geometry is a vast ocean and only a few drops of it are presented here. While the work touches upon the complex nature of geometry's axioms and theorems, the presentation aims to be accessible to all technically minded readers.

From the original conception, this publication was intended as a companion for the Author's earlier work—*Heat Transfer Modelling Using COMSOL: Slab to Radial Fin* [1]. Thus, the contents of this publication are complementary to the earlier one, with minimum overlap between the two. This also means that most of the examples used in this work are the same as in the earlier one, providing the reader with opportunity to learn about model creation in this publication and to continue to learn about the analysis step by working with the earlier book.

Chapter 1 is a brief introduction to the world of geometry. The chapter highlights how the geometry surrounds us in all possible forms, in the natural environments and in the artificial ones. It is a science of coordinates, relative locations, and borders that forms part of the worldview that we develop. Some background is provided on the advancements in this field by humans (and other living creatures) from the beginning of creation to the modern time. Our non-human cohabitants of this planet either contributed directly

to this cause (e.g., building honey combs) or indirectly as sources of human inspiration (e.g., dragonfly wings). Honeycomb's hexagon is constructed to the highest level of perfection with no material wasted after the process completion—demonstrating the honeybee's strictest possible adherence to the Lean Six Sigma process. Highly corrugated and stiff dragonfly wings have been mimicked to create flapping micro air vehicles producing thrust and lift by flapping the wings that deform like sails [2,3].

Chapter 2 presents a historical perspective on the progress in geometrical sciences. The developments are organized by time, being grouped into ancient, medieval, and modern periods. Other historical developments covered include information on the subjects of measurement methods and units of measurement. Progression of geometrical concepts from the fundamental to the most complex ones is described along with the interconnections between the geometrical hypotheses.

Chapter 3 talks about connections between creative arts and the science of geometry. Concepts of form, fit, and function, part of the Lean Six Sigma methodology which aims to improve processes, are connected in this work with the geometrical concepts. As part of this concept development, travel dimensions and the archaic Earth's flatness theories are discussed. Interestingly enough, the idea of the Earth's flatness seems to still have its proponents today, demonstrating how even the most obvious realities may still be questioned by some. Examples are provided of geometrical connections to the works of great creators such as Leonardo Da Vinci and to the ones associated with the natural elements of the universe. Design optimization is discussed in this chapter as part of the topology, size, and shape concepts.

Chapter 4 discusses the elements of geometry. These include the definitions such as axioms, topology, and symmetries along with the basic elements used in geometrical constructions such as lines, angles, curves, planes, surfaces, and volumes. Concepts such as topology and symmetry are explained as well.

Chapter 5 discusses the methods to navigate a geometry. Since geometrical coordinates of vertices need to be defined relative to some common frame of reference (point of origin), it is paramount to have such point identified and also define the coordinate system in which their connectivity becomes meaningful. For example, although it may be possible to locate the points on the perimeter of a circle by means of the Cartesian coordinate system using two linear orthogonal coordinates, it is considerably easier to do this using a polar coordinate system by means of the radius and the angle. An overview of the vector algebra is presented at the end of this chapter since knowing its fundamentals is a must to fully understand the examples provided.

Chapter 6 discusses the methods by which geometries are created using commercial software packages. Solid Edge®, PTC Creo®, Autodesk, CATIA™, and SolidWorks® are briefly introduced, highlighting their origins and features. Two main products of MathWorks®, MATLAB® and Simulink, are presented, with the focus on MATLAB and its capabilities to create geometry by not only dragging-and-dropping the pre-designed shapes but also using commands,

formulae, and scripts—with the capability to implement each method of geometry creation as an input file that can be executed like a program. A mapping toolbox, an add-on to MATLAB, is introduced showing how this tool can be used to create custom maps using different projections.

Chapter 7 presents the methodology by which the model geometry can be imported into Finite Element Method (FEM) software, either by using commands or Graphical User Interface (GUI). Challenges which the user may face before, during, and after performing this process (CAD-FEM interaction) are discussed and the recommended methodologies are addressed. One of the commercial FEM tools available is COMSOL Multiphysics®, which is employed in this work for the majority of the geometry creation examples. FEM (and CAD) tools often share common creation, presentation, and organization techniques. Therefore, geometry creation and modeling fundamentals presented in this chapter can be extended to other similar tools. Interfacing tools that are either part of COMSOL Multiphysics core package or add-on modules are presented as well—with the Computer Aided Design (CAD) import module receiving most of the attention.

Chapter 8 focuses on using COMSOL Multiphysics to generate the geometry, and shows how a model in this FEM environment is created. As mentioned earlier, most of the examples used in this work are created using COMSOL Multiphysics add-on CAD module, which requires a dedicated license; however, the core package also provides the basic tools to create geometries, using geometrical elements introduced in Chapter 2 (lines and arcs) and, for example, Boolean operations. Three-dimensional (3D) geometries can be created from the two-dimensional (2D) profiles by extruding these profiles linearly, sweeping them along any curve, or revolving them about an axis. Geometry creation is the first step in any analysis. Next steps are defining physics, materials, analysis types, calculating a solution, and finally post-processing the solution results to extract useful information from them.

Chapter 9 provides case studies with an increasing level of complexity from a slab to radial fin. These shapes, known as *extended surfaces* in the field of heat transfer, have been selected due to their broad range of applications, from bridge construction to furniture, to their use for thermal management—employed in applications from cooking to electronics and aerospace. Variety of shapes are discussed herein so that the reader can review each and decide if they can address some of the design challenges they face. The Webbed radial fin geometry is the most complex one and although it could have been created in a dedicated CAD tool such as Solid Edge, it is created using the FEM CAD module to show what can be done when a dedicated tool is not available and also to highlight the FEM tool geometry creation features. The effort to use this methodology increases the complexity of the task; however, it also facilitates interacting with the FEM tool when setting up the physics and boundary conditions by eliminating any potential misinterpretations that may happen as a result of the geometry import. A *rotini* fin, another structure with complex

geometry, is introduced in this chapter; it is created in a dedicated CAD tool (Solid Edge) and then imported to the FEM tool. A relatively detailed description of the steps taken in order to generate the geometry model, especially the complex ones, where parametric surfaces are extensively used are discussed along with related figures.

Chapter 10 offers an overview of the applications that were created based on the geometry models presented in Chapter 9 ("Extended Surfaces") and that are included in the companion files to this book. The same files also include COMSOL Multiphysics model files, containing only the geometry models. Applications in this context are a feature provided by COMSOL Multiphysics that allows the analyst to create a user-friendly interface to the problem, providing only a limited number of input controls and output displays. The applications presented here provide the capability to vary some basic geometry and mesh parameters and display the results. A brief overview of the model and application along with the images of the meshed geometry and the application interface are presented for each case study. The purpose of this work is to provide initial guidance for the reader to understand the geometry creation process in a FEM tool, interact with it, review the content on their own, and attempt the examples. Building on these steps, the reader can come up with their own ideas to set up the geometry (or mesh), and identify the steps to be taken to complete the analysis and obtain the solutions.

Chapter 11 talks about *good practices*—the concepts of *recommended* methods versus *preferred* ones are clarified. It is emphasized that there is usually more than one method when creating geometries. The purpose is therefore to choose among the possible methods the one that is both most efficient and most appropriate for the available FEM tool. Several approaches are provided so that the reader can look into the possible solutions when issues occur. There are also examples of how the human mind can be inspired and the nature becomes the source of inspiration when creating geometrical features in the form of ecosystems or places of residence and solace. The steps to be taken before, during, and after geometry import to the FEM model as well as interacting with the imported geometry by specifying tolerances, and under what conditions to relax the tolerance are presented.

Chapter 12 talks about the Lean Six Sigma concepts and how they can apply to this field. The chapter explains how all levels of the product lifecycle, including interactions between the design tool and the designer (e.g., geometry creation) should be considered when process improvement decisions are made. The concept of entitlement is presented as well and what it means to be *entitled* to something but not fully realizing it. It talks about the sources of waste and the approaches to take in order to make more responsible decisions to reduce waste to the extent possible. Furthermore, it is explained what it means to be operating within 99 percent full potential and if this capability is acceptable or considered an excellent or poor performance under the given circumstances. These concepts are expanded upon with regard to geometry creation, importing, defeaturing, and processing within the CAD-FEM environments.

Chapter 13 concludes this work, encouraging application of fair assessment in terms of processing or constructing geometries when initiating the challenge of making geometrical features so that it is effectively employed within a CAD or FEM tool. Much can be learned by studying, but doing must always accompany it.

Appendix A lists COMSOL Multiphysics parameters and geometry sequence steps used to generate a Fibonacci spiral (an approximation of the golden spiral) for the illustration shown in Chapter 4.

Note that there is not a single way to approach geometry creation; neither are there single approaches to create relationships between the components. For example, a part may be rotated to the desired orientation before importing it into an assembly. It is also possible to import the same part into the assembly and then rotate it to the desired orientation. In the latter scenario, mating between certain geometrical features; for example, main coordinates or construction lines may be endeavored to arrive at the same conclusion. Therefore, the Author encourages the reader to attempt the *Examples* and *Case Studies* after completion of the related subject matter more than once, exploring other possibilities to build the same geometry using the knowledge building blocks provided. Moreover, the Author encourages the reader to review the suggested approaches to tackle the geometrical problems both to visualize and understand them, but also to construct them as individual parts first, and then use computer-aided design tools to create the parts as single components or in assemblies only after the conceptualization process has been completed.

ACKNOWLEDGMENTS

My family, my teachers, and my publisher—I am infinitely grateful to them all for their generous support and the positive influence on my life that they have had.

They guided me during this long voyage by improving upon my insights and making me envisage the new possibilities. They did all that with respect and love and so they are remembered as this work is being completed.

I cannot fully express the depth of my gratitude toward these individuals whose sacrifices and whose kind, polite, patient, and determined attitudes have been inspiring.

The world of scientific publication entails challenges but also rewards. This book, from its conception to publication, underwent a long journey.

To all those who prepared me for this journey and who supported me along the way:

There are some who raise you up to be the person you aspire to be…
There are some who put divine fire in your heart to create and to succeed…
There are some who nurture you to spread your wings and fly the wild blue yonder…
There are some who bring heavenly hope back to you when you are hopeless…
There are some who encourage your confidence, dignity, and humanity so you rise after you fall…
There are some who spread love and respect as they stride on the Earth…
There are some who stand by you in adversity, when they are needed most...
And where are they, the best among the best…?
How much longer are we to wait, how many more breaths, seconds, days, weeks, months, and years...?

CHAPTER 1

INTRODUCTION

The word *geometry* comes from the Greek words *geo* (the Earth) and *metron* (measurement) [4]. It is the science of points, lines, curves, shapes, positions, and boundaries. The origin of geometry as a tool for the Earth measurement can be traced back to ancient times when it was applied to land surveys, which were needed for collecting taxes and for laying out the plans of great structures. The geometrical techniques were used to measure a piece of land and also to locate points with respect to a reference. For example, the circle of Stonehenge in the United Kingdom was laid out using pegs and ropes. Similar techniques were reportedly employed in the construction of the pyramids of Giza in Egypt. Ancient Egyptians knew how to calculate the areas of geometric shapes such as circles, triangles, and rectangles and volumes of pyramids and cylinders. They considered the squares and triangles as sacred geometries and used them for construction of places of worship.

As humans, we both appreciate shapes and create new shapes, be it for art or for practical use, often combining both in one object. We also move in the world of geometry, traveling from point to point, along straight lines or curved trajectories, while using the positions laid out on great circles of latitudes and longitudes, on our planet and beyond. We learn to recognize shapes from the earliest age—the triangles, rectangles, and circles. In my own childhood, I spent many an hour building fire trucks, spaceships, or houses with Lego blocks: a two-by-four brick to use for the window mantle, a four-by-twelve brick to make the bench on the porch, or a two-by-three brick sloped 45 degrees to use over the window to provide shade. It was the brick shape that mattered most,

not the color. Some claim that we even have subconscious emotional responses to shapes: sharp corners are threatening; circular, rounded shapes are soothing. Perhaps this is why we find the flowing organic shapes of the twentieth-century Spanish architect Antoni Gaudi so pleasing, his unique architectural genius most famously expressed in the building of the Sagrada Familia cathedral in Barcelona.

The child who created the Lego block structures grows up to design real buildings, trains, aircraft, spaceships, or maybe to write thought-provoking books that may then inspire others to see a different world, such as the works of a genius like Jules Verne, a French novelist, poet, and playwright. In his famous work, *From the Earth to the Moon*, written in 1865, he tells the story of a group of ambitious individuals who attempt to build a Moon-landing projectile. Despite a lack of mathematical data at the time, some of the figures he presents are very accurate. It was only on July 20, 1969 that Apollo 11 reached the Moon's orbit and human walked on its surface for the very first time—a century was needed for Verne's vision to be realized [5,6,7].

Looking around oneself, one notices lines, curves, surfaces, volumes of all shapes, and angles. Artisans of the Renaissance and medieval Europe employed concepts of geometrical features that were designed using formulae that came from the Persian, Greek, and Egyptian pre-Renaissance mathematicians and polymaths. A definition for sacred geometry was developed as a result of this human curiosity, where the ratios of the numbers are two consecutive integers (e.g., 1/2, 2/3, and 3/4); it is shown that following this pattern results in the most harmonious and pleasing structures. Applying this principle to music also results in harmonic series that may be heard in many of the masterworks by classical Western composers such as Johann Sebastian Bach and in Eastern music such as that originating in Tibetan temples. Johann Wolfgang (von) Goethe, an eighteenth-century German author, philosopher, and statesman, said that music is fluid architecture. Another famous sacred proportion is known as the *Golden Mean*, or *Golden Section*, in which the ratio of the whole $(a + b)$ to the larger part (a) equals the ratio of the larger part (a) to the smaller part (b): $(a+b)/b = b/a$. Solving that equation yields a ratio of $(1+\sqrt{5})/2 = 1.618$—a number widely used in many historical artistic and architectural creations. This is the basis for the major six (3/5) and minor six (5/8) ratios that are found in many natural marvels. People have commented that they experience harmonious feelings when inside a sacred environment, where the architectural features follow the *Golden Mean* rule [8,9,10].

Geometry has been employed in many forms of divine expressions seen in manuscripts, decorative graphics, and statues as tributes to deities and divinity. For example, *Book of Kells*, created circa 800 AD

in Ireland, is a gospel book written in Latin in the form of an illuminated manuscript (Figure 1.1). This manuscript is a masterwork of Western calligraphy, embedding geometrical designs within the text and the images, filled with lavishly decorated pages. This work used insular script, which is the medieval script system developed in Ireland. Insular ornamentation applied to the monograms, images, and texts are architectural elements of the work with evangelist symbols claiming the space around the arches [11].

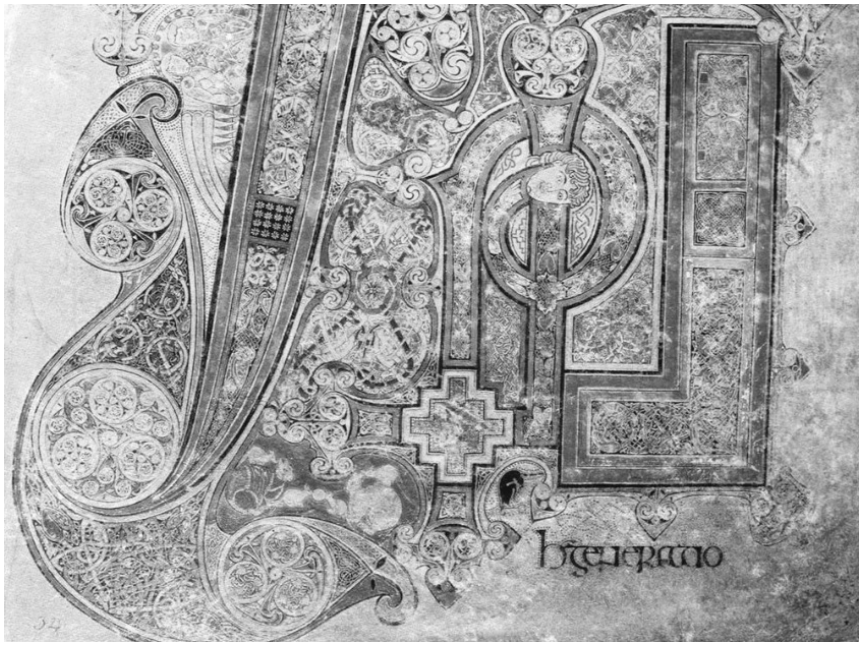


FIGURE 1.1. Geometrical patterns in *Book of Kells*, the ninth century, Ireland, Scotland, and England. Folio 34r contains the Chi Rho monogram, the first two letters of the word *Christ* in Greek. Reproduced image is a portion of the original image that is licensed under the public domain [12].

Mosaic, which is an art of assembling glass and stone tiles into decorative patterned images, is another example of art rooted in the Persian Sasanid Empire of 224 AD; origins of the *non-patterned* tiles go back to the second half of the third millennium BC in Mesopotamia. These small colored pieces, also known as tesserae, were widely used throughout the Byzantine empire from the sixth until the fifteenth century and onward, finding their way to palaces of the Norman Kings, where hunting themes of animals and landscapes decorated the walls and ceilings. This secular mosaic art was inspired by the work of the Persian (Iranian) artisans with patterns that are traceable to the Persian (Iranian) rugs and poetry books, displaying the art and culture of the kings, heroes,

and love for the divinity in Persia (old Iran). Pre-Islamic Persia (Iran) was interested in illustrating images of living things in the Persian (Iranian) mosaic art; after the advent of Islam, some of these images had to be replaced with geometrical designs in any drawings and architecture used in a religious context. Thus, the artisans' creativity was redirected to conception and realization of highly complex patterns in architectural features (Figure 1.2) [13].

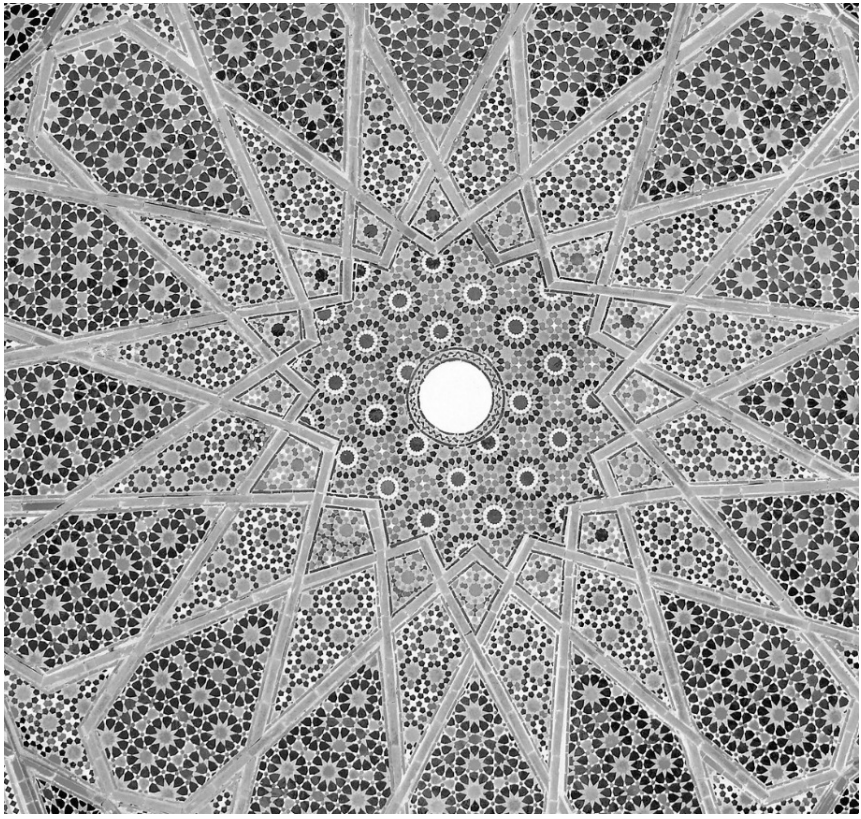


FIGURE 1.2. Enameled tile mosaic, Hafez tomb roof, the eighteenth century, Shiraz, Iran. The reproduced image is a portion of the original image that is licensed under the Creative Commons Attribution 3.0 Unported license, used with permission, author: Pentocelo [14].

The arabesque decorations, which are repeating intricate patterns often depicting leaves, tendrils, or plain lines, are seen in much of the Persian (Iranian) architecture (Figure 1.3). They are employed, for example, in the decoration of stucco muqarnas vaults, also known as *Ahoopy* or *honeycomb*, that were installed under the roofs to make the wall-to-dome transitions as smooth as possible. This architectural design originated in Persia (Iran) in the mid-tenth century and was exported to other parts of the world by expert Persian (Iranian) artisans and is seen

in many monumental Islamic structures such as the Alhambra fortress, located in Granada in the Andalusia region of Spain. The muqarnas is a symbolic representation of the complexity and organization of the world created by the omnipotent and omniscient God, depicting the precise connection between its detailed substructure and the visible world—its dome represents the heaven above, and the infinity power is installed over the entry portals. This symbolizes the human spirit connecting to the creator—showing eternal life with the downward hanging geometrical shapes (imagine a stalactite or an icicle), depicting the presence of the almighty in the physical world [15,16].

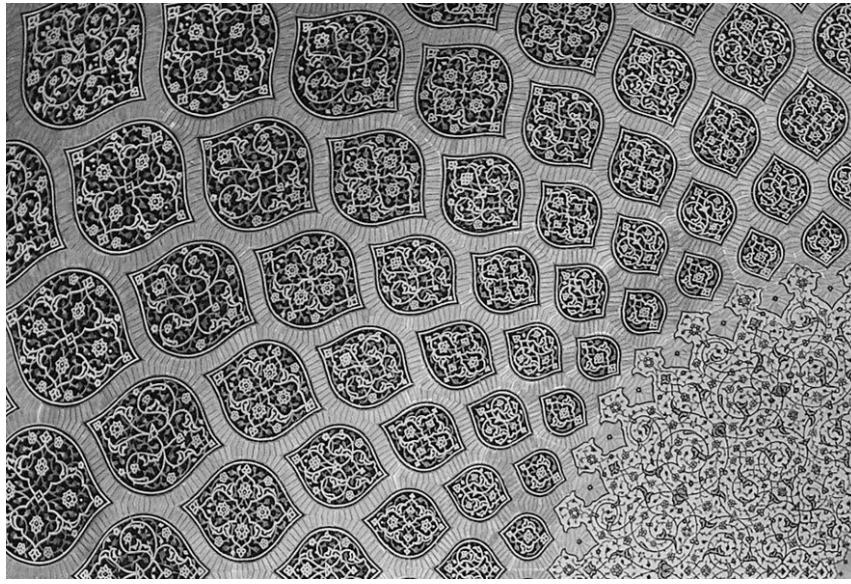


FIGURE 1.3. Arabesque design, interior dome of Sheikh Lotfollah Mosque, the twentieth century, Isfahan, Iran. The patterns get smaller as one approaches the center from any of the corners, depicting use of the *scale* feature in geometrical expressions. This image is a portion of the original image that is licensed under the Creative Commons Attribution 3.0 Unported license, used with permission, author: Phillip Maiwald [17].

Patterns and geometrical features may be found anywhere in nature. They may be used in self-defense and in reproduction. Think of a peacock expanding his tail into a fanlike structure to show his potency to potential female partners. The hen looks for the highest number of eyespots on the male's tail to decide if he will be a good father for her chicks. The chicks from this heavily decorated father are also heavier, resulting in better adaptability and higher survival rates. This example shows the substantial influence of geometrical features in the animal world, affecting all three main elements of mating success, creating offspring, and survivability of the species [18].

CHAPTER 2

SCIENCE OF GEOMETRY THROUGH THE AGES

The word *geometry* has several meanings: (1) a division of mathematics that is related to points, lines, planes, and higher order elements, their situation in space, and their relation to each other; (2) a specific coordinate system; and (3) the shape of subcomponents and their relation to each other as part of a more complex component [19]. The mathematician or *geometer* works with geometry features such as shape, size, relative position of the geometry components, the reference of origin, and how the parts individually and as a whole are situated in space. Geometry is the science of characterization of the shapes of any form. Although any creative work is a piece of art, if done following proportions and geometrical concepts, this artwork interweaves with the science of geometry, demonstrating how art and science can influence one another. Geometry is a science with a history that begins as early as the second millennium BC. Art, mathematics, physics, chemistry, architecture, and engineering all use geometry and related theories [20].

In its contemporary form, geometry can be subdivided into the following:

(1) Euclidian geometry—study of geometrical elements such as points, lines, angles, planes, solids;
(2) analytic geometry—study of geometry that employs coordinate systems, with applications in modern mathematics such as crystallography;
(3) differential geometry—interface with calculus and linear algebra with applications in physics such as relativity theory;

(4) topology—mapping of objects without changing their proportions; study of the objects that focuses on the connectedness (all in one piece) and compactness (inclusion of the features in a subset of a body);

(5) convex geometry—study of convex sets for which all linear combinations of point coordinates belonging to a region fall within the same region; for a 2D space, it is equivalent to drawing a straight line from point A to point B and requiring that all points on that line remain within the same region;

(6) algebraic geometry—study of the zeros of multivariate polynomials (having more than one zero); a zero or a root is any value of x that makes the equation $f(x) = 0$ true;

(7) discrete geometry—study of the location of geometric features such as points and lines with respect to each other, also known as combination, consisting of the methods of arrangement of the geometry components and their configuration, while satisfying certain criteria, the method of their construction and optimization, and the best solution for the given conditions; and

(8) computational geometry—study of algorithms expressed in geometrical terms, with applications in medical image processing. Recall the motion picture *Arrival*, where a linguist and a mathematician are hired to decode the language of alien visitors. The aliens communicate by shapes displayed on a transparent wall, separating them from their visitors. In the movie, these unknown shapes are analyzed and decoded based on the responses received at different times [21].

2.1 PRE-HISTORY AND ANCIENT HISTORY

While geometry has evolved significantly throughout human history, there are some general concepts that are more or less fundamental to geometry which are shared among different schools of thought. These include the concepts of points, lines, planes, surfaces, angles, and curves, as well as the more advanced notions of manifolds and topology or metrics. They have found records as early as the second millennium BC of geometrical features used by ancient Mesopotamians and Egyptians. Growth of knowledge in this area was based on empirical findings—lines, areas, volumes, and angles were needed to identify the size, location, or proportions of bodies such as a piece of land, a house, or the location of stars.

Nubian architecture dates back to 6000 BC, being contemporary with Jericho, the walled town in Palestine. Among the numerous

architectural features in this region is the third Nubian pyramid site Meroë, which has great archeological value. The ancient Nubians are famous for founding a geometrical system that includes Sun clocks, an example of which is located at Meroë. It is also believed that they used triangulation techniques similar to those of Egyptians during the Meroitic period [22].

Currently held at Columbia University in New York, a Babylonian clay tablet (Plimpton 322) written between 1800 and 1600 BC includes an example of Babylonian mathematics. This tablet consists of a table of four columns and fifteen rows of numbers in the cuneiform language. This table lists two of the three numbers in what are now called the Pythagorean triples—positive integers a, b, and c satisfying relation $a^2 + b^2 = c^2$; the third number was meant to be filled in as an exercise. It is said that the author of this tablet was a scribe, developing the problem as an example for teaching students at his school, knowing that this problem is so widely used in the world of geometry and mathematics. This Babylonian clay tablet used trapezoid procedures to compute Jupiter's position and motion over time. Similar geometrical procedures were used centuries later by a group of philosophers in the fourteenth century working in Merton College in Oxford. They became known as *Oxford Calculators*, as they took the logical-mathematical approach to philosophical problems. One well-known example is their proof of the mean speed theorem, which states that a body that uniformly accelerates from rest (i.e., zero initial velocity) travels the same distance if it moves at a constant speed with a magnitude that is half of the speed of the body after it is accelerated (acceleration by time) [23,24].

Ancient Egyptians had a system for documenting the formulae used for the geometrical shapes employed in constructing the pyramids. Their data (formulae) management system consisted of a three-column table presenting problem definition, its source or where it was used, and equations. They also employed a similarity concept when deriving the geometrical relations. This is seen in the expressions using the ratio of run over rise, which the ancient Egyptians called a *seked*. Seked is proportional to the inverse of what we call today a *slope* or *gradient*. Using the seked and the base length, they were able to calculate the increase in height along any inclined line.

The geometric knowledge of the ancient Egyptians has been recovered from translations of documents known as mathematical papyruses. For example, the Rhind Mathematical Papyrus, currently residing in the British Museum in London, dates back to 1550 BC; it was found during an excavation in Egypt and purchased in 1858 in Luxor, Egypt, by Scottish antiquarian Alexander Henry Rhind. Also kept in the same museum, the Rhind Egyptian Mathematical Leather Roll consists of

a row of twenty-six added unit fractions resulting in new unit fractions. This work was copied by the scribe Ahmes—dating back to King Amenemhat III (the twelfth dynasty). It is 33 cm (13 in) wide and over 5 m (16 ft) long. The translation started in the nineteenth century and is still not fully completed. The introduction claims the text to be an accurate estimate for inquiring into things, the knowledge of all things, mysteries, and secrets [25].

Another ancient Egyptian document, known as the Moscow Mathematical Papyrus (or the Golenishchev Mathematical Papyrus), also dates to the twelfth dynasty of Egypt (approximately 1850 BC) and was purchased in 1892 in Thebes by Russian Egyptologist Vladimir Golenishchev. It currently belongs to the Pushkin State Museum of Fine Arts in Moscow. It varies between 3.8 and 7.6 cm (1.5 and 3 in) in width and is 5½ m (18 ft) long. In 1930, its contents were categorized by Soviet Orientalist Vasily Vasilievich Struveto into twenty-five distinct problems with solutions. The topics covered include the calculation of areas for a triangle and a hemisphere as well as the volume of a truncated pyramid—also known as a *frustum* [26].

After the developments in Egypt, the ancient Greeks made the greatest strides in furthering humanity's geometrical knowledge. Their work was not advanced further until the second half of the first millennium in the Middle East. Working in the sixth century BC, a mathematician known as Thales of Miletus derived by deduction a Thales theory (Figure 2.1). This hypothesis states that if points A, B, and C located on the circumference of a circle are connected by lines, a triangle is formed—providing that the line connecting the two points (A and C) passes through the center of the circle (O)—where the third point (B) is at the vertex of a 90-degree angle—meaning that the lines (AB and BC) are perpendicular. The importance of this observation in navigation and architecture cannot be overstated—its use has been reported in calculating the height of pyramids and in approximating the distance of a ship from shore.

$$\because \theta = \phi \; and \; \psi = \gamma$$

$$\therefore \begin{cases} 2\phi + \alpha = 180^\circ \\ 2\psi + \beta = 180^\circ \end{cases}$$

$$\therefore 2(\phi + \psi) + (\alpha + \beta) = 360^\circ$$

$$\because \alpha + \beta = 180$$

$$\therefore 2(\phi + \psi) = 180^\circ \text{ and } \phi + \psi = 90^\circ$$

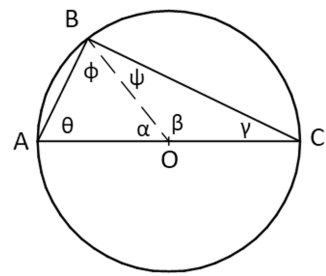


FIGURE 2.1. A proof of the Thales theorem (drawings created in Solid Edge).

Thales employed both theory and practice in order to prove his geometrical hypotheses. His work was mainly concerned with the position of things such as points, lines, planes, and solids and their relationship expressed with angles and distances between the entities. Thales was aware of the seked, the concept developed by the Egyptians, representing the ratio of run over rise. It is reported that he measured the height of the pyramids, by means of triangular concepts such as similarity and rightness, using their projected shadows. This was done by comparing to his own shadow during the time of day when his shadow was as tall as him. He concluded that a right triangle with equal sides also has equal angles opposing the equal sides. Therefore, the length of the shadow measured from the center of the pyramid equals its height. He used a similar technique to identify the distances of ships. This was done by inserting a stick in the ground, with another one perpendicular to it in a horizontal position, and the third one measuring the sight, using the seked concept, and this is what is known today in navigation as the line of sight. Thales's intercept theorem is the proportionality theorem that expresses a relationship between the segments of the intersecting lines that cross two parallel lines (Figure 2.2). For example, the ratio of AB to AD in the figure equals that of BC to DE.

Thales also stated and proved a number of other fundamental geometrical theorems:

(1) a diameter bisects a circle (i.e., divides it into two equal parts);
(2) the two base angles of an isosceles triangle are equal; and
(3) the two vertical angles are equal. Vertical angles are the opposite angles formed when two straight lines intersect (such as β and β' in Figure 2.2). He made this proof based on his observation during a visit to Egypt that Egyptians would always ensure that the vertical angles were the same when laying out two crossing lines.

To prove the last theorem, he stated two additional basic assumptions: (1) all straight (90-degree) angles are equal; and (2) equal quantities added to or subtracted from other equal quantities were also equal [27].

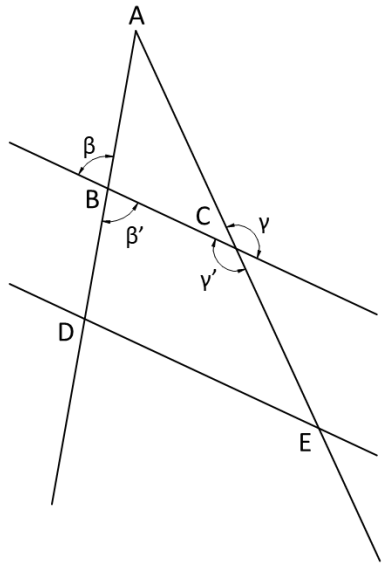


FIGURE 2.2. A demonstration of Thalesís intercept theorem.

Pythagoras, who lived in the sixth century BC, can be counted among the most famous and influential Greek philosophers. His teachings in the fields of politics, religion, and science inspired generations of Greek philosophers and through them influenced Western thought ever since. Around 530 BC, he established a school in the south of Italy where a Greek colony existed at that time. The Pythagorean school of thought encouraged its followers to lead a life adhering to intellectual pursuits, even expecting them to dress in a particular way. For example, their code of ethics prohibited use of beans, legumes, and meats. It is not known for certain why such rules were created, but there are stories suggesting that it was thought that the generated gas leaving the human body could also take away part of the soul.

The Pythagorean theorem is among the most important relations in Euclidean geometry. It states that, for a right-angle triangle, the square of the hypotenuse (a, the side that faces the 90-degree angle) equals the sum of the squares of the two remaining sides (b and c)—$a^2 = b^2 + c^2$. Even though the relationship itself was well-known long before him, it is believed that Pythagoras was the first one who developed a formal proof of this theory. Other cultures, such as Mesopotamian, Chinese, and Indian, also independently arrived at the idea. Proof of the Pythagorean theorem is based on rearrangement of triangles within a square. Finding another proof of this theorem has fascinated mathematicians through the ages, and many alternative proofs have been found using geometric, algebraic, and other approaches.

Plato was another important figure in the history of Greek intellectual development and whose ideas were influenced by the teachings of Pythagoras. Plato initiated the Platonic school of thought, which was taught in his academy founded in Athens in 387 BC. He was a student of Socrates, and his own most famous disciple was Aristotle. Plato's philosophy was formed based on his dialogs with Socrates (as described in his famous work *The Republic*). Aristotle, the Greek philosopher and scientist, studied and taught in this school for over twenty years before establishing his own school of thought, the Lyceum became a place where students learned rhetoric, martial arts, and philosophy; they used the arts of mathematics, logic, and articulation to present hypotheses and address common misconceptions, to argue for and against each, and to advocate for and against the theories simultaneously, in a respectful manner and environment, so that the opposite views were heard and eventually theories were accepted, turned down, or conditionally accepted based on their merits. Detailed observation, collaborative research, and documentation of findings are the methodical approaches introduced and perfected during this period. Today, any candidate who obtains a doctoral degree is given a title of a Doctor of Philosophy (PhD), no matter if the degree is in linguistics, physics, or engineering. This connection between science and philosophy is rooted in ancient Greece, where the methodology adopted to prove theories is the same as the one employed today to attain a PhD. The idea of a formal debate is also rooted in this ancient tradition, though these roots are only imperfectly remembered today, as witnessed in the occasional lack of civility of modern debates, where emotions are often allowed to reign over reason [28,29,30,31].

The value of what we call *Pi* (i.e., a Greek letter π), a constant equal to the ratio of a circle's circumference to its diameter, has been known since ancient times, and precision of its calculation has progressed with our knowledge of geometry in general. In the fifth century BC, Antiphon of Rhamnus, the earliest of the ten Attic orators, and who some historians claim is the same person as Antiphon the Sophist, was the first to establish the upper and lower limits on the value of *Pi*. He accomplished this by recognizing that the circle's area must be less than the polygon which circumscribes the circle but greater than the area of the polygon inscribed in the circle. Increasing the number of polygon sides gives progressively better approximation (Figure 2.3). Eudoxus of Cnidus, a Greek astronomer and mathematician of the fourth century BC, further developed and applied the previous technique to prove that the ratio of any two circle areas is equal to the ratio of the squares of their radii and that the ratio of sphere volumes is equal to their radii cubed. This method of calculating areas by successive approximation

with polygons was given the name of *method of exhaustion* in 1647 by Gregory of Saint Vincent. This method is considered a precursor of the techniques later applied in calculus—it is based on the concept of a *limit*: as the number of polygon sides is increased to infinity, the polygon area will approach that of a circle [32,33,34].

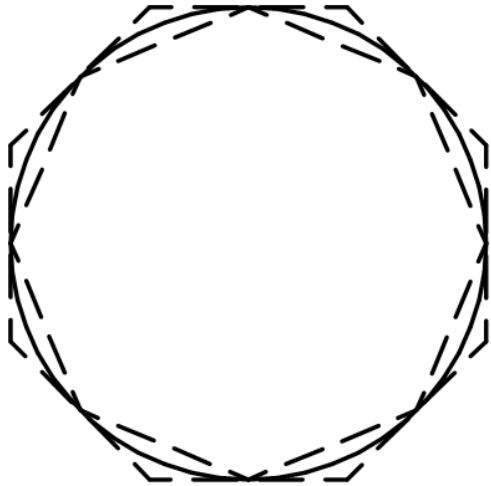


FIGURE 2.3. An octagon circumscribing a circle and also inscribed in a circle (drawings created in Solid Edge).

The method of exhaustion was also employed by Archimedes of Syracuse, a Greek mathematician, physicist, engineer, inventor, and astronomer in the third century BC, to estimate the area under the arc of a parabola with a three-decimal digit approximation for *Pi*. Archimedes was able to advance the precision of *Pi* calculation considerably by going up to a polygon of ninety-six sides to estimate *Pi*. In his approach, he started with a regular hexagon inscribed into and circumscribing the circle and was estimating its perimeter. He established a recursive procedure, explaining how to calculate the perimeter of a regular polygon with twice as many sides, and then followed it up to the ninety-six-sided polygon. This approximation established *Pi* to be between 223/71 and 22/7 (which is accurate to between 2×10^{-4} and 4×10^{-4}). An Archimedean (2D) spiral is another geometrical innovation of this mathematician. It describes a path of a point moving away from a fixed reference point at a constant speed along a line rotating at a constant angular velocity. This can be represented by $r = a + b\theta$ in polar coordinates (r,θ), in which the a and b constants are real numbers, a turning the spiral and b changing the distance between successive turns. Figure 2.4 illustrates the effect of each constant.

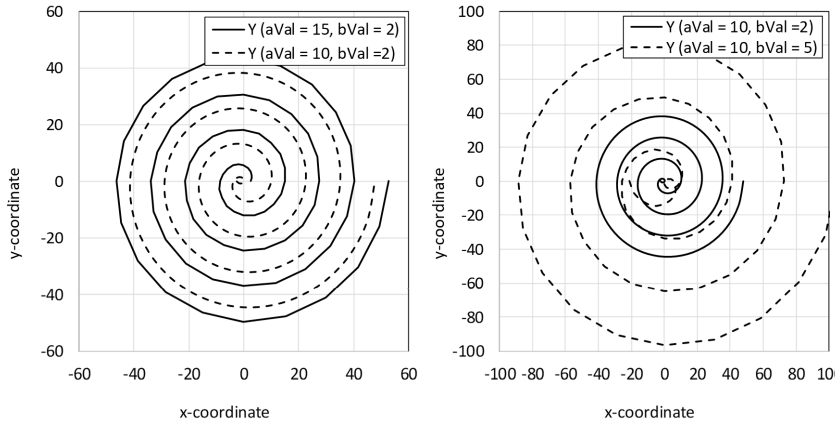


FIGURE 2.4. Archimedean (2D) spiral: (a) Effect of *a* parameter (left), (b) Effect of *b* parameter (right) (drawings created in Microsoft Excel®).

Many centuries later, these methodologies of *Pi* calculation were taken to considerable length. In 1424, Jamshīd al-Kāshī, a Persian astronomer and mathematician, calculated *Pi* with sixteen decimal digits of accuracy (the world record held for 180 years until a twenty-decimal accuracy representation of this number was calculated by Ludolph van Ceulen). In 1596, Ludolph van Ceulen, a German-Dutch mathematician, achieved a thirty-five-decimal approximation, and therefore *Pi* was named after him as the Ludolphian number in Germany until the early twentieth century. In 1630, Christoph Grienberger, an Austrian astronomer after whom the crater Gruemberger on the Moon is named, arrived at a thirty-eight-decimal approximation, the most accurate value known up to this point [35,36,37,38]. In modern times, using progressively more powerful computers, ever greater numbers of digits of *Pi* are being obtained. The current record is over 22.4 trillion digits, achieved by Emma Haruka Iwao in 2019 using the cloud computing service provided by Google, employing only 12 percent CPU utilization, spending two weeks on the computations, and fifteen weeks on the Input/Output (I/O) [39,40].

The Shulba Sutras or Śulbasūtras—the term *śulba* means the *string* or *cord*—are among the Sutra texts generated by Śruti, which contain information on how to use geometry for fire-altar construction, the shapes of the fire-altars having a religious significance. These sutras provide the only source of Indian mathematics information from the Vedic period (500 BC). Diophantine equations, in which only integer values for the unknown variables are acceptable, are discussed in these texts. These works present application examples for the Pythagorean Theorem and triples. They also describe methods of constructing squares and rectangles as well as changing between shapes while

keeping the area constant, which includes going from square to circle and vice versa. The latter task of constructing a square of an area equal to that of a given circle using a finite number of steps and a compass and a straightedge has fascinated ancient geometers for centuries. The impossibility of this was only proven in 1882 and relates to *Pi* being a transcendental (not an irrational) number [41,42].

Euclid of Alexandria, a Greek mathematician who lived in the third century BC, is known as the father of geometry. His work, *The Elements*, consists of thirteen mathematical books and is among the most well-known works in the field of mathematics, serving as the foundation for teaching the subject even up to the early twentieth century. In it he proves from a set of axioms the fundamental postulates of geometry (which later became known as Euclidean geometry, once other types had been discovered in the nineteenth century). Euclidean theories include plane and solid geometries along with incommensurable lines (lines with length ratios being represented by a rational number—the ratio of the two integers). *The Elements* was very important in the development of logic and modern science. Among Euclid's other works are theories related to perspective, spherical geometry, and conic sections—the curves resulting from a plane intersecting a cone, which can be a hyperbola, parabola, or ellipse, with a circle as a special case [43].

In 510 AD, Aryabhata, an Indian mathematician and astronomer, wrote Aryabhatiya—the main surviving mathematical Sanskrit text from the sixth century AD. It presents ways to calculate areas and volumes and also introduces methods to calculate positions of celestial spheres by means of geometric and trigonometric relations. Another sixth-century Indian mathematician and astronomer, Brahmagupta, wrote *Correctly Established Doctrine of Brahma* (*Brāhmasphuṭasiddhānta*), which includes sixty-six examples of Sanskrit verses consisting of two main categories: (1) basic—operations such as cube root, fraction, and proportion; and (2) practical—operations such as series, plane figures, sawing of timber, and piling of grain. In the practical part of his work, his theory on the diagonals of a cyclic quadrilateral—a quadrilateral with vertices all being located on a single circle—is presented. This theory generalized the formula attributed to Hero (or Heron) of Alexandria, a Greek mathematician and engineer from around 60 AD. Hero's formula describes the area of a triangle using its three sides. [44,45].

2.2 MEDIEVAL HISTORY

During the Middle Ages, while the science in Europe lay dormant, mathematics in the Islamic societies flourished in large part thanks to

the work of the Persian (Iranian) mathematicians. Among the most famous of these was Al-Khwarizmi (the ninth century AD). It is from his name that today's word *algorithm* comes. His many great contributions include full development of the decimal system, trigonometry, geometry, and in particular, algebraic geometry—studying zeros of multivariate polynomials. Another ninth-century Persian mathematician and astronomer, Abu-Abdullah Muhammad ibn Īsa Māhānī, is known for explaining the work of Euclid (*The Elements*), Archimedes (*On the Sphere and Cylinder*), and Menelaus (*Sphaerica*), as well as developing his own hypotheses. His most well-known contribution to the field of astronomy is the calculation of azimuths. His astronomical observations led to highly accurate (within half an hour) estimation of the start times of three consecutive lunar eclipses. He also attempted, but did not succeed, to solve Archimedes's problem of dividing a sphere by a plane into two unequal volumes of a given ratio [46,47].

A ninth-century Persian astronomer, geographer, and mathematician, Ahmad Hasib Marwazi, described for the first time the trigonometric ratios now known as sine, cosine, tangent, and cotangent. His *Book of Bodies and Distances* includes geometrical data such as radii, diameters, and circumferences of the Earth, Moon, and Sun; the radius, diameter, and circumferences of the closest and farthest distances to the Moon; diameter and circumference of the orbit of the Sun, and one degree and minute along the orbit of the Sun. Also working in the ninth century in upper Mesopotamia (currently Turkey), Thābit ibn Qurra, a mathematician, physician, astronomer, and translator, developed a formula by which a subset of amicable numbers could be obtained. Amicable numbers are pairs related as follows: a sum of proper divisors of one number is equal to the other number. For example, proper divisors of 6 are 1, 2, and 3. The smallest amicable number pair is (220,284) [48,49,50].

Abū al-Wafā Būzjānī, a Persian mathematician and astronomer in the tenth century AD, made important innovations in spherical trigonometry and established trigonometric formulae such as the sine of the combination of two angles and the law of sines in the present format instead of expressing them as a chord (the modern term is *sine*), as was done earlier by the Greek mathematicians. He also introduced negative numbers for the first time in an Islamic mathematical text. Abū Bakr Muhammad ibn Zakariyyā al-Rāzī, also known as Rhazes, a Persian polymath, scientist, physician, philosopher, and alchemist, known for discovering chemical compounds such as alcohol and sulfuric acid in the late tenth century AD, is the originator of performing experiments in controlled groups leading to a methodical way of trial and experimentation in medicine and other fields of science. This is known today

as the design of experiments, a methodology that is the foundation of many scientific experiments from medical to engineering fields [51,52].

Hasan Ibn al-Haytham, a Persian polymath, mathematician, astronomer, physicist, theologist, and physician, is known as the father of optics, famous for his work *Book of Optics*, written in the period from 1011 to 1021. He is the first scientist who explained that vision happens as the result of light bouncing back after hitting objects and then being directed to the human's eyes, with the vision happening in the brain and not in the eyes. He is also the one who presented the scientific method of proving a hypothesis (experimentation) using repeatable and reliable procedures (design of experiments). Based on mathematical evidence (modeling), the three elements of his established scientific method were employed five centuries later by the Renaissance scientists. Muhammad al-Jayyānī, an Andalusian (present-day Spain) scientist, mathematician, Islamic scholar, and judge, wrote significant interpretations for Euclid's *Elements* as well as the first known hypothesis on spherical trigonometry in the eleventh century AD [53,54].

Omar Khayyám, a Persian mainly known for his poetry, was also an accomplished mathematician and astronomer who wrote several mathematical works. These include *A Commentary on the Difficulties Concerning the Postulates of Euclid's Elements* presented in 1077, *On the Division of a Quadrant of a Circle* some time before 1079, and *On Proofs for Problems Concerning Algebra* in 1079. Khayyam's work in analytic geometry is considered as the ancestor of Descartes's later achievements. In his work *On the Division of a Quadrant of a Circle*, he investigated whether it is possible to divide a circular quadrant into two parts such that the line segments that are projected from the dividing point (on the circumference) to the perpendicular diameters of the circle can create a specific ratio for the segment areas. The by-products of this work were curves containing cubic and quadratic terms. He is the first mathematician who developed a general theory of cubic equations and geometrically solved them for cases of positive roots. His work categorized cubic equations into ones that can be solved using a compass and straightedge, by means of conic sections, and ones that include the inverse of unknown variables [55].

Archimedes's problem of dividing a sphere into two unequal parts of a given volume ratio was finally solved by Abū Ja'far al-Khāzin, a Persian astronomer and mathematician in the twelfth century AD. Al-Khāzin's work *Tables of the Disks of the Astrolabe*, which was widely referenced, presents designs for a number of astronomical instruments, including an astrolabe with cover plates with tables along with commentary on their usage. Al-Khāzin is also famous for his interpretation of *Almagest*—famous mathematical and astronomical treatise related to the apparent motions of the stars and planetary trajectories, written by Claudius

Ptolemy, a Greco-Roman mathematician, astronomer, and geographer (the second century AD). Al-Khāzin offers nineteen interpretations for the statements presented in Ptolemy's work. He also proposes a solar model different from that of Ptolemy [56,57,58].

Muhammad ibn Muhammad ibn al-Hasan al-Tūsī, a thirteenth-century AD Persian polymath, scientist, architect, philosopher, physician, and theologian, wrote his famous work on trigonometry as an endeavor independent from astronomy. His work *Treatise on the Quadrilateral* provided a thorough explanation for spherical trigonometry, presenting the relationships between trigonometric functions of the sides and angles for spherical polygons. Al-Tūsī developed the science of trigonometry as an independent entity separate from astronomy; trigonometry was considered inseparable from astronomy until then. Al-Tūsī was the first mathematician who studied six independent examples for right triangles in the field of spherical trigonometry. In his famous work *Sector Figure*, he presents a proof for the law of sines for plane and spherical triangles. He is the discoverer and prover of the law of tangents for spherical triangles. His trigonometry work used some of the knowledge presented by Menelaus of Alexandria, a Greek mathematician and astronomer in 100 CE—who wrote his book *Sphaerica* on spherical trigonometry—and Būzjānī and Al-Jayyani. The early hyperbolic geometry developments were based on the theorems of al-Haytham, Khayyam, and al-Tūsī on quadrilaterals (polygons with four vertices and edges) and the Saccheri quadrilateral (equilaterals with two equal sides perpendicular to their base). These works, along with their alternate hypotheses such as Playfair's axiom (replacing the fifth Euclid hypothesis), led to revolutionary non-Euclidean geometries of the modern age [59,60].

2.3 MODERN HISTORY

Among the works of modern non-Euclidean mathematicians are the ones belonging to Vitello (or Witello), a Polish mathematician and scientist, known for his work *Perspectiva* (from the thirteenth century AD), based on al-Manzir's earlier (the eleventh century) work on optics [61]. Geometry gradually strengthened by the seventeenth century when analytical concepts were added to the field, even though the fundamental definitions remained the same throughout. The resulting two important developments in the field of geometry are analytic geometry, which is the creation of geometry with coordinates and equations presented by two famous mathematicians René Descartes and Pierre de Fermat, and systematic geometry, which is the study of projective geometry

presented by Girard Desargues. This introduction was the precursor to the development of modern calculus, quantitative physics, and projective geometry, which is a geometry dealing with how points are related to each other without consideration of the actual distances from a reference point.

René Descartes, a French mathematician, philosopher, and scientist, is famous for introducing the Cartesian Coordinates in 1637. The well-known expression, "I think, therefore I am (je pense, donc je suis)," also known as *cogito*, is attributed to this philosopher. This is the testament of knowledge when in radical doubt based on the foundation that one's own existence proves the reality of the mind. His theory is therefore a logical axiom. He is also called the father of analytic geometry, which is the connection between algebra and geometry, and it is the foundation for infinitesimal calculus, presented in 1636 by Pierre de Fermat, a French mathematician, lawyer, and politician. Infinitesimal calculus takes its name from working with infinitesimally small numbers (e.g., greater than zero but smaller than fractions such as 1 and 1/3) with properties that are similar to continuous (real) numbers. Fermat is also the innovator of adequality, a method used to calculate the maximum and minimum of functions, area, center of mass, and tangents to curves, to name a few [62,63].

Gersonides, a French philosopher, mathematician, astronomer, and physician, is known for his work *Maaseh Hoshev* written in 1321 that focuses on arithmetical operations such as root and cubic root extractions, the sum of consecutive squares, and binomial equations, as well as geometrical models for the motion of the Moon using a camera obscura—also known as a pinhole image. The latter is a reverse and inverted image projected through a small hole on a screen. John Wallis, a British clergyman and mathematician in the seventeenth century, is known for partial development of infinitesimal calculus. Although he had also proven the Pythagorean theorem by means of similar triangles, ibn Qurra, his predecessor by six centuries, is the original prover of this hypothesis, expanding his proofs to all types of triangles. Giovanni Girolamo Saccheri was an Italian mathematician, philosopher, and priest, known for his 1733 treatise on non-Euclidean geometry, as the second most important work in this field, *Euclid Freed of Every Flaw*. This work effectively built upon Khayyám's eleventh-century treatise on the same subject, though it is not known whether Saccheri was aware of its existence [64,65,66,67,68,69,70].

Two developments in geometry in the nineteenth century changed the status quo. It started with the discovery in the 1820s of non-Euclidean geometries by Nikolai Ivanovich Lobachevsky, a Russian mathematician and geometer, after whom a century and a half later

a lunar crater would be named. He is known for his work on hyperbolic geometry and Dirichlet integral principles, also known as the Lobachevsky integral formula. Carl Friedrich Gauss, a nineteenth-century German mathematician and physicist, is well-known for his contributions to mathematics, physics, and number theory as well as his most famous publication, which describes modular arithmetic (*Arithmetical Investigations*). Farkas Wolfgang Bolyai, a German mathematician in the nineteenth century, Gauss's friend, attempted to prove the parallel axiom (Figure 2.5). This postulate states that if a line crosses two straight lines and two interior angles ($\angle A$ and $\angle B$ in the figure) that sum to less than two right angles are created as the result (the same side), then the two lines, if extended, meet at a point (on the side the summations were made). He is famous for his work, the *Tentamen*, which investigated geometry, arithmetic, algebra, and analysis in a systematic fashion. He used iterative procedures to solve converging solutions to problems. His method was able to predict if an infinite series is converging or diverging. This theory is independent of but equivalent to Raabe's test, which performs the same function [71,72,73,74].

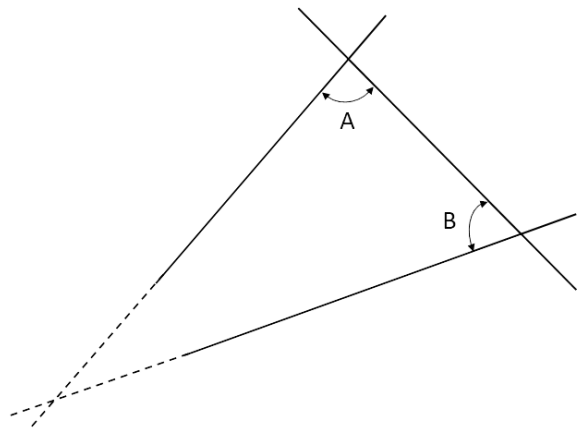


FIGURE 2.5. Parallel postulate (axiom) (drawings created in Solid Edge).

Farkas Wolfgang Bolyai's son, János Bolyai, became an accomplished mathematician, linguist, and musician and distinguished himself with the discovery of non-Euclidean geometry that represents the universe more accurately. Reportedly, after publishing his work on non-Euclidean geometry, Gauss wrote to Farkas Bolyai that praising Bolyai's work equaled praising himself, since it was aligned with the thoughts occupying his mind for the three decades prior to the time of its publication. It is argued that Gauss was fearful of the controversies surrounding publishing his non-Euclidean theory even though

he had developed the idea of parallel lines, which was different from non-Euclidean geometry by Bolyai. János Bolyai is known as one of the founders of non-Euclidean geometry [75].

Saccheri's work was rediscovered by Eugenio Beltrami, an Italian mathematician known for his work in differential geometry and mathematical physics, in the nineteenth century. Beltrami used constant curvature, a pseudosphere, and the interior of an n-dimensional unit sphere to model non-Euclidean geometry, proving it was consistent. This is also known as the Beltrami–Klein model. For example, the Klein-Beltrami model of hyperbolic geometry is an open disk in a Euclidean plane with open chords connecting with hyperbolic lines. Beltrami was a contributing scholar in developing tensor calculus by Gregorio Ricci-Curbastro, an Italian mathematician in the twentieth century, and Tullio Levi-Civita, an Italian mathematician in the twentieth century, known for his work on absolute differential calculus and its application to the theory of relativity [76,77,78,79].

The other non-Euclidian concept developed was as part of the Erlangen program, a geometry characterization method, considering group theory (think of a Rubik's cube) and projective geometry. In this method, lines and vertices are projected into the space of a given perspective. In this method, the lines are mapped (projected) into other lines so they are collineations or mathematical transformations in colinear features such as vertices and lines that are being transferred to other planes or spaces. This method of categorization was presented by Christian Felix Klein, a German mathematician and educator in 1872. Georg Friedrich Bernhard Riemann, a German mathematician in 1854, is well-known for his work on the Fourier series and development of the Riemann integral, as well as Riemann surfaces focusing on complex analysis. He is also famous for developing the prime function method in which the number of prime numbers before a certain real number are counted, which is an important contribution to analytical number theory, which specializes in solving mathematical problems about integers. The theory of higher dimensions was developed by Riemann as a student under Gauss in 1853; this theory was eventually presented in 1868 under the title *On the Hypotheses which Underlie Geometry* and is considered one of the most important theories in geometry. In this work, Riemann expanded the differential geometry of surfaces into $n°$ dimensions. Adolf Hurwitz, a German mathematician known for his works on algebra, analysis, geometry, and number theory, expanded in 1898 on Riemann surfaces as one of the important bases for topology along with Klein [80,81,82,83,84].

Jules Henri Poincaré, a French mathematician, theoretical physicist, philosopher, and engineer, was also known as the last universalist,

as he excelled in many related fields in his lifetime. He made important contributions to math-related sciences such as physics and mechanics, both in the pure and applied sense. Poincaré also postulated the Poincaré conjecture, which was only solved in 2003 by Russian mathematician Grigori Perelman. Conjecture theory states that, if simply connected, a closed three-dimensional 3D manifold forms a sphere. In his research on the three-body problem, Poincaré became the first person to discover a chaotic deterministic system, which laid the foundations of modern chaos theory. He is also considered to be one of the founders of the field of topology. As a consequence of these major changes in the conception of geometry, the concept of *space* became something rich and varied, as well as the natural background for theories as different as complex analysis and classical mechanics [85,86].

A hypersphere is a set of points that are located at a constant distance from a point that is fixed over time and space, also known as its center. It may be extended to a 2D plane; for example, a circle in this scenario is a hypersphere in a 2D space, but it is not a hypersphere in a 3D space; a 3D sphere is considered a hypersphere in a 3D space, but it is not in a four-dimensional (4D) space. By increasing the distance from the center of the hypersphere, the represented curvature decreases. Duncan Sommerville, a Scottish mathematician and astronomer who originally introduced the hypersphere idea, wrote a review on non-Euclidian geometry in 1900. His book on *An Introduction to Geometry of N Dimensions* discussed polytopes, which are geometries with flat sides. They also form parts of 3D objects [87,88].

2.4 LAND SURVEY

Land survey—an ancient practice dating back to 2000 BC, or possibly even prior to that—has been both the driving force for geometrical knowledge development and naturally the beneficiary of new geometrical concepts. The purpose of surveying is to locate points on the Earth's surface horizontally and vertically by measurement of distances and angles, though, of course, the same techniques can be applied to the surface measurements of any other heavenly body.

Among the earliest surviving examples of surveying work is the famous Stonehenge site, located in Wiltshire, the United Kingdom, which was built around 2500 BC. It is believed that pegs and ropes were used to locate the stones. Researchers have examined the accuracy of using this method, particularly for constructing the Neolithic 56 Aubrey Hole circuit at Stonehenge. Studies carried out with CAD tools found that this technique resulted in a regular fifty-six-sided polygon, which

can accurately approximate a regular heptagon. The comparison of this method with earlier methods suggests that it is the simplest yet most accurate method of its type. It is speculated that the Great Pyramid of Giza in Egypt, built about 2700 BC, also used this method [89,90,91].

One of the earliest known surveying instruments is the Groma. Likely invented in Mesopotamia in the fourth century BC, it was later adopted widely by the Romans. It consisted of two horizontal sticks connected perpendicularly with plumb lines hanging from their tips. These were placed on top of a pole that could then be planted into the ground. This tool was used to demarcate square pieces of land.

Abel Foullon, a sixteenth-century French engineer, presented what is believed to be the earliest description of a plane table—a surveying device used to make a level and solid surface to draw field variations or maps. Leonard Digges, a sixteenth-century British mathematician and surveyor, was first to describe a theodolite—an instrument to measure angles in horizonal and vertical planes, still used for land surveying, meteorology, and even rocket launches. A compass was first incorporated into a theodolite in 1576 by Joshua Habermel, a Czech watchmaker and instrument maker. In 1615, Willebrord Snellius, a Dutch astronomer and mathematician, introduced the modern systematic use of triangulation. This is the method of using multiple chains of triangles in order to find the distance and position of the points within a region. Specifically, one side's data is used in combination with the baseline to obtain the lengths of the other two sides and the opposing angles.

Edmund Gunter, a British mathematician and astronomer, introduced in 1620 a Gunter's chain—a distance-measuring device used for land survey for legal and commercial purposes. The first telescope was incorporated into a theodolite in 1737 by Johnathon Sission, a British instrument maker. This was the commencement of modern instrumentation in this category. In 1785, Jesse Ramsden, a British mathematician and instrument maker, introduced the first precision theodolite used for measuring the latitude and longitude of London and Paris as well as for the Principal Triangulation of Great Britain [92,93,94,95,96,97,98].

The method of triangulation was adopted by Jacques Cassini, a French astronomer, and his son César-François Cassini de Thury, also a French astronomer and cartographer, in the first triangulation of France in the eighteenth century, resulting in the resurveying of the meridian arc and the first map of France in 1745. Triangulation spread to the rest of the Europe by 1784. Major-General William Roy, a Scottish engineer and surveyor, started in 1784 the first high-precision trigonometric survey of the whole of Great Britain, also known as the Principal Triangulation of Great Britain, which was completed in 1853. The Great Trigonometric Survey of India,

surveying the entire Indian subcontinent with scientific precision, began in 1802 under the supervision of General William Lambton, a British surveyor and geographer, and would take nearly seventy years to complete. By 1830, his successor in this monumental endeavor was George Everest, a British surveyor and geographer for whom Mount Everest is named. The survey continued in 1843 under Major-General Andre Scott Waugh and was completed under General James Walker in 1871. Mount Everest and the other Himalayan peaks were mapped during this grand survey. During the nineteenth century, surveying became more accessible and modernized, and detailed geographical features, such as city and road maps, were further developed [99, 100,101,102,103,104].

In 1957, Trevor Lloyd Wadley, a South African engineer known for the development of the Wadley loop circuit, invented the tellurometer to measure long distances using microwave transmitters and receivers. Electronic Distance Measurement (EDM) equipment, or the range finder, was introduced in the 1950s and has been used in virtual reality applications since the 1990s. It employs a multi-frequency phase shift of light waves and saves considerable time of chain measurement requiring measuring between points kilometers apart. EDM equipment decreased in size with further development. In the 1970s, equipment was developed, integrating the electronic theodolite with EDM in order to read the slope between the instrument and a reference point; the unit was equipped with a DAQ system connected to a computer and a program that could make advanced coordinate-based calculations [105,106,107].

The Navy Navigation Satellite System (NNSS), also known as the Transit system, was the first satellite navigation system, and it was launched in 1960 for the purpose of providing coordinate positions to Polaris missile submarines. It was later adopted by surveyors to find the location of benchmark points at remote locations. Global Positioning System (GPS) was launched by the Unites States Air Force in 1978. This improved the accuracy of the Navy's system by using a larger constellation of satellites and better signal transmission. With the improvement of the GPS systems, the original accuracy that was achieved only by recording and averaging data by a static source over hours was improved to become the efficient and effective method of Real Time Kinematic (RTK) surveying. RTK made the collection of accurate data possible by means of a fixed base station and a second roving antenna whose position may be tracked [108]. Nowadays, theodolite, total station, and RTK GPS survey methods are the main surveying techniques. A total station is a device that combines the functionality of a theodolite of measuring angles with the determination of its location. Remote

distances are being sensed and imaged topographically by satellites by means of 3D scanning and lidar—a distance-measuring method illuminating the target with pulsed lasers and measuring the reflected pulses with a sensor. UAV technology, or Remote Pilot Vehicles (RPV)—also known as drones, aircraft without a human pilot aboard—along with photogrammetric image processing, the science of using photographs to make measurements by recovering the exact locations of the surface points, are employed as well [109,110,111,112,113].

2.5 UNITS OF MEASUREMENT

Records of the earliest measurements made for agriculture, construction, and commerce date back to the third and fourth millennia BC. Smaller communities had their own standard units for lengths, areas, volumes, and masses. Volumes, for example, were used to measure dry grains. As communities became more interconnected through trade, standardization of measurement techniques and units increased over time. By the eighteenth century, the science of metrology (measurements) was developed, around the time of French revolution [114,115,116].

The earliest unit of length measurement, originating in the third millennium BC, is the Mesopotamian cubit. Later came the Indus Valley units of length, also referred to as the Egyptian cubit. A cubit was the length of the forearm—the elbow to the tip of the middle finger. Its subdivisions, based on the distances derived from the human hand, were: (1) the length between the tip of the little finger to the tip of the thumb, which was referred to as one-half cubit; (2) the palm or width of the hand was known as one-sixth cubit (Roman palm); and (3) the width (digit) of the middle finger was referred to as one-twenty-fourth cubit. The royal cubit (or Egyptian cubit) was introduced later on as the standard cubit and was equal to seven palms. Units of measurement which came later (e.g., foot, inch, and yard) were obtained from volume (cubic) measures or possibly proportions of the cubit.

The foot as a unit of measurement originated in Egypt and found its way to Greece and Italy; therefore, the Roman foot (0.296 m) was introduced. Five Roman feet (*pedes*) equaled 1.48 m (4 ft 10 in)—double steps, and one Roman mile was known as the mille passus (1,000 paces). England adopted the Roman mile of 5,000 feet during the Roman occupation, which was later changed by Queen Elizabeth I to 5,280 feet (1 mile) or 8 furlongs—one furlong is 40 rods (= 202 m = 220 yards = 660 ft = 10 chains). The commonly known unit of yards (0.9144 m) arrived later. Some believe it was a double cubit, dividing

it into multipliers of 2 (i.e., 2, 4, 8, and 16), and half-yard, span, finger, and nail were defined. Body measurements (grid) such as waistline or the distance between the tip of the nose to the end of the thumb were introduced later on. These measurements were employed in Great Britain for many years as part of dressmaking practice. Rods, poles, and perches were also employed as units of length measurement [117].

Decimal numbers were later on added to integers. Simon Stevin, a Flemish mathematician, physicist, and engineer, introduced the use of decimal numbers for everyday purposes in his work *The Arts of Tenths (De Thiende)* in 1585. Bartholomaeus Pitiscus, a German trigonometrist, astronomer, and theologian, used this notation in his trigonometrical tables in 1595. John Wilkins, a British philosopher and author, suggested a system of measurement similar to what is currently known as the metric system in 1668. He proposed to keep the second as the basic unit of time (commonly used at the time) and suggested length of a pendulum crossing the equilibrium position once every second (a two-second period) be employed as the standard unit of length (994 mm). Gabriel Mouton, a French scientist, published a similar proposal to that of Wilkins's in 1670, except his base unit of length was 1/1,000 of a minute of an arc (about 1.852 m) of geographical latitude, calling it *virga*. He also suggested using prefixes for units of length [118,119,120,121].

Thomas Jefferson, an American politician, lawyer, and architect, in his capacity as the Secretary of State, submitted a report in 1790 to the United States Congress that proposed use of a decimal system of coinage and of weights and measures, using a foot as a unit of length (he proposed the units of mass and volume as ounce and bushel). He employed a rod versus a traditional pendulum, which was oscillating at a 45-degree latitude; the foot was adopted as a fifth of the rod length. He also proposed the use of a decimal system similar to the French; however, he was not keen on employing prefixes. Having said that, he used the old names for the new decimal associates. His suggested foot length was a quarter inch shorter than the foot used at the time (the old foot was equivalent to 0.2985 m). Jefferson intended for the new measurement system to substitute for the Gunter chain and the traditional acre, but it was not fully realized. The metric system was presented in 1668 and only accepted in 1799—some countries, such as the United States and China, continued using their own systems of measurement. Some nonmetric units such as the Scandinavian mile (10 km) are among the ones that adapted themselves to equal multiples of the metric system, except for the American system, whose units have not been adapted as such [122,123].

CHAPTER 3

GEOMETRY, CREATIVE ART, AND PROPORTIONALITY

Geometrical concepts have been employed by many creative artists and scholars in their work. One of the most famous of them was Leonardo da Vinci (1452–1519), an Italian polymath. During his lifetime, he made significant contributions to engineering by developing conceptual designs centuries ahead of their time, as well as to the science of geometry by developing a rigorous system of proportions for his geometrical designs. A universally known example where he carefully applies proportions in a sketch is the body anatomy study known as the *Vitruvian Man*. Da Vinci created this drawing based on the writings of Vitruvius, the Roman architect and engineer in the first century BC. In his work *De architectura (Book III)*, Vitruvius described what constitutes a perfect human anatomy and how its proportions can be connected with geometrical shapes.

Da Vinci's paintings reflected his intensive studies of the human body's anatomy and its proportions, the formation of its elements (e.g., muscles) when exposed to the light, varying as its direction changes, the time of the day, the expression on the face, the curves and straight lines and the 3D representation of the figures that otherwise look flat, using shading and the concept of perspective. His biological sketches of the human skull are perhaps among the most accurate representation of this organ. For example, before his work, the shapes of the frontal sinuses were unknown. His drawings of the human skull precisely depicted the sinus cavity shapes and their exact locations over the eyebrows.

3.1 GEOMETRY COMPONENTS

Similar to how matter is built up from elementary particles, geometrical entities can also be thought of as being built up from fundamental elements. The most fundamental geometric entity is a point or a vertex. Then, vertices are connected to create edges. Edges are linked at vertices to make planes; planes are joined at edges to form 3D features; sub-volumes (sub-domains) are combined at vertices (or edges or surfaces) to form domains. These linkages continue until the most complex structures are created either by nature or by humans. One must note though that organic shapes often only have surfaces and no distinct edges or vertices.

Let us examine the geometrical structure of two very common substances: hydrogen and oxygen. In both cases, their molecules are made up of two atoms (one can think of them as vertices); thus, such molecules are known as diatomic elements. Each forms a linear molecule, since they have symmetrical polarities—meaning that the positivity and negativity of the charges are evenly distributed. When these two elements come together and form a water (H_2O) molecule, the polarity changes, as the hydrogen is less negatively charged than the oxygen; thus, they create dipoles, with the negative pole located at the oxygen atom and the positive pole somewhere in between the two hydrogen atoms, forming a nonlinear, symmetrical (about one plane) linkage. However, not all molecules made up of two different elements arrange themselves in this angular manner—for example, in the case of carbon dioxide, all three atoms arrange themselves along a line, since a carbon dioxide (CO_2) molecule consists of two carbon-oxygen (C=O) polar bonds with the dipole moments canceling.

There are three components when creating a domain (geometry): (1) size—relative dimension of the object with respect to a reference point; (2) proportionality—size ratios; and (3) location—relative positions of the subcomponents with respect to each other. Let us provide an example to clarify this further. Figure 3.1 shows the linear carbon dioxide (CO_2) and nonlinear water (H_2O) molecules. Each atom has its own size, and distances between the centers are shown in this diagram. In this scenario, to obtain the diameter of the atom, the elemental volume is first calculated (i.e., the molar mass divided by the density—this is to be divided by Avogadro's number ($N = 6.022\times10^{23}$) to result in the volume of the individual atoms forming the molecule) [124]. Depending on the assumed shape of the atom—cube or sphere—the characteristic length of the atom may be calculated. For a cube shape assumption, the side length is the cube root of the volume ($D = \sqrt[3]{V}$)—Figure 3.2.

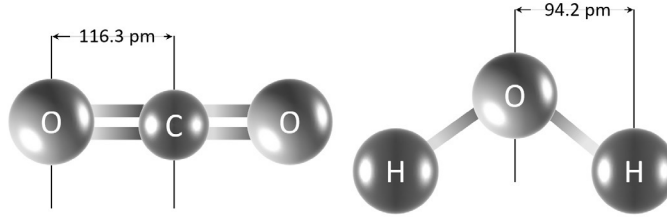


FIGURE 3.1. Linear and nonlinear molecules—dimensions are to scale: (a) Linear CO_2 (left), (b) Nonlinear H_2O (right) (drawings created in Microsoft PowerPoint®).

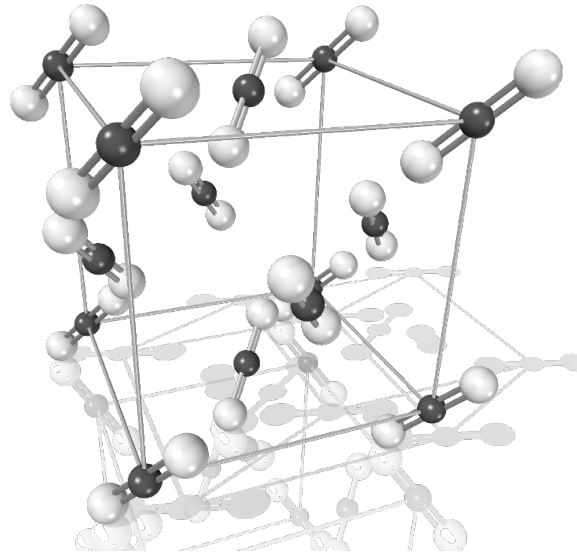


FIGURE 3.2. Linear CO_2 molecules in a crystal form (dimensions not to scale) (drawings created in Solid Edge).

For a spherical shape assumption, the diameter is given by $D = \sqrt[3]{6V/\pi}$. To calculate the proportions, the size of an individual atom is considered as the reference size (e.g., oxygen in this example). The size of other elements is compared with respect to this reference element. For instance, if the size of the oxygen atom in Figure 3.1 is assumed to be the unit of measurement (i.e., cm or in), the ratio identifies the size of the carbon and hydrogen atoms with respect to the oxygen atom (0.82 and 1.00, respectively). This means the representation is to *scale*—meaning that the diameter of the reference atom (oxygen) may be represented with whatever size desired to the *scale* intended. The next geometry specification is the distance of the subcomponents (atom bond lengths in this example). Table 3.1 provides data used to obtain the sizes and proportions listed in Table 3.2 and depicted in Figure 3.1 [125,126].

TABLE 3.1. Molar density and radius for elements presented in Figure 3.1 (raw data from [125] and [126]).

Element	Molar Weight (g)	Density (g/cm^3)	Volume (cm^3)	Cubed Particle		Spherical Particle	
				Diameter (pm)	Radius (pm)	Diameter (pm)	Radius (pm)
Oxygen	16.00	1.429	1.86E-23	264.91	132.46	328.68	164.34
Carbon	12.01	1.95	1.02E-23	217.07	108.53	269.32	134.66
Hydrogen	1.01	0.08988	1.86E-23	265.06	132.53	328.86	164.43

TABLE 3.2. Size ratio for the elements presented in Figure 3.1 and Table 3.1 (raw data from [125] and [126]).

Ratio		1/Ratio		Size		
C/O	0.82	O/C	1.22	Size (O)	Size (C)	Size (H)
H/O	1.00	O/H	1.00	1.00	0.82	1.00

3.2 FORM, FIT, AND FUNCTION

The concepts of size, proportion, and relative location have other applications such as beauty. For example, a proportionate face, where the eyebrows, eyes, lips, and cheeks are situated at certain locations, is perceived as pleasant to the eyes of viewers and is associated with a beautiful face. It is true that beauty is in the eye of the beholder. A playful example is a mother cockroach's perspective of the beauty of her children, their clear fair complexion on their femur, tibia, tarsus, and tibial spurs beautifully decorating their tibia. If you wonder where those parts are, think of a leg, divide it into three segments, starting from the top section (femur) to the bottom part (tarsus) with an addendum (spurs) in the middle section (tibia). While most people are not their fans, cockroaches may be among the most interesting creatures on the Earth. They can remain active for a month without food, without air for an hour, under water for half an hour, and can be exposed to nuclear radiation for up to fifteen times longer than humans. Now that you know this, you may think twice if you have a choice to resurrect as a unicorn or a cockroach—cockroaches may inherit the Earth [127,128].

The concepts of size, proportion, and relative location have other more practical applications where form, fit, and function play vital roles. Imagine a buttonhole, for example. The main application of the hole is to secure your pants or shirt on you by using a button. The buttonhole should be situated so that the button can easily be reached, large enough so that the button can pass through it, and of a size in correct proportion to that of the button so that the button can be easily but safely secured. These three requirements are so important that if

they are not satisfied and this critical system fails to fulfill its function, it may lead to historical events like those unfortunate wardrobe malfunctions. Some garments actually resort to a fastener redundancy approach similar to critical systems in an aircraft.

Another good example is a shiny pair of yellow designer shoes that have caught your eye. You intend to wear them to your book party with a matching blue satin outfit. The yellow patent leather shoes, with the regal status they represent, serve the function of being worn to a formal occasion. They also have the correct form, beautifying your ensemble; however, if they are not comfortable on your feet—imagine wearing them while dancing through the day—they will not satisfy the fit criteria. You realize that it would have been better if you had chosen another more comfortable pair in order to satisfy all three requirements of form, fit, and function.

As you see, the concepts of form, fit, and function may be applied to a variety of applications, from the simplest to the most complex, such as the creation of a spaceship. The location of the vertices, the main points of connection where the geometry is under the most stress (e.g., the joints on the insect legs and corners of the buttonhole) are to be designed so that the load bearer (i.e., the insect and human) can tolerate the exposed loads when the product is used (i.e., creeping and dancing). The connecting links between the vertices should form correct bonds so that the finished product as a whole can function properly. The finished product you see, from the reference point, along the three coordinates, in straight or curved fashion, with the links that may be in straight or curved fashion (concave—curving inward, or convex—curving outward), to the infinite number of points, all the connecting dots that make the trip possible, the "experience" along the way, the points of being but also of doing, satisfy the form requirement.

However, let us expand on the concept by presenting functionality in simple to complex systems (structures). The term *simple* (versus *complex*) is a relative concept in our familiar world of vertices located at different orientations and therefore forming simple or complex 2D or 3D geometries. The geometries may be as complex as Gaudi's Sagrada Familia or as simple as a pinpoint cavity inside the same structure [129]. The functionality of the former mainly lies in its use as a location for congregating while looking with awe at this massive structure. It is like Michelangelo's Sistine Chapel ceiling, where the hands of the engineer and invention touch for the very first time. The pinpoint cavity functions as a light trap, as the light passes through its opening, hits the opposite surface, then is reflected from that surface, hitting the surface it is directed to; this continues until its energy is fully absorbed by this small light trap.

The concept of functionality also may be described as meeting performance expectations efficiently with the intent of meeting the requirements. Utility relates to mechanics of the structure, in terms of carrying loads. Imagine doing yoga poses. When performing planks, the body is like a beam bridge, and as it moves to a downward facing dog, the bridge pose changes to an arch bridge. The hands are firmly planted on the ground, receiving energy from it and giving it back to the Earth by the toes and, for the more seasoned yogi, by the feet as points of connection. When moving through vinyasa to warrior pose, the body alignments change until the final pose is achieved. The biomechanics of the pose—if performed accurately (for example, the front knee staying at a 90-degree angle and not going beyond the ankle)—serve the intended function. It is all about load distribution, the forces that are applied to the joints and muscles. Imagine a truss bridge you learned about when studying the strength of materials.

3.3 DESIGN OPTIMIZATION

Design optimization refers to varying the part's structure to fulfill certain requirements. Optimization is carried out subject to specific loading and constraint conditions for a structural component, for example, such as where it is attached to a wall, where the loads on it are, and the load size. The optimization goal may be to minimize compliance while using mass equal to a certain fraction of the initial design domain mass. There are three types of optimizations: (1) topology optimization, where material can be added or removed anywhere within the design domain; (2) size optimization, where the dimensions of existing features within the geometry are varied—for example, if there are holes inside a domain, the hole dimensions can vary but their number or circularity does not change; (3) shape optimization, in which the holes can be stretched into any elliptical shape but holes cannot be added or removed.

With the development of technology and changing design requirements, the need for introducing lighter and stronger structures for performance enhancement has increased. Social responsibility and environmental considerations have also increased the need for new materials and techniques to be developed to make geometries and designs that are both aesthetically pleasing and also serve the function with the highest efficiency and effectiveness possible. For example, in 1941, at the start of the World War II, the thirty-cylinder Chrysler A57 multibank gasoline engine was originally employed to power the M4A4 medium tanks; later it was replaced with the eight-cylinder Ford GAA

gasoline engine, which was able to deliver comparable power. This new engine with enhanced simplicity reduced the use of valuable resources such as fuel, mechanical parts, and manufacturing person-hours. Just like in 1941, nowadays, the engineers continue to work to make the parts lighter [130,131].

For example, a lighter aircraft equals less fuel consumption, more payload, and often better serviceability, which is even more important both for shorter and longer hauls. The fleet that operates on shorter hauls is exposed to more cyclical structural fatigue due to the greater number of times that the engines are turned on and off, due to the fuselage experiencing pressurization cycles as well as increased stresses during takeoffs and landings. Therefore, a stronger structure, either reinforced by the material or design, seems only natural. In long-haul flights, on the other hand, fuel consumption plays a paramount role, and for this reason fuel storage is to be kept to a minimum to be able to carry more payload. Therefore, the need increases for reducing fuel consumption by introducing lighter designs. Note though that *lighter* or, to put it more accurately, less dense material, does not necessarily make a lighter structure. More material may need to be used so that it can carry the same load and the total weight may be the same or even more. Thus, engineers talk about the strength-to-weight ratio for the material as the key parameter.

Another method would be introducing fuel that burns more efficiently. For a piston-engine aircraft, think about the use of green 100/130 aviation fuel with a 100-octane rating for a lean setting (smaller fuel-to-air ratio) during less demanding operations such as cruise, compared to a 130-octane rating for a rich setting (higher fuel-to-air ratio) during more challenging operations such as taking off. Replace it with a blue 100LL (Low Lead) version that has half of the Tetra-Ethyl-Lead (TEL), a toxic ingredient used to reduce engine knocking (detonation) in the former version. A more environmentally friendly option is 100LL, for it introduces less toxin into the atmosphere. The unleaded version of 100LL, G100UL, has been introduced with energy efficiency that is said to be better by 3.5 percent than its predecessor. A turbofan-powered aircraft, such as a Boeing 777-E300, burns about 7.5 tons of fuel per hour, cruising at 892 km/hr (554 mph) at 30,000 ft. The upgrade package presented in 2016 increased the fuel efficiency by 2 percent, resulting in fuel usage of 7.35 tons per hour. Further increase of efficiency may improve the fuel consumption to 7.1 tons per hour. Assuming that the operating cost of the Boeing 777-300ER in 2018 was about $10,250 per hour, from which 60 percent is the cost of fuel, the said efficiencies result in fuel cost of $6,027 and $5,816 per hour compared to the original cost of $6,150 per hour. That is equivalent to

a saving of about $1,230 and $3,340, respectively, for a ten-hour flight. Assuming the aircraft is operating in a two-class configuration, the cost per seat is about $27.12 for the Boeing 777-300ER. The said savings are about $4 and $9 per seat for the same flight length—the original total fuel cost per seat was about $163. With taxes and other surcharges, the savings per seat increase even more [132,133].

Optimizing part creation methods and geometry design are among other techniques used to improve structure efficiency. This can be done by introducing better-performing materials such as plastics and composites that have higher strength-to-weight ratios and by distributing these materials within the designed structure in a way that makes the most efficient use of their mechanical strength (topology optimization). An example is the galley of the new Airbus A300. This part separates the cabin from the service area, includes an opening where a portable patient bed may pass through, and accommodates a double-seated murphy platform that is attached by hinges to the galley panel. The seating area is as large as the previous version, and so is the panel design; however, the weight is reduced by 45 percent. The new design is made of special plastic with a strength that surpasses its predecessor and has only four connecting points to remain in its place. The inspiration for this one-inch thick panel capable of tolerating up to *16G* acceleration loads comes from the human bone. The parts are created individually by 3D printers and then joined together. Although it is possible to print small-size components in a single piece, current 3D printers have limitations in creating larger components. Space panels on overhead storage components are among the small printed pieces that are certified and used in the Airbus Finnair.

These designs are able to use high strength-to-weight ratio materials that are deployed strategically such that they can provide the greatest benefit, often employing complex lattice structures that can only be manufactured with Additive Manufacturing (AM) 3D printing technology. In AM, the 3D computer models are sliced into layers and parts are manufactured by incrementally adding material at each layer. This manufacturing technique frees the designer from shape complexity constraints imposed by more traditional subtractive (machining), molding, or casting methods.

The most commonly widely available AM technique is based on the Material Extrusion (ME) principle, where a polymer filament is pulled through the extruder head, is molten, and is deposited in lines on each layer. The 3D printers of this type are available priced as low as a few hundred dollars. Another technique is generically referred to as Vat Photopolymerization (VP) but is better known by its proprietary name of Stereolithography. In it, parts are produced by selectively curing a

thin layer of photosensitive liquid resin with an Ultraviolet (UV) light. Finely detailed parts with accurate dimensions can be produced by this method, though the material is a specialized thermoset resin, not a thermoplastic as in the ME technique. A third common AM approach can be generally characterized as Powder Bed Fusion (PBF). Here, a material in powder form, which can be a polymer or metal, is repeatedly spread into a thin layer and then fused by local application of heat, most commonly provided by a laser beam. Metal-powder-based processes operating in chambers filled with inert gas or in a vacuum are now able to produce functional components that are used in production for low-volume manufacturing.

Ongoing improvements in AM technologies have opened new possibilities for the designer to introduce creative features that make the product both functional and stylish at the same time. Making educational props, mechanical parts, decorative objects, architectural structures, and clothing pieces that fit the available space and follow the shape, curves, and surface of their mating objects has inspired many artists, engineers, scientists, car manufacturers, and spacecraft designers [134,135,136,137,138].

One shoe manufacturer has created shoes with 3D-printed midsoles, making it possible to produce custom-fitted local variations of support pressure [139]. Injection or compression molding manufacturing methods are not capable of fabricating such structures, while 3D printing made it possible to test over fifty different lattice designs to quickly arrive at the final design solution. They use a newly developed variant of the VP method called Continuous Liquid Interface Production (CLIP). It is able to manufacture parts at a much higher rate than was possible previously, making this method suitable for high-volume production.

Another example is the use of metal 3D printing (PBF technique) to make titanium eyeglass frames that can be made to custom-fit the face of the user; they provide a stylish appearance, are comfortable to wear, and are lighter but stronger than traditional designs [140]. Using 3D AM technology to make parts out of metal has grown exponentially since 2000, reportedly 875 percent for the last five years (since 2014), out of which 220 percent of growth was in the past two years (since 2017). A faster production rate, reduced postprocessing costs, topology optimization techniques, and the introduction of less expensive materials have been the contributing factors. This technology that has opened the door to new possibilities is the source of inspiration at the preprocessing (e.g., materials and their applications, process modeling, and sensitivity analysis to process parameters); processing (e.g., method of applying, bonding, curing the layers, and process parameters); and postprocessing (e.g., nondestructive inspection methods and model validation) levels.

CHAPTER 4

ELEMENTS OF GEOMETRY

This section discusses key concepts in the science of geometry. Examples include axioms—theories that connect geometry components through hypotheses and proofs—the all-true axioms that are accepted as laws without any questioning of their validity. The logic used in order to derive any of the axioms with attached controversy only enhances the reasoning as to how to pose questions, respond, and argue for or against the hypotheses. One can also argue against the outcomes and move forward with either accepting the consequences or coming up with ways of revising the past theories.

The story of the Earth's flatness, how it was rejected with its geopolitical and religious roots, is an example of an axiom that can be considered controversial. This concept that one may think has long been universally agreed upon is still being debated, even in a few academic environments. Still there are people who believe that the flat Earth is the center of the universe, even while our satellites are circling the Earth, providing an avalanche of images of our blue planet, and our space probes are exploring the solar system, beaming back to us close-up shots of Mars and Saturn's rings. Nevertheless, we are living in the age of persuasion when any idea can be easily disseminated, which can lead both to good and bad consequences, as is the case with any other human technology.

Axioms are clearly connected to geometrical elements such as points, lines, curves, planes, surfaces, angles, and features such as topology and symmetry. The concepts of form, fit, and function, introduced earlier, present a fresh outlook, introducing a Lean Six Sigma concept (to be explained later) that is mainly discussed and experienced in process

improvement projects. This demonstrates that the science of geometry combined with Lean Six Sigma methodologies is capable of shedding light on some geometrical concepts.

Straightedges, compasses, dividers, T-squares, protractors, parabolas, ellipses, and templates have been historically used to construct geometries of all levels of complexity from the simplest form—a straight line—to a most complex form—sacred geometries. Scientists and mathematicians have proven that natural patterns such as those of the chambered nautilus follow sacred geometries. These structures grow in a logarithmic and continuous fashion with a constant growth factor—also known as the *golden spiral*—forming a spiral shape. The Archimedean spiral is a special form of this 3D geometry projected onto a plane. There are different ways to obtain this shape. A close approximation of a *golden spiral* can be made by drawing quarter-circle arcs that connect the opposite corners of squares in a Fibonacci tiling (Figure 4.1). Imagine a rectangle made of twenty-one by thirty-four one-unit squares. There are a total of eight squares inside this rectangle with the square side lengths listed by following the spiral in a clockwise direction as: 21, 13, 8, 5, 3, 2, 1, 1. The spiral in Figure 4.1 was generated in COMSOL Multiphysics, and parameter definitions and geometry sequence steps for this exercise are provided in Appendix A. There are other natural phenomena that follow this pattern. For example, one of them is the *Whirlpool Galaxy* (NGC 5194) located in the constellation Canes Venatici at a distance of twenty-three million light years from us.

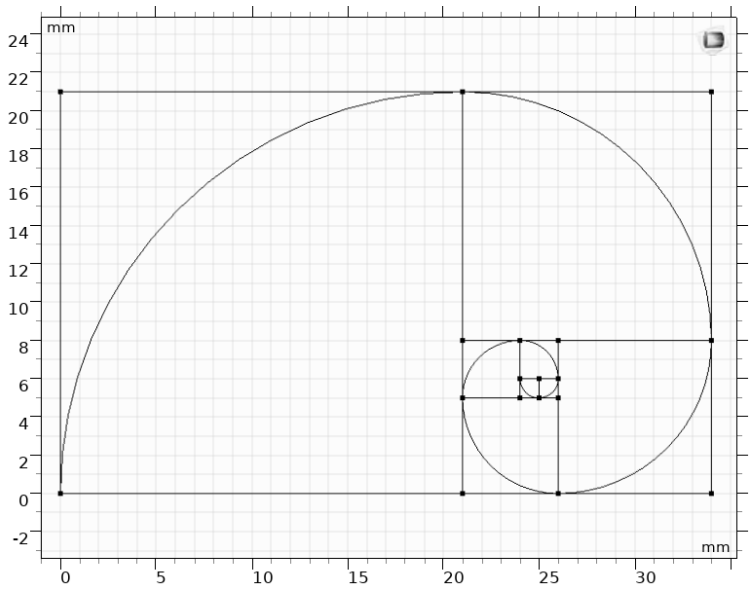


FIGURE 4.1. A *golden spiral* approximation with Fibonacci spiral (created in COMSOL Multiphysics).

4.1 AXIOMS

An axiom, also known as a *hypothesis* or a *postulate*, is an obviously true statement upon which all its successors rely. It commands itself as self-evident, such as the existence of a divine force in the world for believers. All future arguments are based upon the accepted axiom. Its literal meaning in Greek is that *which presents itself as evident or worthy of fit*. Therefore, an axiom is to be both evident and generally accepted—it should not cause any controversy. There are logical and nonlogical axioms. The formulae that are valid universally form logical axioms. An example is $x = x$. The definition of equality (=) is an agreed upon concept that is universally known; the meaning of equality needs to be established, however. On the other hand, nonlogical axioms are not always true on their own, and sometimes complementary axioms are needed to make them meaningful—they are theory- or domain-specific. Assume that there are two equal terms ax and ay ($ax = ay$), where a is a constant or successor function (a real or imaginary number or recursive function). It can be concluded that $x = y$ given the nonlogical axiom hypothesis—meaning it cannot be proven otherwise. In this scenario, the hypothesis or postulate may also be referred to as the assumption. Recall that, when solving a problem, the first question is: what are your assumptions and what are they based upon?

Let us clarify the concept of a nonlogical axiom with another example. Assume Alpha and Beta each boil water to make a cup of tea. They share their observations with Gamma. Alpha says that the water boiled in two minutes, reaching 100 °C, the suitable temperature to brew black tea. Beta interjects by stating that it took him about ten minutes to boil water. To determine who is telling the truth, Gamma reasons that the same physics apply to both processes, and so she asks Alpha and Beta to reveal the assumptions on which their conclusions had been reached. The first question would be whether the water was initially at room temperature. But even an affirmative answer to this can be further questioned: was this a room with no air conditioning in summertime Saudi Arabia or a room with no heating on a Canadian winter day? As you see, the assumptions and how they are presented make a difference in the specifics of arriving at certain conclusions.

Other assumptions that may need to be added to this nonlogical axiom are: (1) local air pressure; (2) water grade (e.g., use of heavy water—deuterium oxide—versus regular water—protium oxide); (3) existence of impurities or other elements such as iron and copper (resulting in increase of boiling temperature); (4) water mass; and (5) thermophysical properties of the test container (e.g., use of a copper

kettle versus a steel one); and (6) shape of the vessel in which the tests were conducted (e.g., having a larger heating surface area in one versus the other one). In other words, for Gamma to come to an accurate conclusion as to which one is correct, she should read complete and accurate test reports of the boiling processes conducted by Alpha and Beta as well as conduct interviews to provide a fair assessment. In this scenario, it turns out that Alpha and Beta both were telling the truth—both tests were conducted at the same time, in the same location, and under the identical environmental conditions. The difference was the initial temperatures: Alpha started from 80 °C, since there was previously boiled water in the kettle that cooled from 100 °C for two minutes while Beta used the tap water at room temperature (20 °C), which obviously required more time to boil. Alpha, Beta, and Gamma decide to celebrate their conclusionary victory with a perfect pot of green tea and biscuits. This example presents an axiomatic system in which a number of axioms work together to logically prove a theorem. This is the deductive form of mathematics.

Given the definition of an axiom and the said examples, George Orwell's statement in *Animal Farm* that "all animals are equal, but some animals are more equal" is not a logical axiom but a nonlogical one, since the comparison between rights is made between different animals (e.g., pigs and horses) [141]. Let us set aside the source of this statement, the immortal declaration, "All men are created equal," from the United States Bill of Rights and present this in a different light. These creatures are not equal in their appearances nor nature, so the hypothesis is inherently illogical. However, one may say that given equal conditions (specific theorized conditions or domains), they *can* be equal or *are* in fact equal. In this scenario, a smart donkey, a gifted pig, and a skillful monkey—assuming that they all know mathematical functions—are equal *enough* to work as a *calculator*.

The importance of understanding logical versus nonlogical axioms cannot be overstated, since there are numerous geometry-based hypotheses expressed. To prove or reject these theories, either valid assumptions are made and the conclusion is reached (e.g., the congruity of the two corresponding angles that have the same position with regard to the parallel lines which a transversal line intersects); or invalid assumptions are made under which clearly invalid conclusions are reached, and as a result it is determined that the assumptions are not correct, so the opposite assumption(s) is valid. For example, if the two angles in a triangle are equal, then the opposing sides facing those angles are also equal. This may be concluded by first assuming that the angles opposing two sides are not equal but the two sides are equal and proving that this assumption is not correct.

There are exceptions for which neither the logical nor nonlogical axioms apply. For example, you cannot conclude that if an entity is your enemy, the entity's enemy is your friend. It also cannot be concluded that three vertices not along the same line necessarily form an axis; this makes neither mathematical nor geopolitical sense. The sense beside the logical or nonlogical axioms is that they assist you not to accept "nonsense." The science of geometry provides a platform for a logical presentation of nonlogical axioms based on unproven theories, meaning that no logical hypotheses have been made to prove they are valid. One of these axioms is the theory of the Earth's flatness that will be discussed later.

4.2 POINTS

A point is the most fundamental part of geometry and is used extensively in axioms—it is a stand-alone entity not being built upon other objects. It has no dimensional characteristics such as length, area, or volume. Ordered pairs (x,y), triplets (x,y,z), or tuples $(a_1,a_2,a_3\ldots,a_n)$ represent a point in 2-, 3-, and n-dimensional Euclidian spaces. In 3D space, the numbers represent horizontal, vertical, and depth dimensions locating the point. While in general a point does not have any dimensions in physics, such as when defining an electron, a point representing the entity (i.e., electron) has mass. A function that is used as a multiplier to identify a zero value for a function anywhere but at a certain coordinate or under certain conditions in the space, also known as the *Kronecker delta*, represents an integral of one for the entire line it is associated with. This concept was introduced in 1928 by Paul Dirac, a British physicist, and so it is sometimes called the Dirac delta. There is also a type of geometry, associated with connection theory, that is point-free; it is built by means of regions rather than its components [142].

4.3 LINES

A line is the second most important part of geometry (after a point); a line forms the connecting element between the two points by a relationship which is a linear combination of the set of points (i.e., start and end point coordinates). A line does not have depth or width and only has a length; it is in other words the extrusion of a point along one dimension. Ancient mathematicians used the line as an abstract representation for straight objects that have no thickness or width.

There are also geometries other than Euclidean, where a circle can turn into a line. An example of non-Euclidean geometry is spherical geometry, in which it is convenient to describe features on the Earth's surface. Thus, the shortest line between two points on a sphere is known as a geodesic and is part of a great circle (a circle around the Earth of the greatest radius). This is the path followed by airliners traveling between distant points on our planet. On the other hand, if you just decided to fly on a constant heading (constant compass needle indication) from point A to point B, you would be following a Rhumb line, which would take you on a longer route.

4.4 CURVES

A curve can be thought of as a generalization of a line and, conversely, a line is a special case of curve. You can think of stretching a string tightly between your hands to form a line or letting it hang loose to form a curve. Zooming in sufficiently on any small segment along a curve will make it look like a line. A curve may be contained in a 2D plane or occupy a 3D space. A closed curve means that if one starts from any point on this curve and follows the curve in one direction, one will return to the starting point.

A curve can be described algebraically in Cartesian (or other) coordinates by a relationship in which there is only one independent variable. Thus, a curve can be a graphical representation of a mathematical function. For an x-y plane (2D), for example, a parabolic curve can be described by $y = ax^2$. Here, x is the independent and y is the dependent variable. A circle of radius R can be described by $x^2 + y^2 = R^2$. Either x or y can then be treated as independent, making the other variable dependent. If we were describing a curve in x-y-z coordinates, we would only be able to choose independently one of the (x,y,z) values while the remaining two would be the dependent variables defined by the functional relationship describing this curve in space.

4.5 PLANES

A plane is one of the fundamental geometrical entities. It is a flat 2D infinite surface, a special case of a generalized curved surface. It is also a special case of a subclass of surfaces known as *ruled surfaces*. Through any point on a ruled surface, one can draw a straight line which is contained in the same surface. A plane can be defined by any of the following: (1) three points which do not lie along a straight line; (2) a line

and a point which does not belong to this line; (3) two non-parallel intersecting lines; and (4) two parallel lines.

Any algebraic description of a plane in a 3D space will be an equation with two independent variables (representing the two degrees of freedom) and one dependent one. For example, a plane can be described in Cartesian 3D space by $ax + by + cz + d = 0$. Any pair from the (x,y,z) triple can be chosen to be independent and the remaining variable is then the dependent one.

4.6 SURFACES

Mathematically, a surface is a generalization of a plane (similar to a curve being a generalization of a line). Think of a sheet of plastic stretched tightly over a rectangular frame. Pressing it over a ball will stretch it, forming a curved surface from a plane. Conceptually, a surface can be infinite, such as in a paraboloid, or it can be confined by boundaries; it can have holes or be a boundary of a solid object, like a sphere.

Surface is perhaps the most philosophical of all the basic geometrical elements. There has been great debate over its existence; however, it is argued that surfaces present the first sights of objects. They are the outer layer, the shell, of an object. *What you see is what you get* is the very definition of the surface. It is filled with color and texture in its essence. In many cases it is defined with adjectives that present the sensations to the touch (e.g., smooth and rough) or how they are seen (e.g., shiny, opaque). It is a fair assessment to say that they are rooted in our perception of objects on atomic levels and their optical properties. The former is due to the fact that matter is made of atoms. Even though they may not be located in distinct layers, they may be considered as having organized layers—each layer having multiple atoms that are situated in raster patterns with respect to one another, forming a composite layer of atoms. The latter is easily comprehended as one becomes familiar with the science of thermal radiation and how what we see is the result of the light that is emitted by a light source. Given that any object absorbs, reflects, and transmits the received light, it may be seen as transparent or flat. The color also follows similar description, depending on the wavelength at which the light is being absorbed by the object—for example, it is seen as black (i.e., all light wavelengths are absorbed) or white (i.e., all light wavelengths are reflected). The color on its own may only embellish a part that is drawn as a geometrical component; however, when the same colors are used to shade the parts to show light reflection, they represent the perspective as a vital part of geometry. They make the three-dimensionality of the part quite obvious.

Any object's interaction with the surrounding world takes place through its surface. Think of the Sun's Coronal Mass Ejection (CME), which produces a very uneven surface that is still seen as a shiny golden flat object from the Earth. If a satellite approaches the Sun (e.g., the recently launched Parker Solar Probe) or any other powerful telescope observes its surface, one will see how uneven its surface is, much like oceanic waves interacting with wind.

This is why a surface is so fundamental a concept in our world—it represents human sensations in their purest form and how they interact with the surrounding world. As you guessed already, surfaces of different roughness also have different senses of touch but also sound; think of walking over the dry leaves on an autumn day. Surfaces also demonstrate diverse aerodynamic behaviors, and for that the surface drag is introduced as the force opposing the aircraft's forward motion. Spilling your milk, it becomes one with the surface it comes in contact with and there is no point to cry over it. That introduces the concept of an interface between two substances that is widely used in physics—it can be an interface between two fluids or a fluid and a solid.

4.7 ANGLES

The word *angle* originates from the Latin term *angulus*, meaning corner. An angle describes the inclination between two geometrical elements such as lines, planes, and surfaces. Angles are formed when these features intersect (or their extensions do). The study of angles is the foundation of trigonometry, where the relations within the polygons and triangles—a polygon's simplest form—are investigated. An angle formed by two intersecting planes is called a *dihedral angle*. When nonflat surfaces or curves intersect, the angle at the intersection line or point is defined by the tangent planes or lines at that line/point. This definition is applicable to spherical angles that are formed due to normals of the planes forming great circles. Four angles are formed when two features (e.g., lines, curves, surfaces) intersect. Two of these four angles that are opposing one another are called vertical (or opposite) angles.

Depending on its magnitude, an angle can be described as: (1) right angle (equating 90 degrees); (2) acute (smaller than 90 degrees); (3) obtuse (greater than 90 degrees); or (4) reflex (greater than 180 degrees). Two angles can be described as: (1) complementary angles (adding to 90 degrees); (2) supplementary angles (adding to 180 degrees); or (3) explementary or conjugate angles (adding to 360 degrees).

4.8 TOPOLOGY

Topology is a Greek term rooted in *topos* (place) and *logos* (study or reason). It defines the spatial properties that are preserved under deformations such as stretching or twisting. This property is related to the connecting points that follow a continuous pattern even if their distance is changed due to elongation, twisting, or bending. Breaking the material apart (tearing) or connecting spaces (gluing) changes the topology. The space has to fulfill connectedness (points are connected, all in one piece) and compactness (limit points are included, located at a constant distance from each other). This applies both to a Euclidian plane on which a straight line connects two points, and a non-Euclidian plane on which a parabola (curved line) connects two points. The space may be either continuous (e.g., real numbers) or discrete (e.g., integers), open (e.g., a box with one of six sides removed), or closed (e.g., a box with all closed sides). The connectedness and continuity make it possible for the points within the object to be measured. The dimension is the number of independent coordinates used to define the points belonging to the space, and it may be defined as one-dimensional (1D), two-dimensional (2D), three-dimensional (3D), and n-dimensional (nD). Fractal theory uses the latter scenario where n may be any real number. String theory is a special form of fractal theory that defines a space with ten or eleven dimensions for most important cases. For example, a Koch snowflake is associated with fractal theory with a fractal dimension 1.26, which is the ratio of $log(4)$ to $log(3)$. The four dimensions in physics expressing space (three dimensions) and time are special forms of geometric topology that are employed to represent a manifold being either mapped or embedded into another manifold.

4.9 SYMMETRY

Symmetry comes from a Greek term *symmetria* meaning consistency in dimensions because of the proportion and geometrical arrangements. Symmetry is a feature of geometry that identifies the sameness within the same sets of points (space); it is independent of the coordinate system. It can be about a line or a point in 2D or a plane or an axis in 3D. For example, rotating a half circle along the vertical axis passing through the half-circle's center results in a full circle; the two half circles are symmetrical with respect to each other—meaning that they resemble one another. Mathematically speaking, this resemblance

is related to the distances between the points and the ratios of the distances between the points located at certain distances from the center of the circle or the center of the rotation with which the symmetrical shape was made.

Symmetry is a useful concept to employ in order to reduce complexity when modeling physical systems. Imagine performing a heat transfer analysis on the Vitruvian Man when he is exposed to the Sun's radiation as it is sitting straight above his head, producing symmetrical heating load. Assuming that his entire body is modeled, and assuming that the available numerical tool is only licensed for 250,000 nodes, you are only able to represent his body by a number of small regions (finite elements) up to the number of available nodes. However, assuming that his body is symmetrical with the vertical symmetry plane crossing through his navel, it is possible to define a region with only half of the original elements—as if you were licensed for a 500,000-node model capable of producing a more detailed representation. Even if you kept the original element sizes, resulting in 125,000 nodes, you are able to reduce the solution time—sometimes by a factor of two. Therefore, use of symmetry is beneficial when setting up analytical and numerical models in both physical and geometrical domains, as it reduces demands on hardware resources and saves computation time. Interestingly, the left and right sides of a human face are not fully symmetrical, and that is why people have a *good* or *photogenic side* when taking photos. The more symmetrical a face is, the more attractive or naturally beautiful it appears.

4.10 COORDINATE SYSTEMS

In order to enhance user interaction with any geometry creation tool, either a dedicated one (e.g., Solid Edge) or as part of a FEM commercial software package (e.g., COMSOL Multiphysics), a good comprehension of geometrical concepts at their most fundamental level is paramount. Coordinate systems should therefore be understood clearly so that the user can apply them properly when creating geometries and mating assembly components.

In the following sections, equations are listed which define the relationships between 2D or 3D Cartesian coordinates and polar (2D), cylindrical (3D), and spherical (3D) coordinate systems. Polar coordinates are described by equation *(1)* through equation *(4)*, cylindrical coordinates by equation *(5)* through equation *(9)*, and spherical coordinates by equation *(10)* through equation *(15)*.

In these equations, x, y, and z are the Cartesian coordinates of the point located at the distance r from the axis of the cylinder or the center of the sphere. θ and ϕ are the planar angles. For example, in polar coordinates, which is a 2D coordinate system, the location is identified by $(x,y)=(r\,\mathrm{Cos}\,\theta, r\,\mathrm{Sin}\,\theta)$—Figure 4.2. In cylindrical coordinates, the same point is identified by $(r\,\mathrm{Cos}\,\theta, r\,\mathrm{Sin}\,\theta, z)$—Figure 4.3. In spherical coordinates, the same point is identified by $(r\mathrm{Cos}\,\theta\,\mathrm{Sin}\,\varphi, r\mathrm{Sin}\,\theta\,\mathrm{Sin}\,\varphi, r\mathrm{Cos}\,\varphi)$—Figure 4.4. The user may simplify the previous coordinates to $r(\mathrm{Cos}\,\theta, \mathrm{Sin}\,\theta)$, $r(\mathrm{Cos}\,\theta, \mathrm{Sin}\,\theta, z/r)$, and $r\,(\mathrm{Cos}\,\theta\,\mathrm{Sin}\,\varphi, Sin\theta\,\mathrm{Sin}\,\varphi, \mathrm{Cos}\,\varphi)$, for polar, cylindrical, and spherical coordinates, respectively.

4.10.1 Polar Coordinates

$$x = r\,\mathrm{Cos}\,\theta \tag{1}$$

$$y = r\,\mathrm{Sin}\,\theta \tag{2}$$

$$r = \sqrt{x^2 + y^2} \tag{3}$$

$$\theta = \arctan\left(\frac{y}{x}\right) \tag{4}$$

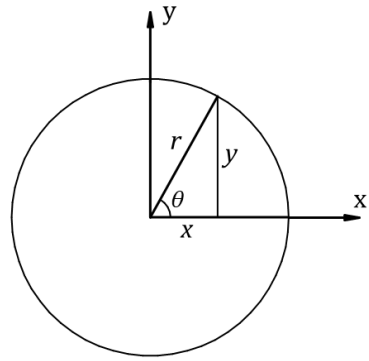


FIGURE 4.2. Polar coordinates (drawings created in Solid Edge).

4.10.2 Cylindrical Coordinates

$$x = r\,\mathrm{Cos}\,\theta \tag{5}$$

$$y = r\,\mathrm{Sin}\,\theta \tag{6}$$

$$z = z \tag{7}$$

$$r = \sqrt{x^2 + y^2} \tag{8}$$

$$\theta = \arctan\left(\frac{y}{x}\right) \tag{9}$$

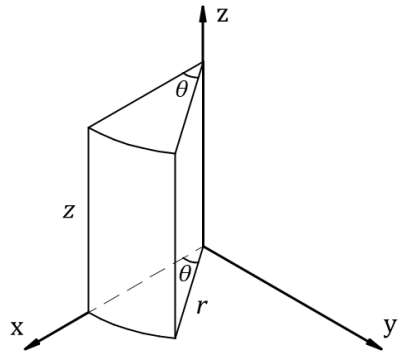


FIGURE 4.3. Cylindrical coordinates (drawings created in Solid Edge).

4.10.3 Spherical Coordinates

$$x = r\,\mathrm{Sin}\,\varphi\,\mathrm{Cos}\,\theta \tag{10}$$

$$x = r\,\mathrm{Sin}\,\varphi\,\mathrm{Sin}\,\theta \tag{11}$$

$$z = r\,\mathrm{Cos}\,\varphi \tag{12}$$

$$r = \sqrt{x^2 + y^2 + z^2} \tag{13}$$

$$\theta = \arctan\left(\frac{y}{x}\right) \tag{14}$$

$$\varphi = \arctan\left(\frac{x}{z}\mathrm{Sec}\,\theta\right) = \arctan\left(\frac{y}{z}\mathrm{Csc}\,\theta\right) \tag{15}$$

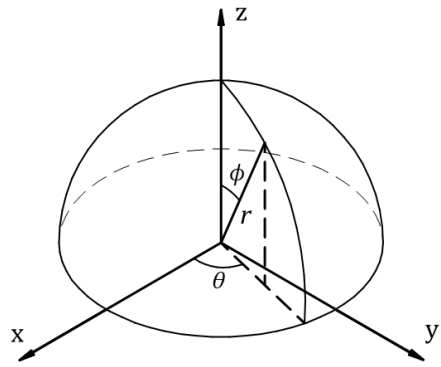


FIGURE 4.4. Spherical coordinates (drawings created in Solid Edge).

CHAPTER 5

NAVIGATING GEOMETRIES

The art of navigation should be able to identify vectors by their magnitudes and directions and create geometrical shapes in order to achieve what is desired—getting to the destination safely and on time. This art, which is a branch of applied geometry, found its way among the seven articles of the liberal arts known—until the ninth century—as weaving and tailoring, agriculture, architecture and masonry, hunting and martial arts, trade, cooking, and metallurgy. The modern interpretation of the liberal arts since the start of the twelfth century, also known as mechanical arts, had added medicine, theatrical arts, and navigation, replacing agriculture, cooking, and trade. In the beginning of the nineteenth century, the concept of the mechanical arts was separated from the fields associated with groups such as art performers, fine artists, and intelligentsia, and became a practical field now commonly known as engineering [143].

5.1 GEOMETRY OF THE EARTH

The conception of the flat Earth was held by the early Egyptians and Mesopotamians. Homer and Hesiod theorized that the Earth was a flat disk, floating on water, with an arched firmament separating it from the solid dome sky, with the heavenly planets being embedded in it—imagine a round pizza covered by a big blue fruit cake. Old East Asia believed the same except that the Earth was square versus flat. Some of the earliest conceptions of the spherical Earth are attributed to Pythagoras of Samos and Parmenides of Elea, Greek philosophers circa 600 BC. Eratosthenes of Cyrene, a Greek mathematician, geographer,

and astronomer, in 240 BC proposed that the Earth was spherical and approximated its circumference—reportedly for the first time. By the second century AD, Ptolemy had derived his maps from a globe and developed the systems of latitude and longitude. *Almagest* by Ptolemy, written in Greek, was translated into Arabic and only translated into Latin in the eleventh century from its Arabic translations, enriching the geometrical concepts [144].

The idea of the spherical Earth was widely accepted until the nineteenth century, when a historical uprising to the idea became popular, claiming that the dominant hypothesis during the Middle Ages was that the Earth was flat. Washington Irving, an American diplomat, historian, and writer, wrote in 1828 that Christopher Columbus, an Italian explorer, faced church opposition in 1492 while attempting to secure funds for his voyage around the globe, for they believed the Earth was flat. This was an attempt to show that there was a large gap between religion and science. The theory of the Earth being flat had other proponents, such as the believers of Zetetic Astronomy, a pamphlet that was published in 1849 by Samuel Birley Rowbotham, a British inventor and writer; he followed it with *Inconsistency of Modern Astronomy and Its Opposition to the Scripture*. The Zetetic Society was the result of this philosophy, founded in 1883 by Rowbotham, and his books sold thousands of copies.

Some contemporaries even offered rewards if someone could prove that the Earth was not flat. John Hampden offered a £500 wager (equivalent to £59,022 in 2019) if someone could prove that the Earth was round by demonstrating a convex surface of the Earth such as a body of water. Intrigued by the challenge, Alfred Russel Wallace, a British explorer, naturalist, geographer, and biologist, designed an experiment by setting up two objects located at the same height from the water—each situated at two ends of a canal at a 10-km (6-mile) distance—and observing them by a telescope mounted on a bridge at the same height above the water. He concluded that the objects were seen at different heights through the telescope and that this demonstrated that the Earth was convex at the selected portion of the canal. Hampden did not accept the verdict of the judge—the *Field* magazine editor who announced Wallace was the winner of the competition—and wrote a letter to the publications in which Wallace was an active member to denounce his findings; their associates shunned Wallace. Wallace in return launched multiple libel and slander suits against Hampden, costing Wallace more than the amount of the wager, and the controversy left him bitter and frustrated for many years [145].

William Carpenter, a British printer and author, was another main proponent for the flatness of the Earth, authoring *One Hundred Proofs*

the Earth is Not a Globe in 1885. His logic was due to the fall of the Nile in Egypt of only 1 foot in a 100-foot length. A friend of Carpenter, John Jasper, an American ex-slave, Baptist minister, and public speaker for Christianity, stated in his sermon *The Sun Do Move*, published in 1878, that if the Earth were not flat, the people on the other side would have moved with their feet upward, like flies on the ceiling inside a room. Joseph Woods Holden, an American politician and the son of William Woods Holden, an American politician and the thirty-eighth and fortieth Governor of North Carolina—appointed by the United States president, Andrew Johnson, in 1865 and eventually elected in 1868 (leader of the state's Republican Party during the Reconstruction Era)—was another proponent who delivered numerous lectures on the flatness of the Earth. There was a case in 1887 in Brockport, New York, where M. C. Flanders advocated adamantly for the flat Earth theory for three consecutive nights against the scientific community proponent to the sphericity theory. The five townsman judges voted for his theory at the end of the debate. It might have been something to do with the exhaustion that the judges were feeling.

Lady Elizabeth Blount founded the Universal Zetetic Society in 1893 in the United Kingdom, publishing a journal entitled *Earth Not a Globe Review*, stating that the Bible was the indisputable source on the natural world and proclaiming that a faithful Christian cannot believe that the Earth is a sphere. She had distinguished followers in the Trinitarian Bible Society at Trinity College in Dublin among archbishops. After World War I, interests in the flat stationary Earth declined. Joshua Slocum, an American seaman who sailed solo around the world for the first time, came across a tribe in South Africa believing the Earth was flat being led by a clergyman in 1898. Wilbur Glenn Voliva, an American evangelist who took over the Christian Catholic Church in 1906—a Pentecostal sect that had established a community in Zion, Illinois—was a passionate advocate for the flat Earth theory, stating that the missing airship Italia that disappeared during an expedition to the North Pole in 1928 fell off the edge of the world. Teaching the non-flat Earth was banned in Zion schools during his time [146]. In 2018 Yaël Nazé, A Belgian astrophysicist, analyzed a controversial PhD thesis proposed by a student at the University of Sfax in Tunisia, which defended the Earth's flatness along with a geocentric model of the solar system, claiming that the Earth is at the center of the Universe. The dissertation was not approved by the defense committee and had already been denounced in 2017 by professor Hafedh Ateb, a Tunisian founder of the Astronomical Society [147].

In conclusion, coming back to reality, one can state, with a bit of humor, that the Earth is neither a sphere nor a disk but a spheroid, also

known as an *ellipsoid of revolution*—so, nearly a sphere. As a result of this deviation from a perfect sphere, spherical relationships need to be corrected to make accurate navigation possible. Earlier in history, there was a period of scholarly argument on the exact nature of the shape of the Earth: whether it is oblate or prolate. The oblate Earth is a spheroid stretched over its Equator (like a tomato); the prolate Earth is a spheroid stretched along its poles (like a lemon). In 1687, Sir Isaac Newton determined that the Earth was oblate (flattening of 1/230); this was later confirmed by the measurements made in the eighteenth century. In 1669, Jean Picard, a French astronomer, measured the size of the Earth with reportedly high accuracy. In 1672, Giovanni Domenico Cassini, an Italian naturalized French engineer, mathematician, and astronomer, and later in 1713 his son Jacques Cassini, a French astronomer, measured the arc of the meridian and concluded that the Earth was prolate. The French Academy of Science then sponsored an expedition—also known as the French Geodesic Mission—in 1735 to investigate this finding, which resulted in confirming Newton's findings. In 1743, Alexis Claude Clairaut, a French mathematician, astronomer, and geophysicist, presented Clairaut's theorem, which stated the relationship between the gravity at the surface of the Earth to the position of the point based on the assumption that the Earth was oblate—a rotational ellipsoid. In the eighteenth century, different astronomers and geodesists studied the Earth's curvature, and as a result several ellipsoid models for the Earth were introduced—by Plessis in 1817, Airy in 1830, Bessel in 1830, Everest in 1830, and Clarke in 1866 [148,149].

5.2 NAVIGATING THE EARTH

To locate anything on our (nearly) spherical Earth, a geographic coordinate system is used, first conceived around the third century BC by Eratosthenes of Cyrene, who is sometimes called the "Father of geography." He essentially created the concept of geography as a field of study. Geodesy or geodetics is the division of geography that deals specifically with the Earth's measurement. In the system that we use today, latitudes and longitudes locate any point on the Earth's surface—latitudes with respect to the poles and longitudes with respect to the reference meridian, also known as the prime meridian (nearly identical to the Greenwich meridian). Longitudes vary from zero degrees at the reference meridian to –180 degrees westward and +180 degrees eastward, which is the dipole of the reference meridian. Latitudes vary from zero at the Equator to +90 degrees at the North Pole and –90 degrees at the South Pole. Lines of equal longitude are called meridians—semicircles joining

the North and South Poles. Lines of equal latitude are called parallels—forming circles around the Earth parallel to the Equatorial plane.

As a result of the Earth measurement efforts, three different kinds of latitudes have been introduced for a point "P" on the Earth's surface:

(1) geocentric (L_G)—angle between the Equator plane and the line passing through the center of the Earth;
(2) astronomical (L_A)—angle between the Equator plane and a plumb line due to gravity; and
(3) geodetic (L_N)—angle between the Equator plane and a normal to the tangent plane at "P" (Figure 5.1). Since the Earth is a spheroid, the normal to the local tangent at "P" does not pass through the center of the Earth, except at the Poles and the Equator. If someone uses the term *latitude* without specifying its type, the convention is to assume that it is a geodetic latitude.

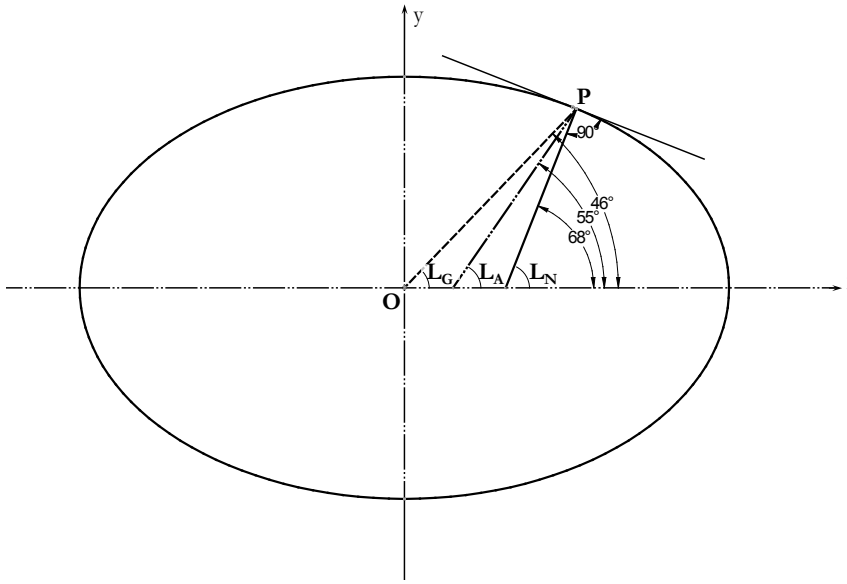


FIGURE 5.1. Geocentric (L_G), astronomical (L_A), and geodetic (L_N) latitudes (drawings created in Solid Edge).

5.2.1 Latitude

Latitude corresponds to the distance from south to north on the Earth's surface and varies from zero degrees on the Equator to +90 degrees at the North Pole and –90 degrees at the South Pole (Figure 5.2). Parallels are lines (circles) of constant latitude on the Earth's surface. When navigating in the Northern Hemisphere by observing Polaris, also known as

the North Star, with a sextant and using tables to correct for the height of the eye and atmospheric refraction, the latitude of the observer is the height of Polaris (in degrees) above the horizon.

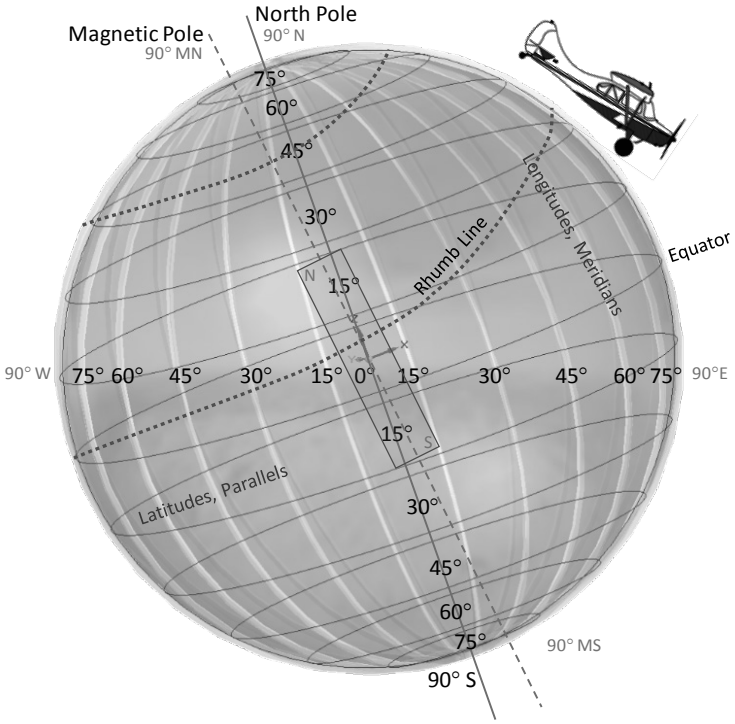


FIGURE 5.2. True and magnetic Poles, rhumb line, latitudes, and longitudes (drawings created in Solid Edge).

5.2.2 Longitude

Longitudes correspond to the distances to the east or west of the prime meridian, also known as the Greenwich meridian (Figure 5.2). They vary from zero degrees at the prime meridian to –180 degrees to the west and +180 degrees to the east. Meridians are lines (semicircles) of constant longitude and run from the South to North Pole. Without access to GPS or other modern navigation tools, longitude can be calculated based on the time of sighting. If not possible, you may employ a sextant to measure lunar distance, also known as lunar observation, which, along with a nautical almanac, can be used to calculate the time at zero-degree longitude. Marine chronometers only became available in the late eighteenth century and were not widely used until the nineteenth century. A mariner with a chronometer could check its reading using a lunar determination of Greenwich time.

5.2.3 Azimuth

Azimuth is from the Arabic term *al-samt* for *the direction*. The vector from a point of origin to a point of interest is projected to a reference plane. The angle between the projected vector and a reference vector on the reference plane is called the azimuth. It is the horizontal direction of a star, or another heavenly body in the sky, in celestial coordinates. In this scenario, the heavenly body is the point of interest, the location of the observer is the center of a reference plane with a 5-km radius at sea level, and the reference vector point located on this reference plate points to true north.

5.2.4 Zenith

The term *Zenith* is from the Persian (Iranian) astronomers, from the Arabic[1] word *samt-al-raas* meaning the *direction of the head*. The zenith is a point directly above a location on the sphere. This is a vertical direction opposite to the Earth's gravity, orthogonal to the local flat surface of the Earth [150].

5.2.5 Rhumb Line

A rhumb line, also known as a loxodrome, crosses all latitudes and longitudes at the same chosen angle (Figure 5.2). At every point along this line, one is moving along a constant bearing relative either to true or magnetic north direction.

5.2.6 Variation

A variation or magnetic declination is the angle in degrees between the direction indicated by the compass needle (which aligns itself to the local magnetic field lines) and the direction of the meridian passing through the same point and going toward the geographic North Pole. This angle depends on the geographical location and also changes slowly over time (two to three degrees per hundred years) as the North Magnetic Pole shifts its location.

Figure 5.3 shows the distance for one degree of longitude versus latitude—*Distance = 60 × Longitude Difference × Cos (latitude)*. It is seen that the distance per degree of longitude decreases with increasing latitude—as one moves closer to the Poles, the meridians converge and thus the distance between them decreases. Note that at the Equator (zero-degree latitude), the distance is exactly 60 NM. This is not accidental—a nautical mile (NM) was originally defined as one

[1] The reader is encouraged to review the related reference [150] for clarification that although the word is not a Persian (Farsi) word, after the Arab conquest, some words were borrowed from Arabic by the Persian (Iranian) scholars.

minute (or 1/60) of one degree of latitude. At the Equator, a distance of one degree of latitude is equal to that of one degree of longitude.

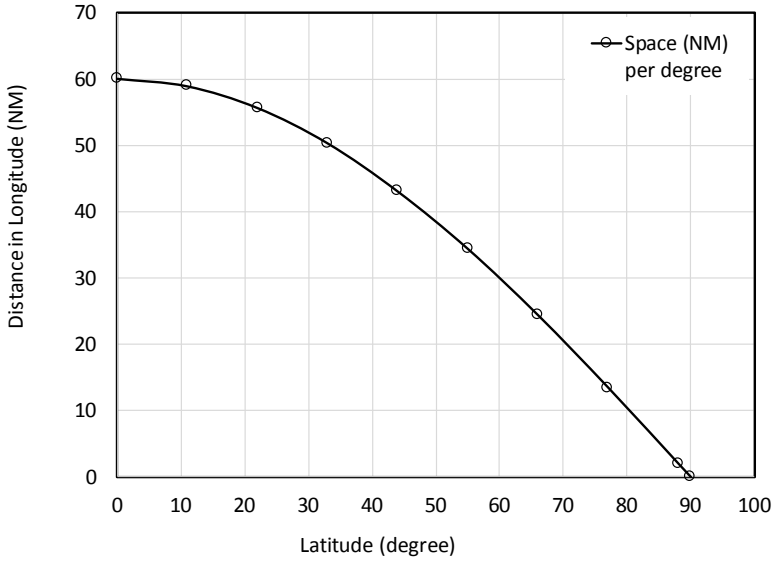


FIGURE 5.3. Longitude distance per one degree versus the latitude.

To understand navigational charts better, you should know the coordinate systems used on these maps. Figure 5.4 shows the azimuth α, elevation ψ, and zenith ζ angles. Based on the projection method—cylindrical, conical, or azimuthal-planar—the main coordinate systems (polar, cylindrical, and spherical) are defined.

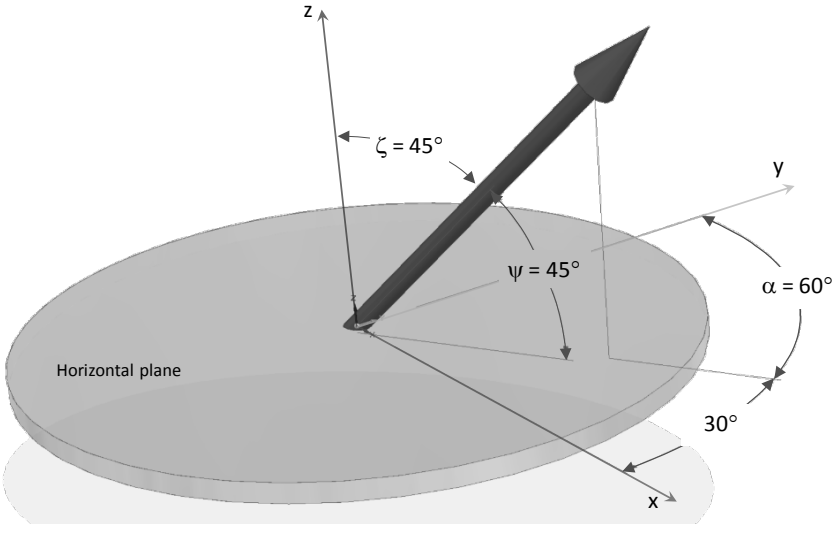


FIGURE 5.4. Azimuth, elevation, and zenith angles (drawings created in Solid Edge).

5.3 VECTOR ALGEBRA

A vector has two characteristics: a magnitude and a direction in space. A scalar, by contrast, only has a magnitude and a sign. For example, an airplane's movement through the air can be described by the true airspeed magnitude (e.g., 100 kt) and a heading (e.g., 60 degrees), making it a vector quantity represented by two scalars. Temperature, on the other hand, is a scalar, since it is identified only by its magnitude and sign. Any group of scalars (more than one) may be treated as a vector. For a 3D space, a group of three independent dimensions describes a vector in this 3D space. Depending on the coordinate system used, these three numbers can be the three Cartesian coordinates *(x,y,z)* or a distance and two angles in spherical coordinates (r,θ,ϕ), and so on. If the vectors are confined to a plane, or any 2D surface, a set of two scalars describes a vector on that surface.

Vectors can be added and subtracted. This can be shown graphically. In Figure 5.5, vector $\overrightarrow{AB}$ is the sum of vectors $\overrightarrow{AC}$ and $\overrightarrow{CB}$. This is written in vector notation as $\overrightarrow{AB} = \overrightarrow{AC} + \overrightarrow{CB}$. Any of the three vectors in this expression may be defined in terms of the other two. For example, $\overrightarrow{CB} = \overrightarrow{AB} - \overrightarrow{AC}$. This can be deduced from rearranging the previous equation by moving vector $\overrightarrow{AC}$ from the right side to the left side. The visualization may be made knowing that $-\overrightarrow{AC} = \overrightarrow{CA}$. If the vectors $\overrightarrow{AB}$ and $\overrightarrow{CA}$ (opposite direction to $\overrightarrow{AC}$) are added, vector $\overrightarrow{CB}$ is obtained, of the same magnitude and direction as $\overrightarrow{CB}$ shown in Figure 5.5.

Vectors can also be added and subtracted numerically. The user can express the vectors in Figure 5.5 in terms of their Cartesian components: $\overrightarrow{AC} = (X'', Y'') = X''i + Y''j$, $\overrightarrow{AB} = (X', Y') = X'i + Y'j$. In this relation, i and j represent unit vectors along the x- and y-coordinates, respectively. Thus, when multiplied by the scalars representing the vector's Cartesian components, they result in vectors equivalent to those written as a set of numbers in brackets. The difference of two vectors may be calculated as: $\overrightarrow{AB} - \overrightarrow{AC} = (X' - X'', Y' - Y'')$. An inner (or dot) product of two vectors gives a scalar. For the given example, it would be $\overrightarrow{AB}.\overrightarrow{AC} = (X'X'', Y'Y'')$.

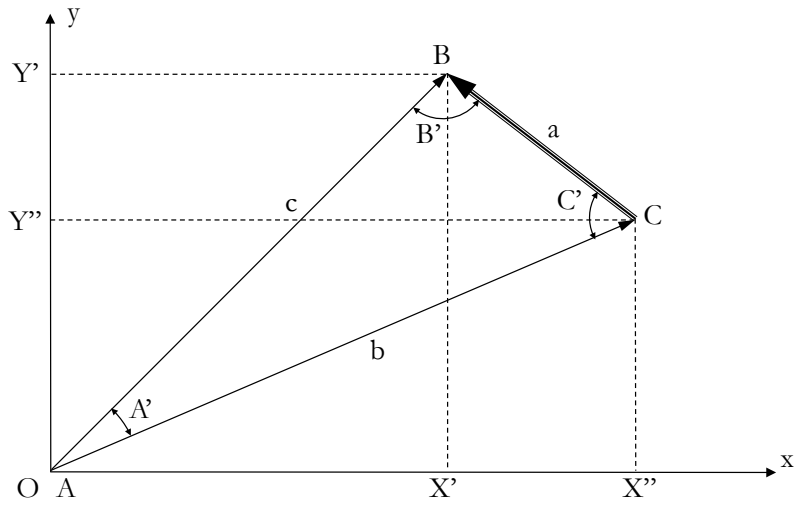


FIGURE 5.5. Adding vectors (drawings created in Microsoft PowerPoint).

A cross product is a vector itself and can only be calculated in three dimensions. This vector points in the direction normal to the plane in which the two input vectors lie. To find this direction, use the *right-hand rule*, where the right hand's four fingers curl in the direction from the first to the second vector and the thumb points in the direction of their cross product. Given two vectors $\overrightarrow{AB}$ and $\overrightarrow{AC}$ in an *x*-*y* plane with the *z*-component equal to zero, we obtain the cross product as $\overrightarrow{AB} \times \overrightarrow{AC} = (X', Y', 0) \times (X'', Y'', 0) = (0, 0, X'Y'' - X''Y')$. Observe that the cross product has zero *x*- and *y*-components (first two numbers in the set of three). Since the two input vectors are in the *x*-*y* plane, the cross product points along the *z*-coordinate is normal to the *x*-*y* plane, and so only has the nonzero *z*-component (the third number).

The following diagrams demonstrate the principles of vector algebra required for navigating geometries. Table 5.1 summarizes the relationships between vectors and scalars as well as vector inner product (also known as dot product) and cross product presented in Figure 5.5 and Figure 5.6. In the table, vector $\overrightarrow{AC}$ is the summation of vectors $\overrightarrow{AB}$ and $\overrightarrow{BC}$. Each vector is identified by its components normal to its three main coordinates (defined in a 3D space)—the coordinates are assumed the same for this example. Vectors may be multiplied by scalars (e.g., *C*) by multiplying each component by the scalar. Addition or subtraction of the vectors is achieved by adding or subtracting the corresponding components. The vector multiplications, either in the form of dot or cross products, follow the rules presented in Table 5.1.

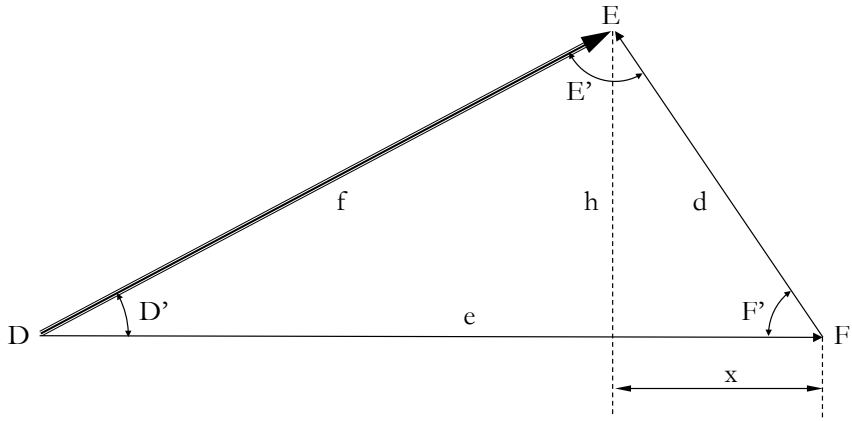


FIGURE 5.6. Law of cosines (drawings created in Microsoft PowerPoint).

TABLE 5.1. Vector operations.

Case	Vector	Scalar	Scalar Product	Inner Product	Cross Product	Angle (°)
1	$\overrightarrow{AC} = X''i + Y''j + Z''k$	C	$C(\overrightarrow{AC})$	$\overrightarrow{AC}.\overrightarrow{AB} = \left\|\overrightarrow{AC}\right\|\left\|\overrightarrow{AB}\right\| Cos\,\theta = X'X'' + Y'Y'' + Z'Z'' = X''X' + Y''Y' + Z''Z'$	$\overrightarrow{AC}\times\overrightarrow{AB} = \left\|\overrightarrow{AC}\right\|\left\|\overrightarrow{AB}\right\| Sin\,\theta = \begin{vmatrix} i & j & k \\ X' & Y' & Z' \\ X'' & Y'' & Z'' \end{vmatrix} = i(Y'Z'' - Z'Y'') + j(Z'X'' - X'Z'') + k(X'Y'' - Y'X'')$	$A' = arctan\left(\frac{X'}{Y'}\right) - arctan\left(\frac{X''}{Y''}\right)$
2	$\overrightarrow{AB} = X'i + Y'j + Z'k$	C'	$C'(\overrightarrow{AB})$	$C(\overrightarrow{AC}).C'(\overrightarrow{AB}) = CC'\left\|\overrightarrow{AC}\right\|\left\|\overrightarrow{AB}\right\| Cos\,\theta = CC'\begin{pmatrix} X'X'' + Y'Y'' + \\ Z'Z'' \end{pmatrix} = CC'\begin{pmatrix} X''X' + Y''Y' + \\ Z''Z' \end{pmatrix}$	$\overrightarrow{AC}\times C'\overrightarrow{AB} = CC'\left\|\overrightarrow{AC}\right\|\left\|\overrightarrow{AB}\right\| Sin\,\theta = CC'\begin{vmatrix} i & j & k \\ X' & Y' & Z' \\ X'' & Y'' & Z'' \end{vmatrix} = CC'\begin{bmatrix} i(Y'Z'' - Z'Y'') + \\ j(Z'X'' - X'Z'') + \\ k(X'Y'' - Y'X'') \end{bmatrix}$	$B' = arctan\left(\frac{X'' - X'}{Y' - Y''}\right) + arctan\left(\frac{X'}{Y'}\right)$
3	$\vec{R} = \overrightarrow{BC} = \overrightarrow{AC} - \overrightarrow{AB} = Xi + Yj + Zk = (X'' - X')i + (Y'' - Y')j + (Z'' - Z')k$ $a = \sqrt{b^2 + c^2 - 2bc\,Cos(A')}$	C''	$C''(\overrightarrow{BC})$	$C''(\overrightarrow{AC} - \overrightarrow{AB}) = C''\overrightarrow{AC} - C''\overrightarrow{AB}$	$C''\times(\overrightarrow{AC} - \overrightarrow{AB}) = C''\overrightarrow{AC} - C''\overrightarrow{AB}$	$C' = arctan\left(\frac{Y' - Y''}{X'' - X'}\right) + arctan\left(\frac{Y''}{X''}\right) = 180 - A' - B'$

Case 3 in Table 5.1 presents the law of cosines, from which you are able to find any angle of a triangle knowing dimensions of its three sides. Note that this law applies in many geometrical scenarios; therefore, it is worth investing some time in learning it and perhaps how to derive the relationships—equation *(16)* through equation *(18)*. Equation *(19)* demonstrates the inner product of the two vectors. Relations for right-angle triangles along with the Pythagorean theorem are employed to derive these relationships. The Pythagorean theorem states that in a right triangle, the square of the hypotenuse (the side opposite to the right angle) equals the sum of the squares of the other two sides.

$$h^2 = f^2 - (e - x)^2 = (d\,SinF')^2 \qquad (1)$$

$$x = d\,CosF' \qquad (2)$$

$$f^2 = d^2 + e^2 - 2ed\,CosF' \qquad (3)$$

$$\overrightarrow{FE}.\overrightarrow{FD} = \left|\overrightarrow{FE}\right|\left|\overrightarrow{FD}\right| Cos(F') \qquad (4)$$

CHAPTER 6

CREATING GEOMETRY

This section briefly introduces several popular computer programs which can be employed to create model geometry. It is possible to represent the most complex curves and diagrams related to physical relations by coding the formula in an advanced scientific compiler such as Fortran, C++, or Borland Pascal, each having their own features and library source codes, capable of being adapted to the input and output types. For example, the thermodynamic relations applicable to moist air can be analyzed and coded to replicate a *Skew-T Log-P* diagram, a thermodynamic chart used for weather forecasts. This diagram can be employed to determine pressure, temperature, cloud base layers, convective layers, and precipitation as functions of altitude.

It is also possible to employ an event-driven program such as Microsoft Visual Basic®, in which the user may record repetitive actions (similar to the macro-creation concept), view the processes within the decoded steps, remove the noise, eliminate the extra steps, and modify them to meet custom specifications. It is also possible to use the looping features in this algorithm to automate the repetitive steps. For example, a large amount of collected data including household information such as age, gender, height, weight, residence type, and energy usage (electricity and hydro), which are identified by connected strings consisting of integer codes—with no sensible meaning to a viewer—may be decoded, sorted into categories, and analyzed for their specific counts, behaviors, relations to one another, statistical significance, and trends.

Any of the said techniques in creating geometry or charts based on physical phenomena or as stand-alone entities require specialized knowledge in two scientific fields (software programming and physical science, in which scientific theories are paramount). For example,

for the previous examples, the knowledge of mechanical engineering thermofluids is required. Given that coding requires step-by-step verification of the written program, it may be laborious as well, which makes it a hindrance for getting the work done, especially where coding resources (time and specialization) are not available. Therefore, commercial software packages have been developed that are already verified (or expedited to have the verification process completed to get licensed) and available for field specialists who wish to spend less time on coding the well-known concepts and more time on developing new ideas and advancing their field of interest. Note that these packages can be either the ones that create parts (i.e., CAD tools) or the ones used to import or analyze CAD-made (or imported) models.

To create the geometry, a user can either interact with the software via a GUI or by creating a script (a program) which defines the geometry by a sequence of commands. For example, MATLAB by MathWorks can create geometry by using a script. Another popular FEM analysis software, ANSYS®, can receive input in the form of ANSYS Parametric Design Language (APDL) which, among many other capabilities, can create model geometry via a dedicated script language. Using APDL script, the user may take either the bottom-top or top-bottom approach in order to carry out the preprocessing (geometry creation, meshing, applying initial and boundary conditions), analysis (solving for transient or steady cases), and postprocessing (visualizing results by means of diagrams and plots).

The user in this scenario is capable of fully interacting with the program in an efficient way, depending on the level of their understanding of the program and process, and specifying the dimensions of the geometrical components, number of nodes or elements, element types, and the exact location at which the boundary and initial conditions are to be applied, along with the temporal and spatial steps. If the user is to run the program multiple times to change the conditions or the size of the parts, this approach minimizes the likelihood of human error due to repetitive command entry. Having said that, the suitability of the programmed approach will depend on the particular problem that needs to be addressed. If only one or few model changes are expected or if the problem is less complex, a programmed geometry generation approach will likely not be the most efficient technique.

The GUI offers a more user-friendly approach (with a less steep learning curve) to creation of the desired geometrical features. For example, ANSYS in addition to APDL script input offers a GUI geometry creation option as well. The GUI is equipped with top and side bars presenting the pinned commands, which are categorized and clustered according to their functionalities. For example, if the user wishes to create 2D shapes, they can access the 2D-geometry-related menu where related

features such as circles, triangles, and so on may be selected. In some commercial software packages, the user is required to make up their mind regarding the geometrical dimensions they are interested in (e.g., 2D versus 3D) before creating a new model. COMSOL Multiphysics is one of these software packages, making it possible for the geometry and physics to be intertwined in a transparent process. Having said that, it is also possible to introduce new models with different dimensional configurations at any stage before, during, or after the analysis is completed.

If the geometry is complex, the user may decide to create it using a commercial package that is dedicated to the modeling of parts and assemblies. These packages may be either mid-level (e.g., Solid Edge by Siemens PLM Software and SolidWorks by Dassault Systèmes) or more advanced ones (e.g., CATIA—Computer-Aided Three-dimensional Interactive Application by Dassault Systèmes—which is a multi-platform software suitable for Computer Aided Design—CAD—Manufacturing—CAM—and Engineering—CAE). After creating the geometry in one of these dedicated packages, the user will normally save it to the hard drive and then import it into the commercial package in which the physics analyses are to be performed (e.g., ANSYS or COMSOL Multiphysics). The physics-analysis software is usually capable of opening only a few specific file formats, such as **.step*, **.iges*, or **.stl*, and thus the file created by the dedicated solid modeling package must be exported in a particular compatible format.

It should be noted that although the geometry created in the modeling package may appear to be perfect, it in fact may not meet the requirements of the analysis program into which it is to be imported. For example, when analyzing geometries in COMSOL Multiphysics, the connecting curves and lines are to be perfectly linked, meaning that if the user mistakenly missed making proper connections in the original model, the imported geometry will lack the geometrical features such as planes or surfaces. It is possible though to select a tolerance setting for the software into which the geometry is to be imported to eliminate these errors.

When representing geometry models, the CAD tool may use wireframe, surfaces, and solid models in combination with manifold and non-manifold topologies—also known as *hyper-modeling*. Models can be created and revised using Boolean operations such as addition and subtraction. Features such as extrusions as well as surfacing, skinning, and thickening are possible in addition to transitioning at the corners such as filleting, tapering, and blending. The parts can be imported into an assembly and joined to the master part by means of relationships such as parallelism, alignment, mating, and tangency.

The common feature of all CAD commercial software packages is their employment of a geometrical modeling kernel which performs the

3D solid modeling functionality in the package. Parasolid® is a geometric kernel originally developed by Shape Data Limited in 1985. It is currently owned by Siemens PLM Software (after Siemens acquired the UGS Corporation) and is licensed by numerous other companies that are developing software employing 3D geometry representation. This list includes SolidWorks, Solid Edge, COMSOL Multiphysics, and other companies in the domain of Mechanical Computer Aided Design (MCAD).

The ACIS modeler was the kernel originally developed by Spatial Corporation (né Spatial Technology)—the successor to Romulus (an earlier kernel used for boundary representation of solid models). Spatial is now part of Dassault Systèmes. It is not known what exactly *ACIS* stands for. Some say it stands for *Alan, Charles, Ian, and Spatial*—the original creators and their company—but the creators themselves point to a character in Greek mythology. This kernel is used in several CAD, CAM, CAE, Architecture Engineering Construction (AEC), Coordinate Measuring Machine (CMM), 3D animation, and shipbuilding technologies. AutoCAD is perhaps the most well-known user of this kernel.

Some CAD tools work with third-party developers to introduce new functionalities and applications such as FEM modeling tools, tolerance checking, and circuit layout. Occasionally, they may license their modeling capabilities to other CAD software vendors [151,152].

Most commonly, the 3D part models are created by a sequence of steps, where a new feature is created at each one. These features are typically created from 2D sketches by extrusion to add or remove material. Extrusion means sweeping the sketched profile by either: (1) following a straight line along the direction normal to the sketch plane; (2) following any curve; or (3) revolving the profile about an axis. Thus, a cylindrical hole, for example, is a material-removing extrusion of a circle (2D sketch) along a straight line.

For any CAD tool, it is possible to generate geometry using its drafting (2D orthographic view), solid modeling, and assembly capabilities and export this geometry to other commercial software packages such as COMSOL Multiphysics. The user can activate or deactivate the parts within an assembly and import only the ones that are to be analyzed [153].

To make it easier to visualize complex designs by all involved in the process, the final design may be rendered to create more realistic-looking parts with their surface appearance determined by the part's material (e.g., metal, glass, plastic, or wood) and surface characteristics (e.g., roughness and reflectivity). It is possible to show the parts and assemblies as see-through objects using wireframes or solids not only in the 3D view but also in any one of the principal planes. The user can define the construction lines (when creating sketches) and layers so that they do not appear in the final sketch models or include certain

features (e.g., lines, planes, and dimensioning) within a given model phase creation so that the phase may be activated (turned on) and deactivated (turned off) when manipulating the entire model. It is possible to include the features (such as lines and vertices) of one layer within the other one to ensure consistency between the parts or simply save time creating the features.

When working with geometries, the analyst should take advantage of the available software features to more clearly represent the components and to work with them more efficiently. Most software packages allow one to generate and save different views of the geometry. The user may add notes, dimensions, and tolerances in the chosen system of units (e.g., SI versus Imperial). It is also possible to hide or show these features or change their size, font, and color as required. The created geometry may be outputted as a printed object, image, or file, which then may be embedded within a presentation as a stand-alone entity, an object that interacts with the original environment—to demonstrate variations due to the input parameters—or imported into other commercial software packages to perform further analyses (e.g., COMSOL Multiphysics). All the mid-level CAD tools are capable of exchanging geometries and data with other mid- and high-level CAD tools by exporting to one of the nonproprietary formats. However, exporting in a nonproprietary format usually will result in loss of the model's feature information. This makes any manipulation of this geometry more challenging unless a synchronous approach is used (as in Solid Edge software).

6.1 MATLAB

MathWorks is a company that specializes in developing software for mathematical computation; its most well-known products are MATLAB (released in 1970 as a free tool for academics) and Simulink (released in 2002) for data analysis and modeling. Subsequently, the code for MATLAB was rewritten in C and MathWorks was founded in 1984, with the first MATLAB release in 1985. MATLAB is the environment for programming and visualization of physical and statistical models and data. MATLAB software package comprises the core application and a set of optional toolboxes dedicated to various specialized applications. There are eight toolboxes in the category of Math, Statistics, and Optimization (listed under the product family); others are listed under application products (e.g., Image Processing and Computer Vision). The latter consists of twenty-two toolboxes that supplement the core module. Curve Fitting and Data Acquisition groups of functionalities are available under the similarly named licenses (Math and Test group).

Simulink is an environment made of components that are graphically interacting so that physical models on the system and subsystem levels can be simulated. They have a variety of applications and are used in number of industries, including aerospace and automotive. It is possible to use other family products in order to identify and troubleshoot problems related to runtime error (Polyspace), as well as add-on modules to drag-and-drop building blocks to model discrete events (SimEvents), and modeling reactive systems using control logic tools with the possibility for hierarchical presentations, parallelism, and history (Stateflow) [154,155,156,157,158,159].

With MATLAB the user may import, export, or save images to any of the formats listed in Figure 6.1 and Figure 6.2. Additionally, a geometry model can be created in MATLAB and viewed in the browser without the need to have MATLAB installed on the computer—Figure 6.3.

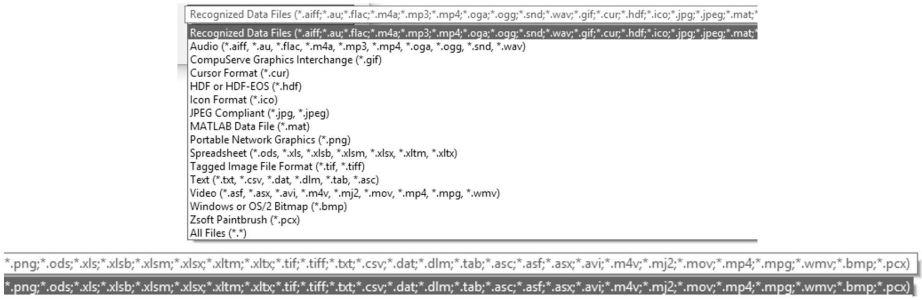


FIGURE 6.1. File formats importable to MATLAB.

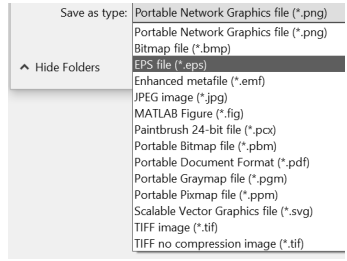


FIGURE 6.2. Image formats that may be created in MATLAB.

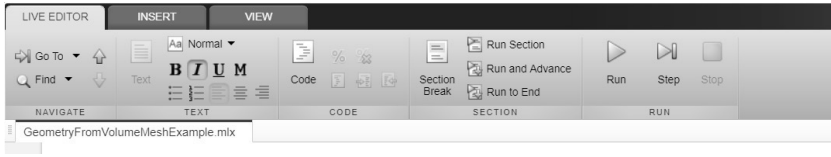


FIGURE 6.3. Geometry creation from volume mesh using MATLAB live editor.

In MATLAB the user may create the geometry by the following means:

(1) using Constructive binary Solid Geometry (CSG) models, which entails combining basic geometrical shapes to create more complex parts. The basic shapes may be cylinders and rectangular cuboids along with their derivatives (i.e., ellipsoids and polygons). The Partial Differential Equation (PDE) modeler is used to create the individual pieces and then these pieces are related to each other by different means of unification such as parallelism in the workspace;
(2) using command line functions (*decsg* function) by specifying the commands specific to each feature and inputting the required set of parameters (formula)—Figure 6.4 and Figure 6.5; and
(3) special functions that may be identified and adopted for a certain geometry, which then may be executed in a parametric fashion—Figure 6.6 and Figure 6.7.

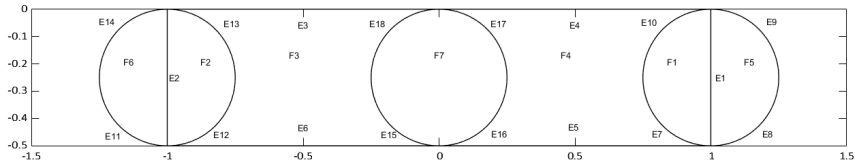


FIGURE 6.4. Geometry creation using circular and rectangular operations created in MATLAB.

```
>> C1 = [1
1
-0.25
0.25];
C2 = [1
-1
-0.25
0.25];
C3 = [1
0
-0.25
0.25];
C1 = [C1;zeros(length(rect1) - length(C1),1)];
C2 = [C2;zeros(length(rect1) - length(C2),1)];
C3 = [C3;zeros(length(rect1) - length(C3),1)];
gd = [rect1,C1,C2,C3];
ns = char('rect1','C1','C2','C3');
ns = ns';
sf = '(rect1+C1)+C2+C3';
[dl,bt] = decsg(gd,sf,ns);
pdegplot(dl,'EdgeLabels','on','FaceLabels','on')
xlim([-1.5,1.5])
axis equal
```

FIGURE 6.5. Geometry creation commands (circular and rectangular operations) created in MATLAB.

The appropriate method of geometry creation may be selected depending on the level of the model complexity. Among the approaches listed previously, the suitable level of geometry complexity increases from the first to the third method listed.

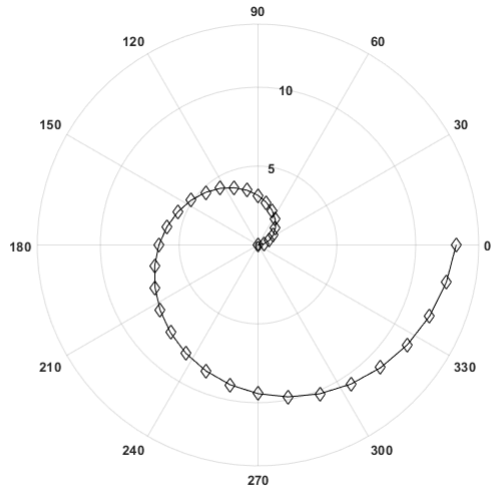


FIGURE 6.6. Geometry creation using radius values at equally spaced angles created in MATLAB.

```
>> theta = linspace(0,2*pi,25);
rho = 2*theta;
p = polarplot(theta,rho);
p.Color = 'black';
p.Marker = 'diamond';
p.MarkerSize = 8;
```

FIGURE 6.7. Geometry creation commands using radius values at equally spaced angles created in MATLAB.

For example, in the first method, one may click and drag to generate the geometrical features inside the workspace and the user may view the creation as the work progresses, making it possible to make revisions directly. It is also possible to confirm if the relationships between the geometry features are defined properly; for example, if the parts are connected at the desired points and lines. However, only aforementioned basic shapes are available with no possibility to represent more complex shapes on the edges or subdomain labels. In MATLAB, labels are the identifying features of the geometrical shapes. For example, the diagram's title is a label, and *xlabel* (representing the *x*-coordinate title for the diagram), *ylabel* (representing the *y*-coordinate title for the diagram), and *legend* (identifying the individual plots in a single diagram) are as well. The second method is capable of controlling all

the characteristics associated with the defined basic shapes; however, the user interface is not as friendly and the model is not viewable as it is being created—Figure 6.4. In this scenario it is not possible to change nodes or vertices nor subdomain labels, and only basic shapes are available. The third method may be employed to create any geometrical shape; furthermore, the node and subdomain labels may be identified—Figure 6.6. Nevertheless, since the user is interfacing with the functions, the instantaneous visual feedback is not possible, and multiple iterations may be required to obtain the desired shape. The user must examine all the functions and their capabilities to identify the ones that will be able to accurately represent the desired features [160].

It is also possible to create geometry from the mesh by importing the model and identifying the associated elements and nodes—Figure 6.8, Figure 6.10, and Figure 6.12. The user is to first import (load) the mesh into the workspace—this activates this feature as a built-in option that is available in MATLAB, identify the number of nodes (e.g., *n*), and then associate a label to the elements (e.g., *tet*). In other words, the user is informing the program that they are interested in creating a model from a mesh—so that the program gets prepared. The following analogy in order to clarify this procedure is presented. The guest is coming; her name is geometry; she is interested in pasta *tet* (elements' label); she likes spaghetti (*tetmesh*); and she eats 250 g (nodes) of food for supper. Then the user asks the host to cook (create *pde*) food (*tetmesh*), ensuring the said constraints. The next step is for the host to mix the ingredients, let the fire heat source (computer processor) and food ingredients (mathematical functions and input parameters) mix food ingredients (data processing) to view the food (*tetmesh*) so that the guest (geometry) is received respectfully (consistent initial conditions) as planned [161].

In the previous example, *tet* represents a mesh made from tetrahedrons (pyramids with triangular base), each consisting of four vertices, six edges, and four triangular faces. *pde* is a PDE solver (*pdepe*) available in MATLAB. It solves initial-boundary value problems related to the systems of parabolic and elliptic equations in 1D transient scenarios. There exists at least one parabolic equation in the system. The *pdepe* solver transforms the Partial Differential Equations (PDE) to the Ordinary Differential Equations (ODE) using specific user-defined nodes. Time integration is then performed using the *ode15s* solver. It is therefore possible to handle Jacobians (first-order partial derivatives) of vector functions. Using *ode15s* capability, both time and step sizes can vary dynamically. In this method, the mesh should also be consistent and fine enough so that the initial and discretization conditions become consistent; otherwise, a discretization error arises, which is similar to errors seen in other commercial software packages performing a variety of analyses. Parabolic equations seem to be more compatible with this methodology.

One can also generate and edit the code for the geometries created using any of the said methods (Figure 6.9, Figure 6.11, and Figure 6.13). The user can thus interact with the program at the lowest level by defining variables (variable data that can be used for inputs or generated as outputs) and parameters (constants), and introducing iteration loops, functions, and mathematical formulae into the program. The output text format (font and color) can also be modified as desired. The code generation feature is similar to creating a macro from the steps taken in order for the geometry or diagram to be generated. Reading through any of the said generated codes, you find that there are multiple building blocks that are presented in an organized fashion. The function is defined with an identifier with the input variable values listed within the parentheses. The figure command is executed to generate the desired shape. The color map (gray versus color scales) is identified. The line or surface plot is defined that may include the formatting data such as color and thickness of the lines and other features. It is then possible to include the axes and plot data in the desired locations. If the exact locations of the vertices are known, you may enter the data in the form of scalars or vectors.

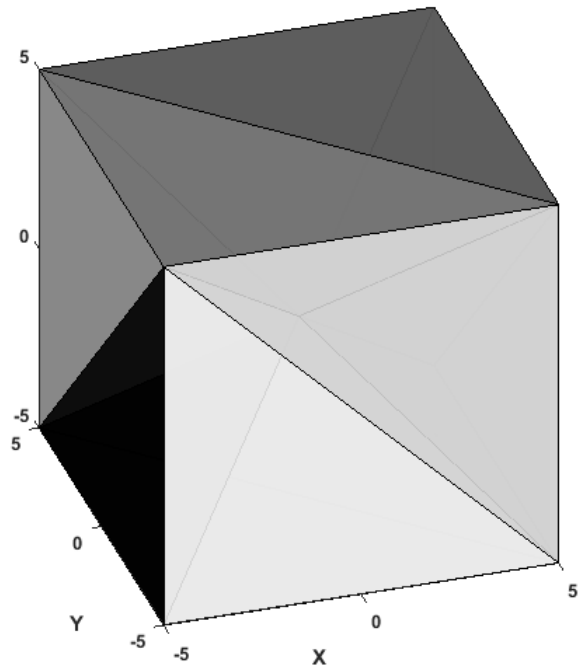


FIGURE 6.8. Geometry creation using a tetrahedral mesh created in MATLAB.

```
>>. d = [-5 5];
        [x,y,z] = meshgrid(d,d,d); % A cube
        x = [x(:);0];
        Y = [Y(:);0];
        z = [z(:);0]; % [x,y,z] are corners of a cube
                                                plus the center.
        dt = delaunayTriangulation(x,y,z);
        Tes = dt(:,:);
        X = [x(:) y(:) z(:)];
        tetramesh(Tes,X);camorbit(20,0)
```

FIGURE 6.9. Geometry creation commands using a tetrahedral mesh created in MATLAB.

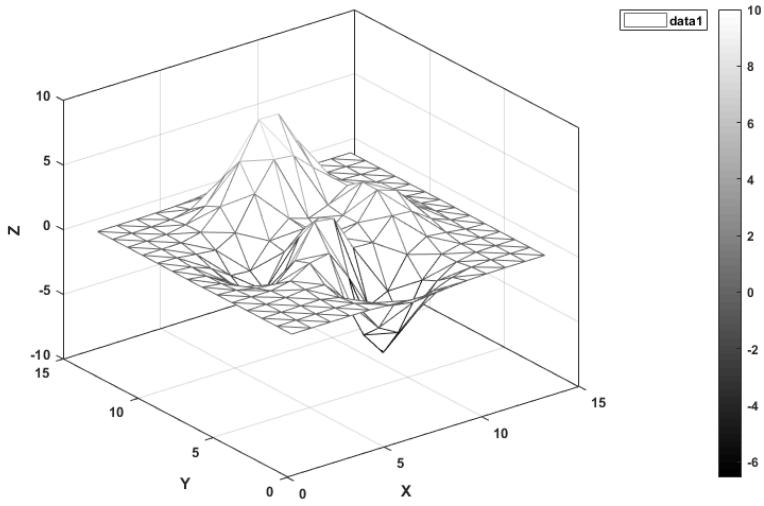


FIGURE 6.10. Geometry creation using a wired triangular mesh (wired using *trimesh*) created in MATLAB.

```
>> [x,y] = meshgrid(1:14,1:14);
     tri = delaunay(x,y);
     z = peaks(14);
     trimesh(tri,x,y,z)
```

FIGURE 6.11. Geometry creation commands using a wired triangular mesh (wired using *trimesh*) created in MATLAB.

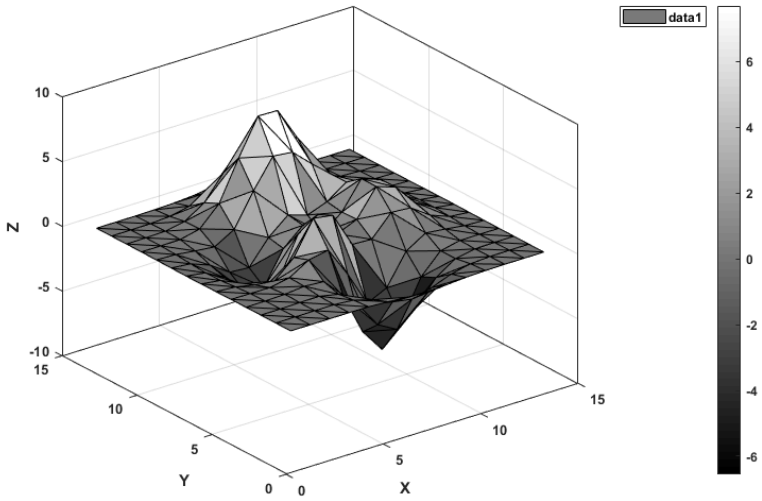


FIGURE 6.12. Geometry creation using a surface triangular mesh (wired using *surfmesh*) created in MATLAB.

```
>> [x,y] = meshgrid(1:14,1:14);
   tri = delaunay(x,y);
   z = peaks(14);
   trisurf(tri,x,y,z)
```

FIGURE 6.13. Geometry creation commands using a surface triangular mesh (wired using *surfmesh*) created in MATLAB.

A mapping toolbox is an optional MATLAB feature that includes functions to generate just about any possible type of map, including projections such as Lambert or Mercator Conformal charts. The polygons representing country boundaries can be read from a library assembled from satellite data. For example, to create the United States map, the *usamap* function may be employed. Figure 6.14 is an example of the map representing the United States from 30- to 50-degree latitude and –125- to –85-degree longitude. State names are part of the database included with the library; these are presented by indices. The state names are generated as the boundaries are indexed within the code (Figure 6.15). This map is a Lambert Conformal Projection Chart (*geoshow*) that gets access to the boundary data that are recorded as grids inside the program library and are fit into the desired projection type. Note that the latitude and longitude limits should be identified properly for the map to be generated when depicting maps in certain regions. Figure 6.16 is an example of the world map using the Mercator Conformal Projection Chart. The latitudes and longitudes of the specific locations (e.g., London in the United Kingdom and Kingston in Canada) can be also defined within the code (Figure 6.17). Lambert

Conformal Projection Chart (Figure 6.15) and Mercator Conformal Projection Chart (Figure 6.17) are the codes used to generate the associated figures.

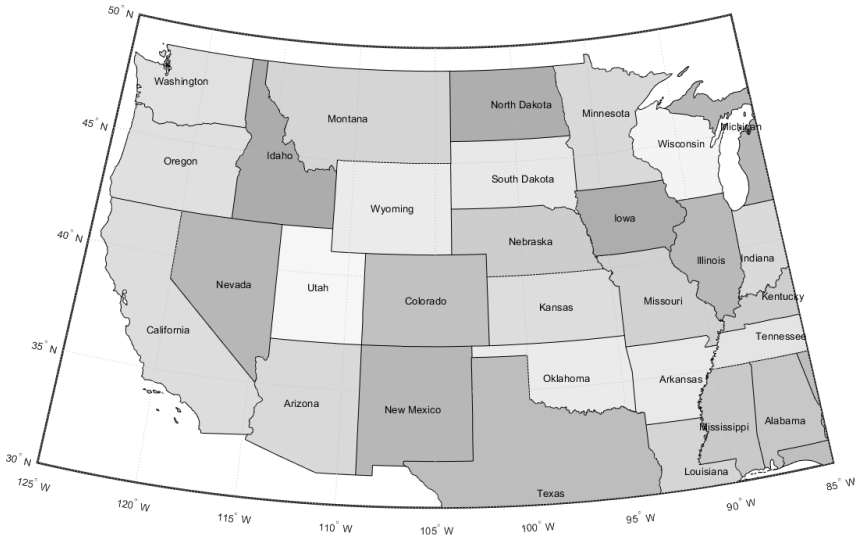


FIGURE 6.14. The United Sates Lambert Projection Chart created in MATLAB Map Toolbox.

```
>> latlim = [30 50];
   lonlim = [-125 -85];
   figure
   ax = usamap(latlim, lonlim)
   axis off
   getm(gca,'MapProjection')
   states = shaperead('usastatehi',...
   'UseGeoCoords',true,'BoundingBox',[lonlimi',latlim']);
   faceColors = makesymbolspec('Polygon',...
   {'INDEX',[1 numel(states)],'FaceColor',polcmap
                                          (numel(states))});
   geoshow(ax,states,'SymbolSpec',faceColors)

   for k = 1:numel(states)
   labelPointlsWithinLimits = ...
   latlim(1) < states(k).LabelLat &&...
   latlim(2) > states(k).LabelLat &&...
   lonlim(1) < states(k).LabelLon &&...
   lonlim(2) > states(k).LabelLon;
   if labelPointlsWithinLimits
   textm(states(k).LabelLat,...
   states(k).LabelLon, states(k).Name,...
   'HorizontalAlignment','center')
   end
   end
```

FIGURE 6.15. MATLAB Map Toolbox script used to generate the United Sates Lambert Projection Chart.

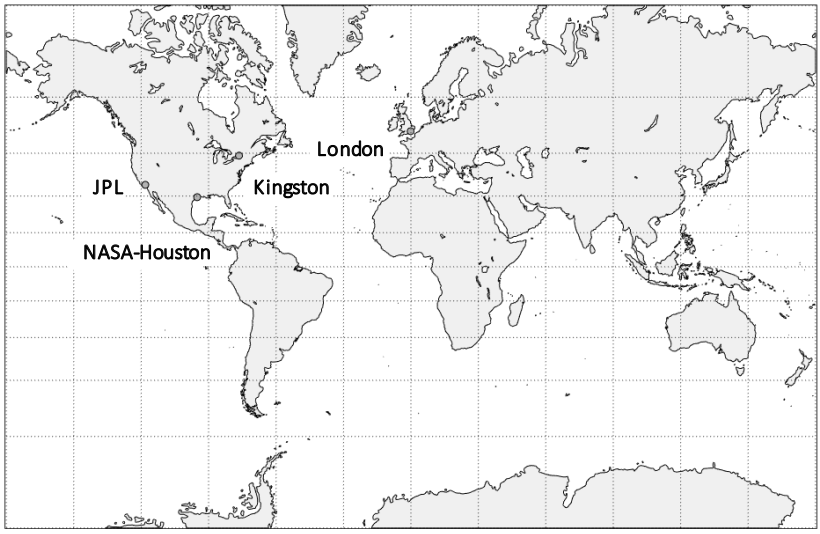


FIGURE 6.16. The world Mercator Projection Chart created in MATLAB Map Toolbox.

```
>> latlondon = 51.50642; lonlondon = -0.12721; % London
latparis = 48.87084; lonparis = 2.41306; % Paris
latkingston = 44.23154; lonkingston = -76.47933; % Kingston
latjpl = 34.201694444; lonjpl = -118.171666667; % JPL
latnasa = 29.55184; lonnasa =-95.09826; % NASA - Huston

% The first field by convention is Geometry
                                                (dimensionality).
% As Geometry is the same for all elements, assign
                                                   it with deal:
[Cities(1:3).Geometry] = deal('Point');
% Add the latitudes and longitudes to the geostruct:
Cities(1).Lat = latlondon; Cities(1).Lon = lonlondon;
Cities(2).Lat = latparis; Cities(2).Lon = lonparis;
Cities(3).Lat = latkingston; Cities(3).Lon = lonkingston;
Cities(4).Lat = latjpl; Cities(4).Lon = lonjpl;
Cities(5).Lat = latnasa; Cities(5).Lon = lonnasa;

% Add city names as City fields. You can name optional fields
% anything you like other than Geometry, Lat, Lon, X, or Y.
Cities(1).Name = 'London';
Cities(2).Name = 'Paris';
Cities(3).Name = 'Kingston';
Cities(4).Name = 'JPL';
Cities(5).Name = 'NASA - Huston';

% Inspect your completed geostruct and its first member
Cities
axesm('mercator','grid','on','MapLatLimit',[-75 75]);
                                                      tightmap;
% Map the geostruct with the continent outlines
geoshow('landareas.shp')
```

```
% Map the City locations with filled circular markers
geoshow(Cities,'Marker','o',...
'MarkerFaceColor','c','MarkerEdgeColor','k');
% Display the city names using data in the geostruct
                                                    field Name.
% Note that you must treat the Name field as a cell array.
textm([Cities(:).Lat],[Cities(:).Lon],...
{Cities(:).Name},'FontWeight','bold');
```

FIGURE 6.17. MATLAB Map Toolbox script used to generate the world Mercator Projection Chart.

There are export options available allowing the user to change the color scale (gray versus color) along with formatting options such as font, size, and color. These features are shared among all MATLAB generated diagrams. The user may also import vectors and raster data from the Web map servers, and the files may be exported in formats such as Shapefile binary files (*shapewrite*), Keyhole Markup Language (KML) text files—*kmlwrite*—and *GeoTIFF*, which write the geodata as vector coordinates and map attributes to the desired file and eventually the Web. *. *geotiff* files are similar to *.*tiff* image files with additional attributes associated with parameters for georeferencing and projected coordinate systems. *.*kml* files are a form of *.*xml* files that store the hyperlink information and map components' relations. *.*shp* files work with vector geodata and tabular attributes. The format of the files importable into MATLAB is shown in Figure 6.1.

6.2 AUTODESK INVENTOR

Autodesk Inventor is a 3D CAD tool with capabilities similar to the other tools described in this section: design, modeling, and visualization. It was released in 1999 and runs on the Microsoft Windows® operating system. It is possible to process the 2D and 3D geometries in one environment and interact with the geometry in a live fashion; this feature is available in the majority of modern CAD tools. Parametric capabilities are also available within this tool with the capability to apply the changes to the geometry directly and instantaneously see the results. Autodesk ShapeManager is the 3D geometric kernel employed in this software. It was developed from ACIS 7.0 in 2002. Metadata (in addition to the drawing data) can be stored in the *.*dwg* files. Metadata is employed by a number of CAD tools including BricsCAD, Caddie, DraftSight, IntelliCAD, and Open Design Alliance. This tool has a multi-CAD translation capability that can import files in a wide range of formats [162].

6.3 AUTODESK REVIT®

Autodesk Revit (Revise-Instantly) is a modeling software used by building and landscape architects as well as structural engineers and contractors that was originated in 1997 by Charles River Software, which later became Revit Technology Corporation in 2000 when it was acquired by Autodesk. Design features such as 3D solid modeling and 2D drafting are available in this geometry tool as well as annotation capabilities. The building modeling and material library databases are among the built-in features. The program is able to link the coordinate information (3D) with time- and schedule-related data and therefore is described as a 4D Building Information Modeling (4D BIM) package. This feature makes it possible for the project contributors from the designer to the contractor to have a high-level perspective on the entire project lifecycle and its milestones; therefore, it functions as a project management tool suitable for complex projects.

The required operating system for this program is 64-bit Microsoft Windows. The main feature of this CAD tool is its capability to provide a parametric 3D model that can accommodate geometrical and non-geometrical attributes—building information model. There is a bidirectional relation between the user and the program, meaning that any changes made to the 3D or draft geometry are automatically translated into the affected features and annotations without direct action by the user, making the tool a *family editor*. For example, if the user decides to move or remove a wall or window, the entire design is revised to incorporate the changes made after the move or removal of the entities. In other words, all the related data are updated instantly [163].

6.4 PTC CREO

PTC Creo employs a parametric feature-based approach similar to the one discussed in Section 6.6 ("SolidWorks") to create geometrical models and was originally developed by Bricsys NV in 2002. This product is well-known among the community of discrete-product manufacturers where the final product is an identifiable object such as aircraft, furniture, or any other mechanical piece—versus process manufacturing such as an oil or sheet metal rolls. The capabilities of this tool allow for design and revision of individual items or unit production with any level of complexity. Bricsys is one of the initiators of the Open Design Alliance, merged with BuildingSMART International in 2016, and was acquired by Hexagon AB in 2018. This CAD tool can be operated within Linux and macOS in addition to Microsoft Windows operating systems

as platforms. When importing files into the software, they are categorized into their building blocks, making it possible for geometry components to be classified for ease of processing. Materials libraries are also introduced into the software. The program provides a convenient GUI, but there is also access to command-line interface [164].

6.5 CATIA

Computer-Aided 3D Interactive Application (CATIA) is a CAD tool that takes advantage of a multi-platform capability (CAD, CAM, and CAE). Its development started in 1977 as an in-house project of a French aircraft manufacturer Dassault Aviation SA as part of their Mirage fighter jet program. In 1981, Dassault Aviation created a subsidiary (Dassault Systèmes) for this software development and distributed it via IBM. In 1992, CADAM, the software developed by Lockheed that was distributed worldwide through IBM, was purchased by Dassault Systèmes and merged with CATIA for its V4 release. While originally developed for IBM mainframes, CATIA now works on Microsoft Windows and Unix operating systems.

Boeing is the largest customer of CATIA, adopting the tool in 1984—CATIA V3 and V5 were employed in the creation of the Boeing 777 and 787, respectively. A large inventory of component designs originally created in CATIA V4 in some aerospace companies were no longer compatible with the V5 version; as a result, the recreation and revision of these geometries caused major delays in the delivery of the products and a loss of billions of dollars (reportedly $6.1 billion)—the Airbus A380 was one of the projects that was adversely affected. CATIA V6R2011x was released in 2010 and development continued until 2012, where V6 2013x was presented. 3DEXPERIENCE Platform R2014x was released in 2014. In addition to the aerospace companies, architecture, automotive, consumer packaged goods, energy, high-tech, industrial equipment, and shipbuilding industries also take advantage of CATIA capabilities [165,166].

CATIA is an advanced, high-level CAD tool. It can be characterized as a product lifecycle management CAD tool that supports all product lifecycle stages, from conception to design, engineering, manufacturing, and documentation. It also facilitates collaboration among the applicable disciplines. Features such as surfacing are available. It is also possible to design electrical, electronics, fluid, and HVAC systems. In mechanical engineering, sheet metal, composite-based designs, and forged tools, to name a few, can be created and revised within this environment. All the stages including Body in White (BIW)—a process in

which cars' body components are joined together using any of the techniques such as spot welding, riveting, laser brazing, clinching—before painting and chassis sub-assemblies—as well as trim installations are also addressed. Parts tolerances, tooling, die, molds, and generative sheet metal and surface design in aerospace industries are also available.

The original design may commence from scratch or 2D sketches, and therefore creation, revision, and validation is possible. The cross-discipline integration in modeling is available, which is helpful when addressing systems engineering problems. Smart products and systems can be addressed using Model Based Systems Engineering (MBSE) and can incorporate the requirements of engineering, configuration management and traceability, system architecture, and system-level simulation. In the field of electrical engineering, the designer can prepare schematic diagrams and 3D paths for wire harnesses and cables. In the field of HVAC design, it is possible to design 3D interactive routing to accommodate pipes of desired geometry (e.g., rounded) given the space available. It is also possible to create 2D diagrams to define hydraulics and instrumentation as well as isometric drawings [167].

6.6 SOLIDWORKS

SolidWorks is a mid-level 3D CAD solid modeling tool currently distributed by Dassault Systèmes; it was developed in 1995 by SolidWorks Corporation and was acquired by Dassault Systèmes, the developer of CATIA CAD tool, in 1997. It is based on a Parasolid geometry kernel. This CAD tool takes advantage of a parametric feature-based methodology in which parameters drive the geometry configuration and relationship between the components and subcomponents of the assembly such as distances, positions and orientations, material properties, and tolerances. This characteristic makes it possible for the assembly database to be created and organized, which is linked to the main model and is revised as changes are made to the geometry. This database is also known as the Bill of Materials (BOM). Assuming that the user has identified certain characteristics, such as color and material type, associated with certain parts of the geometry, they may obtain physical characteristics of the selected features, provided that basic properties such as density and strength are inputted into the software. Similar to other comparable software, this CAD tool can calculate the area and volume of the features by inspecting the selected subcomponents. It can then calculate the total mass of the chosen parts; for example, using the known physical and mechanical

properties. This is an important capability; assuming the ultimate goal of creating a 3D model is to manufacture it, the BOM makes it possible to estimate the type and quantity of the material that may be required based on the simple analysis performed inside the CAD program or a more advanced analysis tool such as Abaqus FEA™. The cost analysis then may be performed. Sometimes, addition or subtraction of material thickness by one millimeter may result in thousands of dollars of savings in fixed costs. If the material type changes, it is possible to estimate the associated cost changes.

The parametric-based features may be used in conjunction with the CSG to represent complex systems. The parameters are generally the input variables to a geometry model, varying from the numeric data such as size and angles to the relationships such as parallelism, equality, and verticality. Another concept that is discussed in SolidWorks is the design intent, meaning that feature's application is considered when setting up the relationships or introducing features. For example, an aircraft door is to be placed on the right side as identified by the designer. This intention is to be communicated with the CAD tool. The CAD tool honors this request and does not interfere with the decision, ensuring that the door is placed on the right side of the plane no matter what the shape of the feature is.

The main thing to remember in SolidWorks is that the parameters such as variables, dimensions, and relationships drive the components' shapes and configurations. Similar to Solid Edge, SolidWorks employs 2D (and 3D) sketches to create the building blocks for the geometry, eventually leading to the creation of parts and assemblies. Most of the features are similar between these two mid-level CAD software packages. Some built-in mating features including gear and cam follower are incorporated into SolidWorks, which make it possible for the assemblies to precisely represent the rotational movement of gear trains. It is also possible to create 2D drawings from the parts and assemblies, as is the case for Solid Edge.

SolidWorks files (*-sld*) were originally categorized into drawing (*-drw*), part (*-prt*), and assembly (*-asm*) files with the capability of being imported to different programs for review and revision. The dash-sign (-) is a placeholder stating that any of the said features are to be added to the *sld* prefix as the file extension (e.g., part model extension becomes **.sldprt*). SolidWorks can interact with 3D rendering engines (e.g., Photoworks), content creation tools for technical presentations from illustrations to catalogs, training materials, and manuals (e.g., 3DVIA Composer), and vendor custom-made part libraries to download parts (e.g., 3DEXPERIENCE Marketplace).

6.7 SIEMENS NX

NX is a high-level enterprise-scale CAD/CAE package currently distributed by Siemens PLM Software. Most likely its biggest customer is General Motors. Among the forerunners of today's NX is Unigraphics (UG), sold under this name from 1975 by United Computing. Acquired from the latter in 1976 by McDonnell Douglas, it was sold to EDS in 1991. EDS was part of GM, and hence the new CAD acquisition was deployed throughout this giant corporation. After acquiring SDRC I-DEAS, the features of both packages were combined into the *Next Generation* or NX software in 2002. Siemens PLM purchased this advanced CAD tool in 2007 and introduced Synchronous Technology (ST) with NX 5 release the same year. Note that ST is also part of Solid Edge software, the mid-level CAD offering by the same company.

NX can be used both to design the solid parts but also to perform engineering analysis (e.g., static and thermal cases are modeled by FEM while fluid models are analyzed by Finite Volume Method, FVM). There are also extensive capabilities to assist with manufacturing that include machining modules. This software has a variety of capabilities such as parametric solid and free surface modeling, reverse engineering, styling and CAD design, engineering drawing, reporting and analytics, sheet metal, kinematics, and numerical control. Said features fall into the three main categories known as CAD, CAE, and CAM, used for design, simulation, and manufacturing purposes. In addition to Microsoft Windows operating systems, this CAD tool also runs on Linux and macOS. Note that the latter two OSes are not supported by Solid Edge. NX uses Parasolid for the geometric kernel and D-Cubed as the (associative) engine for its sketcher and constraints as well as JT for its visualization format for specific data.

Citing as their reason the consistencies that this tool provides among its versions, in a very rare move for a large automotive manufacturer, Daimler AG-Mercedes decided to switch from CATIA by Dassault Systèmes to NX by Siemens PLM in 2010. The 3D experience concept presented by Siemens PLM is one of the selling features, which the company claimed justified this move. An advantage of using the newly adopted tool is that a multi-CAD data model database has been created (SMARGAD) that is capable of interfacing with CATIA products to address the formatting issues [168,169].

6.8 SOLID EDGE

Solid Edge is a mid-level 3D CAD solid modeling tool currently owned by Siemens PLM Software. It is built upon an object-oriented

C++ architecture that makes it possible for the applications to be made so that they are capable of incorporating hyper modeling. It is based on a Parasolid geometry kernel and was originally developed in 1995 by Intergraph. Solid Edge only runs under Microsoft Windows. In addition to the common characteristics that are shared among all the major CAD packages, Solid Edge is capable of incorporating ordered, direct, synchronous, and assembly methods of geometry creation.

The ordered method is very similar to the bottom-top approach, where the basic building blocks of the geometry such as lines and curves are joined to create more complex sketches, which can then be extruded to make features that are combined to create parts, and building blocks for assemblies. The geometrical features can be defined by means of predefined specialized locations such as midpoints, endpoints, or central connectors, which are identified in the model dynamically by display of appropriate graphics.

Assemblies are created in the dedicated Assembly module by importing into its individual components and defining their positions by means of constraints. Reportedly, Solid Edge can make an assembly for as many as 1,000,000 parts. A relationship handle feature is available that may be turned on or off so that the user can identify the nature of the relationship (e.g., parallelism or equality). The base feature is the first step taken to create the model; subsequent features are built upon it. When editing any of the features, no matter at what stage of the design, the process of creation rolls back to the point of revision, meaning that the steps taken before that point remain intact while the subsequent steps are either revised accordingly, producing the anticipated outcomes, revert to an undesirable state, or state of obsoleteness. This is also called operating the geometry based on historical data.

On the other hand, direct modeling may include parametric variables, which may be revised as desired without directly interfacing with the created model. This synchronous method of interaction with the geometry is also known as Object Action Interface (OAI) and directly manipulates the object by varying defined variables as well as the parameters in order to facilitate geometry revision. It is also possible to operate the geometries without having the historical data, which is particularly useful for situations where a third-party geometry is imported into the software to be viewed or modified. This is called the direct method of interacting with the model. After importing, a model may then be worked on in either the ordered or synchronous modes; in case of the former approach, the new historical data is added at the bottom of the model's historical tree.

A 2D model made up of lines, curves, circles, and other 2D geometry elements may be created within this CAD tool as a drawing in the *Draft*

module. This 2D draft model then may be copied, as a whole or in part, to the 2D sketch environment for creating part model features. This method is particularly useful when there are multiple sketches, each having their own plane. On the opposite order, it is possible to explode an assembly to its components on any desired parallel plane (e.g., mid-plane or planes at any other locations) in order to view the details of the geometry at the desired cross section. It is possible to define a handle for the exploded plane so that the user can move it along the selected coordinate and decide upon the specific view that facilitates the way the geometry is presented. It is also possible to present a side, top, and front view of the assembly in the 2D draft environment. Solid Edge can interact with collaborative (e.g., SharePoint, mainly used for document management and data storage) and Product Lifecycle Management (PLM) tools (e.g., Teamcenter). It is possible to export the images to the file formats shown in Figure 6.18 and Figure 6.19.

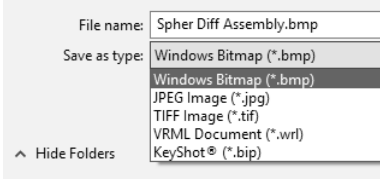


FIGURE 6.18. Image formats that may be created in Solid Edge.

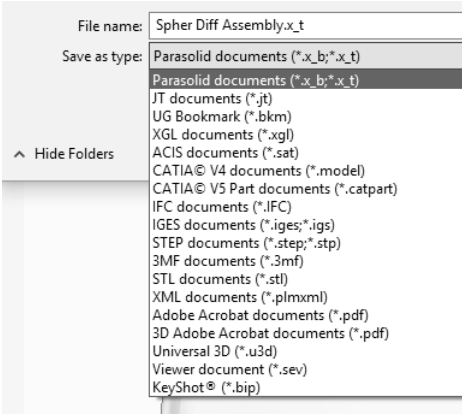


FIGURE 6.19. File export formats (for non-Solid Edge document).

CHAPTER 7

IMPORTING GEOMETRY INTO FEM SOFTWARE

There exist at this time over forty-five FEM commercial tools capable of performing different types of analyses, and they all require a geometry on which to perform the analysis. This geometry can be of any dimension (from 0D to 3D) and can be analyzed in transient (time-dependent) or steady-state (time-independent) modes. Before the FEM tool performs its analysis, it must subdivide the geometry into (small) elements and, depending on the dominant physics, apply different conservation laws (e.g., mass, momentum, and energy). As discussed earlier, before being released, most commercial tools would have normally passed the verification stage in which they were tested for the correct application of physics, continuity at the nodes, and mathematical errors. Most modern FEM software is made to be user-friendly by employing GUI interfaces that simplify the user-software interaction. Nevertheless, if the user is unaware of the applicable physics, FEM analysis, coding, structure of the commercial tool, and the background knowledge, they may not be able to produce meaningful analysis even though the generated colorful contour plots of the post-processed results may look beautiful.

7.1 GEOMETRY IMPORTING

One of the key features of any commercial FEM tool is its method of interaction with the imported or created geometry, meshing it, and the method of handling the mesh with the analysis kernel. The part geometry may be meshed by triangular and rectangular elements (for a 2D analysis) or prismatic, tetrahedral, pyramidal, and hexahedral elements

(for a 3D analysis). The number of nodes associated with each element type and the number of elements determine the number of nodes that are generated and therefore the degree of complexity of the analysis. With the increase in the number of nodes, the problem grows in complexity due to the greater number of equations to be solved, cross-checking to be made for the continuity, and increased solution time to complete the analysis by satisfying the convergence criteria.

In order to employ the available resources most efficiently, the user needs to decide on the optimum number of nodes or elements. There are programs that may assist with this task. For example, Engineering Equation Solver (EES) can be employed to input the physics in general terms as well as discretized physics (e.g., finite difference form of the equations). The program allows for the cross-referencing of the two methods of the physics entry in order for the comparisons to be made, and as a result optimum time step and element size can be determined. One of the considerations when deciding on the appropriate number of elements is numerical errors. Error generation is one of the inherent characteristics of any numerical analysis and is dependent upon the set tolerances and convergence criteria.

Generally, the errors fall into two categories: (1) truncation errors—elimination of the portions of the numeric values calculated after a certain number of decimals; and (2) round-off errors—rounding off the numeric values to the closest value after a certain number of decimals. The mesh, consisting of nodes and elements, generated from said techniques is used in a majority of the scenarios to solve physical models. Every additional node creates more degrees of freedom and thus independent variables for the numerical model. Iteration techniques are then employed so that the system of equations based on the model nodes is assembled and solved. The initial state of the node (temporal and spatial data) should converge to a value that represents the physics of the problem. With an increased number of nodes, the number of equations also increases. Although by increasing the number of equations, a more accurate representation of the physical system is generally achieved, at some point numerical errors will start affecting the accuracy of the results.

While it is possible to mesh the part as one unit, it is also possible to subdivide the part into district regions and mesh each region in a different fashion (methods, element types, and sizes). Obviously, in majority of scenarios, the parts are being analyzed in the same dimension. For instance, it is possible that the thickness of a part is so small that the third dimension may be ignored; however, when analyzing this solid model, the thickness in the form of a shell, or whatever available feature, is introduced for physics consistency and inclusion of the part within the solid model for compatibility purposes. Depending on the geometry

shape, consisting of curves, straight lines, or corners, the element types are selected and even on occasions boundary and initial conditions are set to avoid singularity, errors, or excess analysis time that may be otherwise avoided if the element selection and application are done skillfully. The magnitude and complexity of the geometry that the program is to handle (either in the form of geometry import or creation) becomes important when making said decisions. The methodical approach in most of the scenarios is to perform a sensitivity analysis for both element type and size. The user usually starts from smaller (and potentially simpler) element numbers (and types)—lower versus higher order elements having nodes at their centers, for example—and progresses to greater numbers and higher order elements.

The analysis results at each step are compared with those of the previous step(s). They should eventually converge to similar values (e.g., similar temperature or stress at a particular location). This means that the difference of the values (from the previous to current step) as the iteration progresses approaches zero. Note that the differences between the analysis results are not absolute but are compared *relative* to a given base condition. The base condition in this case could be the results of the previous analysis step(s). For example, a 10 K temperature difference calculated from two consecutive iterations for a 1000 K scale equals 1 percent error, while for 100 K and 10 K scales, the errors would be 10 and 100 percent, respectively.

Some FEM tools have an adaptive meshing capability. When this feature is activated, after each solution iteration is performed, the element sizes are adjusted for the optimum number of elements and nodes at the coordinates where most attention is required, and the solution continues to be iteratively repeated with the new adjusted mesh. Increased element density is commonly beneficial around sharp corners, any high-stress areas, high curvatures, and near wall boundaries where fluids interface with solids—to name a few. In other words, the mesh adapts to the conditions. Activation of adaptive meshing may lead to increased solution time, however. For example, from the Author's previous experience, there have been cases that while performing a stress analysis for a turbine engine disk using nonadaptive meshing took under one hour, activating the adaptive meshing feature resulted in an analysis time increase of over fifteen minutes.

Some FEM tools have mesh-checking features such as collision testing. For example, they are able to detect double nodes or elements, degenerated elements, and elements not respecting the optimum aspect ratio (deviating too far from 1). It is occasionally possible to export or import one set of mesh from a FEM environment to another environment. Some tools are capable of parallel solution computing, making

more effective use of today's multicore processors. One approach that allows parallel computation is to divide the geometry into subsections and solve each subsection simultaneously. By shortening the computation time, parallel computing allows addressing of more complex problems. In some of these cases automatic mesh partitioning may be available. However, the saved time is not usually linearly related to the number of added cores; for example, the user may not conclude that a four-core processor is twice as fast as a dual-core. With an increasing number of cores, the solution time may reach a plateau, making any further increase in the processor core count not justified given the extra cost. So, while one may dream of a 5 GHz 28-core monster desktop processor, it may not do any better than your "basic" 12-core machine [170].

7.2 GEOMETRY-FEM COMPATIBILITY

Another important feature of a commercial FEM tool is the method by which it interacts with the imported geometry from a third-party CAD commercial software package. As briefly mentioned in the earlier sections, the designer who creates the geometry in the CAD tool may "perceive" that they did a good job when creating the solid part—meaning that their design standards are *met*. However, the same file imported to the FEM tool may not meet the standards of the hosting environment. There are errors in creating the geometry in the CAD tool that may go unnoticed even by the most expert designers. They may focus on the correct application of the dimensions, relative positions of the components with respect to each other, or subpart interaction within an assembly. However, they may not necessarily ensure that the line and area connections at the vertices are made perfectly and at the correct locations.

The FEM specialist, however, is aware of such error types and can detect them if they have come across a number of troubled geometries in their geometry manipulation dossier. As geometries get more complex, detecting and providing solutions to address the geometry discrepancies become assets to the project. On occasions, the FEM specialist may decide to simplify the geometry to the bare necessities or create a new one—for example, the negative of the part when modeling air flow within a channel—to make the geometry suitable to the FEM tool and also to expedite the analysis process. They may have to think outside the box (or the square) depending on the analysis dimensions. They can then experience the pure joy of seeing the predictions made based on their virtual model validated by a physical mock-up.

Since in a majority of scenarios the commercial FEM software developers cannot compete with the dedicated commercial CAD tool

developers in either geometry creation or manipulation, they either partner with some—making it possible for their users to employ the geometry kernel directly in the program—or simply cannot afford this union—meaning they do not have a built-in kernel—and opt for interfacing applications that can be employed as a bridge between the FEM program and chosen CAD tool in order for the geometries to be imported under certain conditions. The concept of geometry-tolerance imports and modifications so that the geometry is accepted and meaningful to the hosting program may be better explained with the following analogy.

Imagine a guest entering the house on a rainy day. The host may ask them to take their shoes off as the host does not tolerate the dirt that may be brought in by the wet shoes—*tolerances need to be defined for the geometry import*. The host also asks the guest to leave their dripping umbrella outside the door—*geometry may need to be defeatured or simplified*. Those are the host conditions that assure that the guest-host interactions are respectful of the limitations and therefore doable. Now what if the guest has a soaking wet pet? The host may provide a towel for the pet to be wrapped around and dried; they may also provide some booties. This is isolating the object surface (i.e., fur of the animal) from the rest of the environment (i.e., furniture) on an as-needed basis.

This capability is importing the entire or portions of the geometry as individual entities—disconnected parts—into the FEM tool. In this scenario, however, the user (host) is to ensure that the surface and node connections are made at proper locations when required. For example, the pet is to drink the water with the towel remaining clear of the bowl. The FEM specialist (host) may decide if the head of the animal may be uncovered after it is dried and can come in direct contact with the pillow (i.e., applying boundary conditions to the specific areas). The rest of the pet is either in contact with the surrounding objects or not. Having said that, the interfacing boundaries should be respected so that the continuity laws of physics are applied correctly when applicable. In other words, the desired conditions or requirements for the CAD-generated geometry, which should be imported into the FEM tool, are set by the host. Presumably, the host has every intention for the imported geometry to perform well within the new environment; however, it cannot be manipulated into operations that it was not meant for—as there are operational limitations with any tool (CAD to FEM).

7.3 MULTISTAGE GEOMETRY ANALYSIS

In some cases, it is possible for the geometry to be created in a CAD tool or imported into a FEM environment (e.g., ANSYS Maxwell),

solved for the desired variables (e.g., electromagnetic and electric fields), and for the analysis results then to be exported to another FEM environment (e.g., COMSOL Multiphysics). This is done since the initial environment is either not capable of multiphysics modeling (e.g., thermal analysis induced from electromagnetic analysis), the in-house ultimate analysis tool (e.g., performing thermal analysis using COMSOL Multiphysics) is different from the initial analysis tool (e.g., performing electrical analysis using ANSYS Maxwell), or simply there is no will or expertise available on-site to perform such analysis. In the said example, the electric field data for the individual nodes can be exported from ANSYS Maxwell (raw data), which can then be formulated to relate to the heat generation data associated with the nodes. This may be achieved by means of a third-party program. For very large data sets, MATLAB (or SciLab, a free and open-source MATLAB alternative) may be employed; for medium-size sets, Microsoft Excel may suffice; at the lowest level of data set complexity, text editors such as WordPad may be used. In this case, the data is to be converted to **.csv* files (numbers as text separated by commas), with each type of data collected under the corresponding column heading.

The native languages of the FEM tools may differ (e.g., Fortran 90, Python, C++, and MATLAB) so that the scientist may interact with the programs in different ways when writing the subroutine codes. OpenMP (Open Multi-Processing)—an Application Programming Interface (API) supporting multiple platforms including a number of compiler directives, routines, and variables—may not be supported by the FEM tool. OpenCL (Open Computing Language)—an interfacing platform that is executed across systems with multiple cores or processors in addition to the coprocessors (processors that carry out complementary tasks to the main processor)—may be available in a number of FEM tools. Additionally, optimization solvers that are used for topology, size, and shape optimization (as discussed earlier) may be available either as add-on modules or stand-alone products capable of interfacing with the main FEM platform.

7.4 GEOMETRY IMPORT INTO COMSOL MULTIPHYSICS

COMSOL Multiphysics is capable of interfacing with multiple geometry platforms (e.g., Solid Edge and SolidWorks), sharing a similar modeling kernel (Parasolid) with them. This FEM commercial tool can run on all popular operating systems (Linux, macOS, and Microsoft Windows) and even in a Web browser. It is capable of importing a variety of geometry file formats such as **.step* and **.iges*. One of the

features of this software is its ability to carry out sweep and auxiliary operations. This capability allows the user to select one or more variables (or parameters) to be varied over a specified range and at specified values, evaluating the solution for each combination of values resulting in multiple solutions stored inside a model as a single set. This feature makes it possible to perform sensitivity analysis on the desired variables (or parameters). Since it is possible to define the geometry dimensions using variables, COMSOL Multiphysics can therefore conduct sensitivity analysis on the geometry domain size or shape. For example, the user would be able to perform multiple analyses on a fin model exposed to the heating boundary condition on one side to decide what length should be chosen so that the analysis time is minimized while keeping the solution errors within an acceptable range [171].

COMSOL Multiphysics can interface with platforms that are either directly related to geometry creation (e.g. Solid Edge) or that can map data to another third-party program (e.g., MATLAB). This means the user can export data in a text format that can then be imported into other tools that cannot link directly to the FEM tool. This may also be done by exporting data to Microsoft Excel using the built-in feature (LiveLink™). This section provides a summary of these interfaces and some of their capabilities. These features are not included in COMSOL Multiphysics base package; they are additional features available for extra cost. Some of these optional interface functionalities allow for *live* communication between COMSOL Multiphysics (the host) and the *guest* program; COMSOL Multiphysics refers to this functionality as LiveLink. Thus, if the guest program is working in the background and the connection to the FEM tool has been established, making changes in the guest program will result in automatic changes to the geometry, a property, or any other parameters in the host program (i.e., FEM tool).

MATLAB LiveLink provides the capability of integrating MATLAB scripting feature into the FEM tool. This makes it possible to define geometrical functions using MATLAB toolboxes, similar to what was presented in Section 6.1 ("MATLAB"). This link is available for the entire modeling lifecycle from pre- to postprocessing. Microsoft Excel has a LiveLink that makes it possible for the spreadsheet data to be transferred to the FEM tool. This data then may be employed when defining variables (or parameters) that are associated with the physics or geometry.

LiveLink is available for a variety of popular CAD tools (Table 7.1). These include three Autodesk packages (AutoCAD, Inventor and Revit), two PTC packages (Creo Parametric® and Pro/ENGINEER®), as well as SolidWorks by Dassault Systèmes and Solid Edge by Siemens PLM. The LiveLink functionality for any of these CAD tools makes it possible for the model geometry to be updated in the FEM tool as the

TABLE 7.1. COMSOL Multiphysics interfacing products functionality.

LiveLink™ (Data)	LiveLink™ (Geometry)	Import Functionality
MATLAB®	SolidWorks®	CAD (general)
Microsoft Excel®	Inventor®	ECAD
	AutoCAD®	CATIA™ V5
	Revit®	
	PTC® Creo® Parametric™	
	PTC® Pro/ENGINEER®	
	Solid Edge®	

geometry is revised within CAD. For example, it is possible to define variables inside Solid Edge. These variables will be automatically created inside COMSOL Multiphysics in the form of tables when the geometry is connected by a LiveLink. The user is then able to revise the data within the FEM tool table and see the revised changes implemented in the CAD tool. Synchronizing the geometry ensures that the geometry gets updated instantaneously [172].

The CAD Import module is another geometry interface that makes it possible for geometries created in any dedicated CAD tool to be imported using standard geometric data formats such as **.step* and **.iges*. Note that only import, not the LiveLink functionality, is available in this module. Also, note that any of the LiveLink geometry interface modules are also capable of importing standard geometric data formats supported by the CAD Import module. The Design module is a dedicated add-on that includes enhanced geometric modeling capabilities as well as importing (similar to CAD Import module) and exporting the created geometry in a variety of formats.

CATIA V5 File Import allows accurate model input into COMSOL Multiphysics for this CAD tool's file formats (**.catpart* and **.catproduct*). It functions as an add-on for the CAD Import module, Design module, and the geometry LiveLink products. Electronic and Electrical Computer Aided Design (ECAD) files may also be imported and converted from 2D to 3D geometries and then be employed in a simulation.

CHAPTER 8

GEOMETRY CREATION IN COMSOL MULTIPHYSICS

This chapter will guide you along the process of creating a geometry model within a FEM tool. The process begins with considerations of how you can reduce the complexity of the model without losing any useful information. Then you will learn how to create a new model file and how models with different dimensions can be set up. Finally, work planes, a very useful tool for geometry creation, are discussed.

8.1 METHODS TO SIMPLIFY MODEL GEOMETRY

Jumping directly into creation of a 3D FEM model is not always the best approach. The simplest model that gives the desired information is the best model. The Keep it Simple, Stupid (KISS) concept should be followed here as well as in other pursuits. Simplifying the models will not only reduce the resources required (computational capability, expertise, and time) but also will show the expertise of the FEM specialist—being able to analyze the model in the most efficient way.

A model can be simplified in terms of the analysis type, solution methods, solution parameters, and applied boundaries and initial conditions. Examples of this include choosing 2D versus 3D models (analysis type), direct versus iterative solutions (solution method), auxiliary and mesh-adaptive features (solution parameters), initial load and heat flux (initial and boundary conditions).

Consider for example a FEM heat transfer model of a human eyeball (developed by the Author for a master's thesis project). In this model the eye is exposed to cyclic heat sources analyzed by a Finite Difference

Method (FDM). This model was made in order to compare the accuracy of FDM to FEM, given the curves and complexity of the eye components. It was found that the FDM models produced similar results to the FEM models while substantially reducing the model's complexity and thus solution times. This shows how one can benefit by selecting a methodology that is both efficient and effective.

Geometry should be simplified to the extent that it will not negatively affect the integrity of the problem's physics. For example, the exterior surfaces of a pipe in a flow analysis where airflow movement is the main focus may be ignored when performing such an analysis. However, the interior baffles for the said example cannot be eliminated, since they directly affect the airflow. Deleting the said features from the analysis and guessing that this elimination affects the airflow positively or negatively is not a scientific approach for modeling in this case. Common sense tells us that the baffles will affect not only the flow direction but also the noise level at locations where vortices are formed.

Another example is the case of performing a heat transfer analysis on a geometry. If the model contains small features like screw holes, chambers, and rounds, they can be removed (a process called defeaturing) without affecting the accuracy of the heat transfer analysis results. Removing these features can significantly reduce the number of mesh elements, improve the mesh quality, and reduce the solution time.

Another important aspect to consider when setting up a model is whether one can take advantage of any existing symmetry. Symmetry can be reflectional (about a line in 2D or a plane in 3D) or rotational (about a point in 2D or an axis in 3D). 2D reflectional symmetry can be about one line or two lines (Figure 8.1). 3D reflectional symmetry can be about one, two, or three planes (Figure 8.2). Thus, for 2D, one could reduce the model size to half (one symmetry line) or a quarter of the original size (two symmetry lines). For 3D, a model can be reduced from half to one eighth, depending on how many planes of symmetry exist. Reduction of model size brings numerous benefits (e.g., reduction of memory required and solution time). In some cases, reduction of model size by taking advantage of symmetry can change a solution from infeasible (due to RAM limitations) to feasible.

Rotational symmetry can bring about similar or even greater benefits. If a 3D model is rotationally symmetrical about an axis (e.g., a cylinder), one can reduce the model dimension from 3D to 2D by choosing 2D axisymmetric model geometry when setting up the model. If a 2D model is rotationally symmetrical about a point, one can reduce the solution dimension by employing the 1D axisymmetric model.

If the model has a pattern that repeats an integer (n) number of times when rotating about its central axis, it is said to have a rotational

symmetry of order n. Therefore, for example, Figure 8.3a shows 2D geometry and Figure 8.4a shows 3D geometry, both with a rotational symmetry of order 8. If the boundary conditions and any loads are also symmetrical, the model geometry can be reduced to 360/n-degree sectors without any loss of accuracy. The examples shown on the right in both figures show 45-degree sectors which can be used as the reduced model.

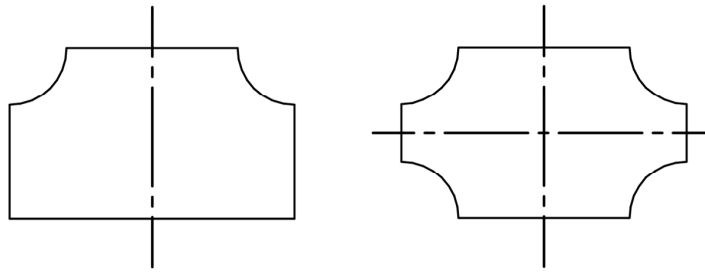


FIGURE 8.1. Examples of 2D reflectional symmetry, about: (a) One line (left), (b) Two lines (right) (drawings created in Solid Edge).

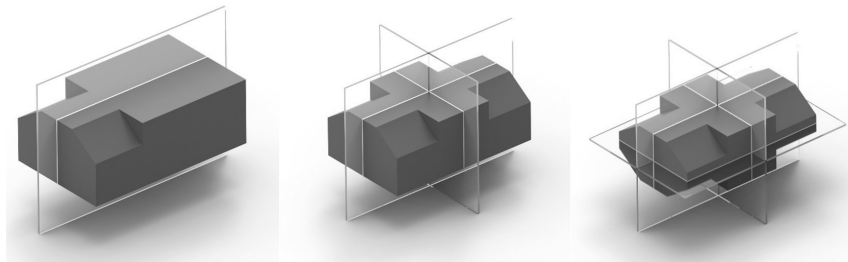


FIGURE 8.2. Examples of 3D reflectional symmetry, about: (a) One (left), (b) Two (middle), (c) Three (right) planes (drawings created in Solid Edge).

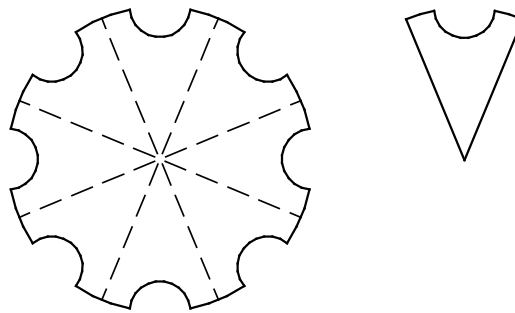


FIGURE 8.3. Examples of 2D rotational symmetry: (a) Complete geometry (left), (b) Sector (right) (drawings created in Solid Edge).

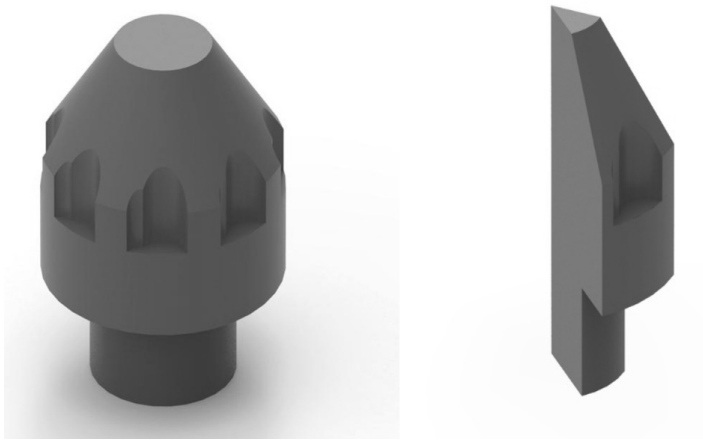


FIGURE 8.4. Examples of 3D rotational symmetry: (a) Complete geometry (left), (b) Sector (right) (drawings created in Solid Edge).

Another way to simplify a geometry is to only include a part of it within your model. For example, if a 20×100 mm^2 plastic bar is being heated by a laser at one end for 10 s, the temperature is going to remain unchanged along most of the bar's length. Thus, there is no need to include the full 100 mm length of the bar in your time-dependent heat transfer model. The challenge here is to determine how little of the bar's length needs to be included in the model to obtain accurate results. This should be done in a methodical fashion by starting with as much of the model as your system can handle and then progressively reducing the model size and noting the point at which the results deviate significantly from the initial solution.

The same methodology used to determine whether the results are affected by the size of the modeled geometry can be applied to any other aspect of the model that you may want to investigate, such as the effect of the mesh size or any other aspect of the solution procedure. This method of validating the solution approach is known as *sensitivity analysis*.

It is prudent to divide the geometry into multiple regions to facilitate the troubleshooting process, especially when importing the solid model from a third-party CAD tool. Imagine a collection (assembly) of multiple countries (sub-assemblies). Each of these countries (sub-assemblies) is made up of cities (parts). The cities are made up of businesses (features), which consist of the people (vertices, lines, and surfaces). Associate a flag to each of the said entities from the assemblies to the vertices. In case a problem occurs during the preprocessing, analyzing, or postprocessing of the imported geometry (or before

geometry is being imported), the entity's flag (presented as a warning message in COMSOL Multiphysics) is highlighted so you know where the problem occurred. Since the assembly is an interconnected entity, you are then to question the associates who allowed for such a problem to happen or the ones that are affected (this is usually provided through messages appended to the geometry node carrying a caution sign in COMSOL Multiphysics). They should be either eliminated or corrected (Figure 8.5). An example is importing a geometry where the vertices do not make proper connection with the lines. This may happen when the entities are overlapping (e.g., multiple vertices at one end), having incorrect relationships (e.g., crossing versus parallelism), or making the wrong connection (e.g., endpoint versus midpoint connection). It should be possible to identify these errors by close examination of the geometry. The form, fit, and function should comply with the work the geometry is intended to perform.

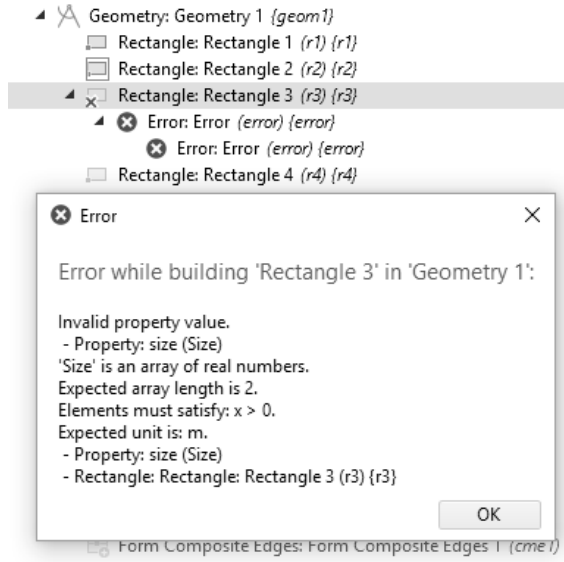


FIGURE 8.5. An example of a geometry error flag and message created in COMSOL Multiphysics.

8.2 SETTING UP A MODEL IN COMSOL MULTIPHYSICS

Before setting up a new model in COMSOL Multiphysics software, it is a good idea to check for the latest updates. The work in this book was carried out on version 5.4. Checking for the latest updates can be easily accomplished from the program interface by going to *File* >

Help > *Check for Product Updates*—located in the main menu ribbon of the program (Figure 8.6).

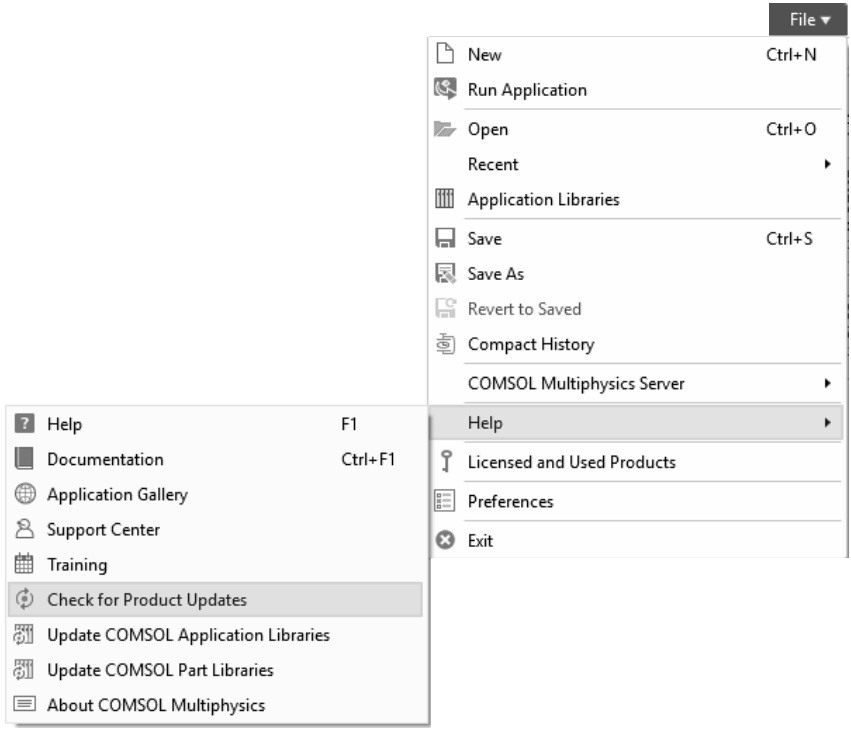


FIGURE 8.6. Checking COMSOL Multiphysics update status.

After selecting *File* > *New*, you are asked to choose between a *Blank Model* or a *Model Wizard* (Figure 8.7a). If you select the *Blank Model*, a new model is created; if you have a previously opened model in the same window, it will be closed (if you answer yes to the pop-up question). The *Blank Model* is usually used to set up specialized models where you input mathematical functions to define the problem physics. For most applications, there will be already an available preset physics which you can select for your model by using the *Model Wizard* (Figure 8.7a). The next step is to select *Space Dimension* (e.g., 2D or 3D) for your new model and whether it is axisymmetric or not (Figure 8.7b). Then the *Model Wizard* asks you to decide what physics to use (e.g., heat transfer, mechanical, the ones included in your base multiphysics package, or a combination of multiple physics). For example, if you wish to set up a model to investigate *Structural Mechanics*, you can select the *Solid Mechanics* physics option under this feature (Figure 8.8).

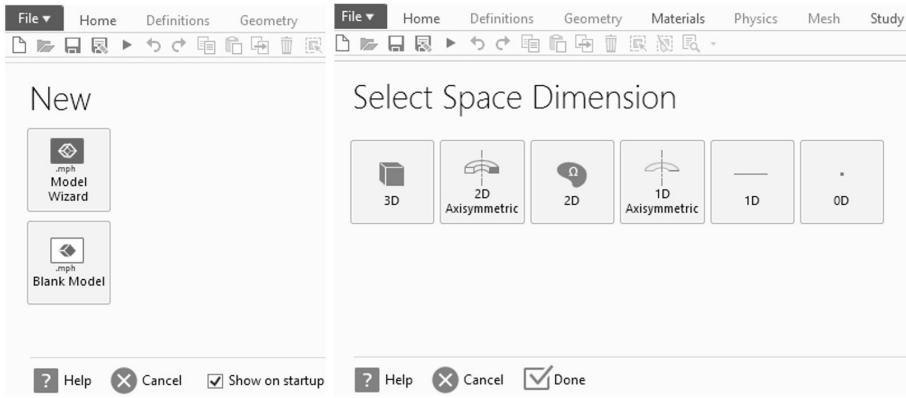


FIGURE 8.7. (a) Setting up a new model (left), (b) Selecting *Space Dimension* (right).

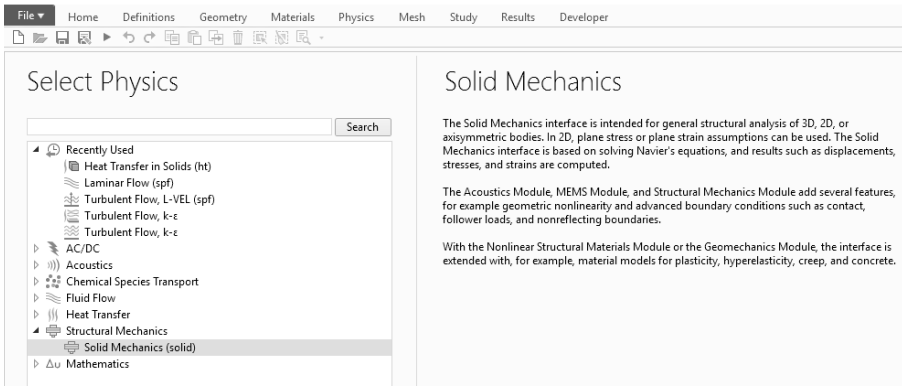


FIGURE 8.8. Selecting physics—*Structural Mechanics* (*Solid Mechanics*).

Note that here you only see one option available to be picked (*Solid Mechanics*); the reason is that the optional dedicated *Structural Mechanics* module was not included in this package. This demonstrates that a user with no access to the specialized modules can still take advantage of the basic capabilities that are available with the base package. In most of the scenarios, it is even possible to revise some input equations and create the desired specialized physics. On the other hand, Figure 8.9 demonstrates the physics selection when a specialized module is available (*Heat Transfer* module in this case). With the added capabilities in this area, you can select among twelve different physics—*Heat Transfer in Solids* is highlighted in this case). When adding the physics, the dependent variable (e.g., temperature—*T*) is also set. It is possible to define your own dependent variable with whatever name and subscript number you choose. After completing this step and picking the Done check mark at the bottom of the window, the *Select Study* window appears where you can select the study type,

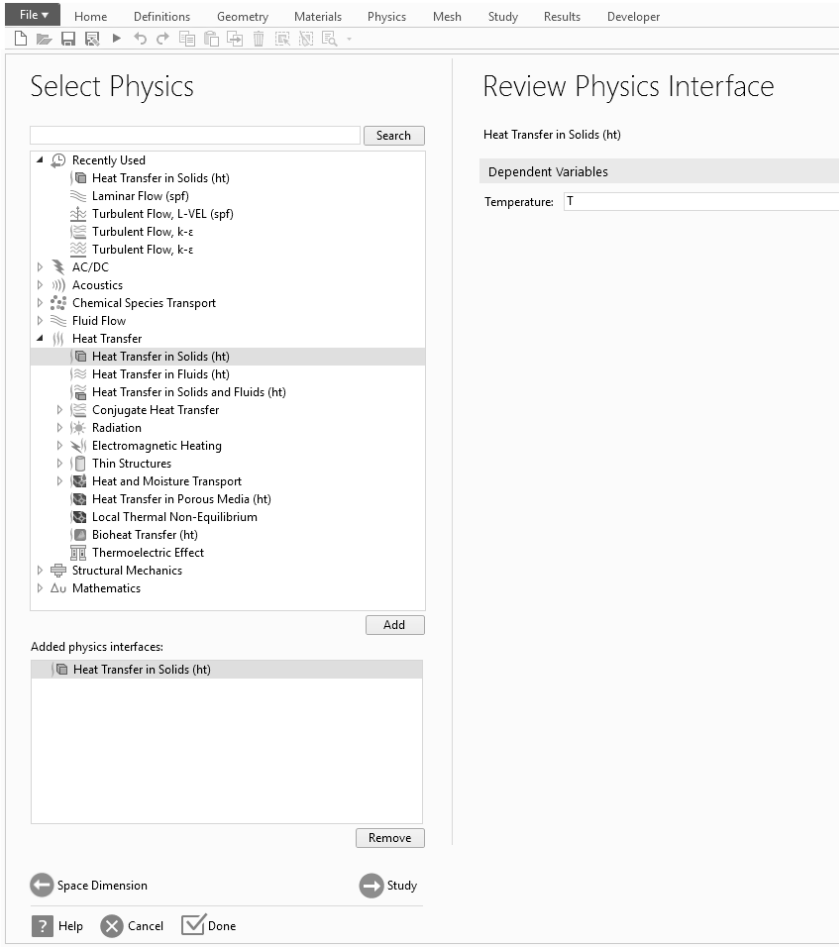


FIGURE 8.9. Selecting physics—*Heat Transfer* (*Heat Transfer in Solids*).

either general (*Stationary* versus *Time Dependent*) or more specialized ones (e.g., *Thermal Perturbation*)—Figure 8.10. In this example, a *Time Dependent* (transient) study is selected. A modeling window will appear after this step.

Welcome to the new home for your model (Figure 8.11)! Let us take a quick tour. There are multiple regions or windows (four in the provided example). You may choose to pin down more regions to the ribbon or reduce the number of regions as desired. On the left side is the *Model Builder* window containing the hierarchical structure of a model tree which defines all the elements of your model: geometry, study type, solution data, and post-processing. Adjoining the *Model Builder* to the right is the Settings window, which will show the settings for the item highlighted in the *Model Builder* window.

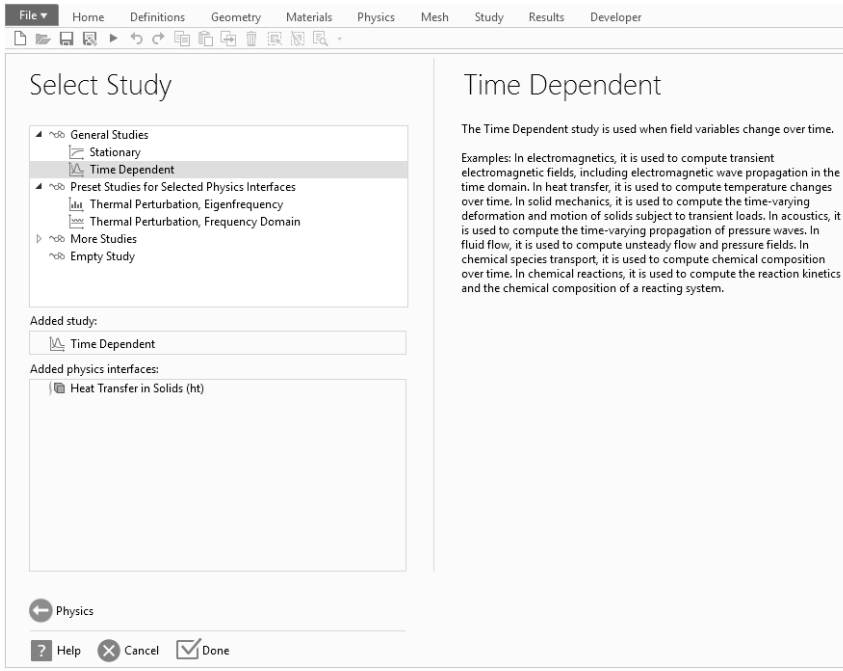


FIGURE 8.10. Selecting study—*General Studies* (*Time Dependent*).

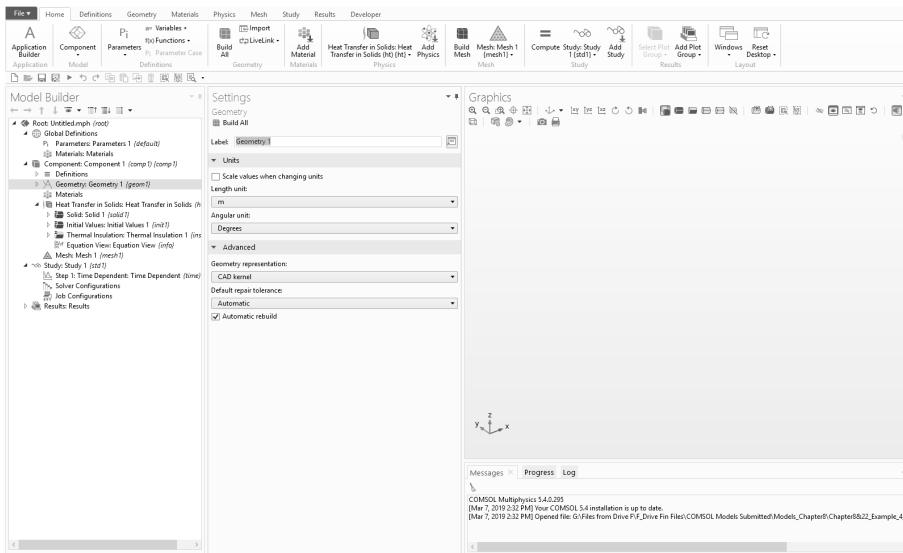


FIGURE 8.11. Modeling and *Geometry Settings* window.

For example, in this case, *Geometry* is highlighted. The Settings window presents the related geometrical characteristics such as the *Length unit* (from nm, to m, up to Gm)—Figure 8.12, *Angular unit* (Figure 8.13), *Geometry representation* kernel (Figure 8.14), and

Default repair tolerance method (Figure 8.15). In this example, two geometrical kernels are available—*CAD* and *COMSOL*; the former is only accessible with the optional CAD Import module while the latter is available in the base package. Selecting *Automatic* or *Relative* identifies the *Repair tolerance* which you feel comfortable for your model to possess (e.g., 10^{-4}). The tolerance setting specifies the threshold below which COMSOL will consider the vertices to be coincident. This may close very small gaps or delete very small edges. If you set this tolerance too high, you may end up with unwanted changes in the geometry; setting it too low may result in geometry errors not being repaired. You may activate the *Automatic* rebuild to automatically update the model as you modify it. *Automatic* selection for *Relative tolerance* may be used to activate the default tolerance recommended for the defined physics.

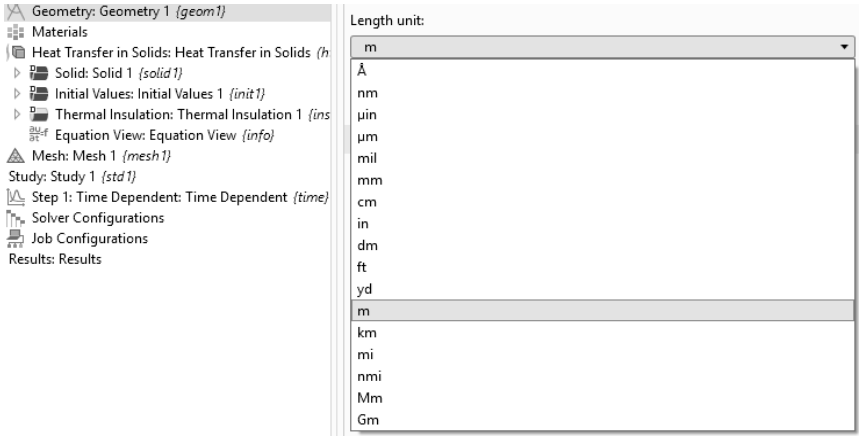


FIGURE 8.12. Geometry *Length unit*.

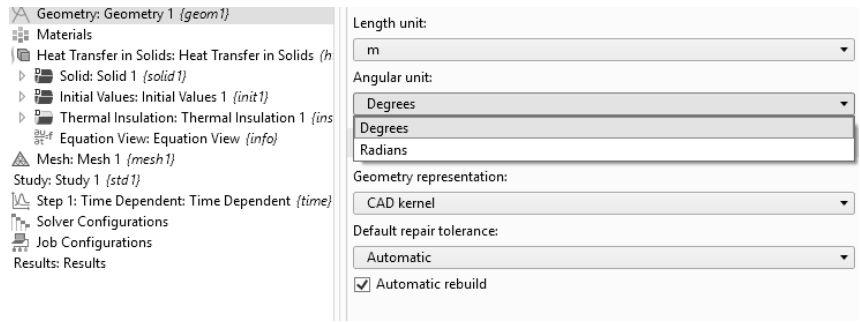


FIGURE 8.13. Geometry *Angular unit*.

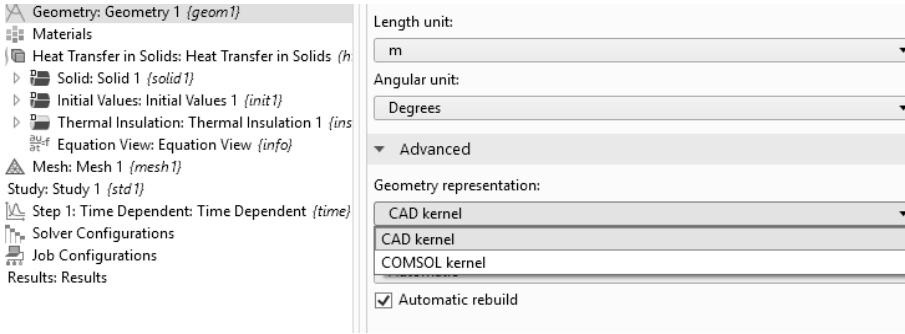


FIGURE 8.14. *Geometry representation* kernel.

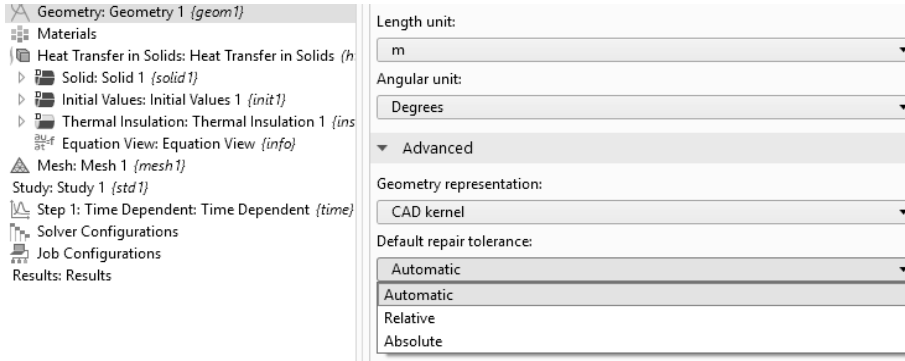


FIGURE 8.15. Geometry *Default repair tolerance* methods.

The next step is to choose the materials for your model, either from the built-in material library or by introducing a blank material and completing the required fields (e.g., mechanical and thermophysical properties depending on the physics), and to associate each geometry region with one of the materials you added to the model. After that the steps are: (1) set up the mesh; (2) mesh the model; (3) set up the study operation controls; (4) solve the model; (5) analyze the solution; (6) present the results in the form of diagrams, contour plots, and animations; and (6) create a summary report for the analysis.

An alternative way to interact with the model is by setting up an application where the user is only given access to certain input variables and can view only certain preselected model elements and solution outputs (e.g., geometry, mesh, diagrams, and contour plots). Creating such an application provides a clear user-friendly interface for those users who do not need to learn the details of the FEM tool operation; it can also be useful for educational purposes. Such an application can run either by accessing a COMSOL Multiphysics Server facility or as a stand-alone executable; the latter is a feature which has been added

recently with an introduction of a COMSOL Multiphysics Compiler. Both *Server* and *Compiler* features are optional add-ons.

Most of the added features such as variables, parameters, and materials may be added either at the local (directly under the subcomponent—tree leaf: local) or global (under the upper-level component—tree trunk: global) levels. In the latter scenario, a link should be created that connects the local property or entity (child) to the global property or entity (parent). Any of the preset parameters or variables, including the material properties and solution control options, may be revised at any time after the initial model setup.

At this stage, although physics and study selections have already been made and the model set up, it is still possible to add a new physics under the current *Component* or a new *Study* under the *Root* (top-level tree). It is also possible to add a new *Component* (of whatever dimension) under the *Root*. The user may add *Study Steps* as well as a variety of *Sweeps* (e.g., *Parametric*, *Function*, and *Material*). *Study Extensions* may also be activated under each of the *Study Steps*. The latter two features make it possible to perform sensitivity analysis for the selected parameter (or variable). Sub-physics and conditions (e.g., boundary conditions such as inflow, symmetry, heat flux, and loads) are added under the main physics. The user should ensure that these sets of input data are completed so that the problem solution can be attempted. In case a boundary condition is missing, the program may employ the built-in pre-assumed conditions for the missing regions (e.g., lines, areas, and volumes) if the user has not excluded them from the physics. For example, when defining the material for the first time, its properties are propagated to the entire geometry. However, this is not the case if the user decides to exclude parts of the geometry from the physics or material definition. Leaving the properties of the materials the same as the default values is advantageous in the sense that errors due to the lack of attributes (types) and properties (attribute values) may be avoided; however, the disadvantage of this method is that the user may inadvertently neglect to set up the attributes that are not shared among all the features such as radiation properties. Therefore, care should be taken when setting up the features' attributes and properties.

If a model consists of multiple components, all of them are selected in the order in which they were defined when the solution is performed. Additionally, the related mesh for each model is highlighted so that the user can ensure the analysis is performed correctly. It is possible to

exclude any number of analyses when performing the desired analysis. This is either done to complete the analysis in stages where the outputs of one analysis are to be used as inputs to another analysis, or where there is no desire in solving all analyses at once; for example, if the effect of including or excluding certain features (e.g., heat flux versus convection boundary condition) is to be studied. In the majority of scenarios, it is possible to set up a parametric study in which this variation of parameters is taken care of systematically; another benefit is that it is possible to compare the results in a single diagram. Employing the former feature, it is possible to select sets of data from each solution. Line and plane data associated with subregions (subsets of data) are created afterward; these items are then presented within a diagram as multiple lines or regions. In this work, the main focus is on the geometry import tool and CAD. COMSOL Multiphysics also includes extensive built-in visualization tools, but these will not be covered in detail here.

8.3 ONE-DIMENSIONAL GEOMETRY

Note that different selections of physics may be available for geometries of different dimensions (Figure 8.16). For 1D geometry, the user has two options to create a geometry—*Interval* or *Point* (Figure 8.17). The former option is similar to inserting a line that also includes its end vertices (points). For example, if the user were to apply specific boundary conditions to the ends (e.g., symmetry, temperature, and load), they are able to do so. Boundary conditions such as heat sources can be added to the line interval if needed.

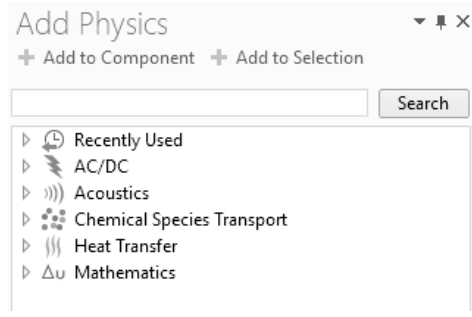


FIGURE 8.16. Example of physics creation options available for a 1D model—licensed module dependent.

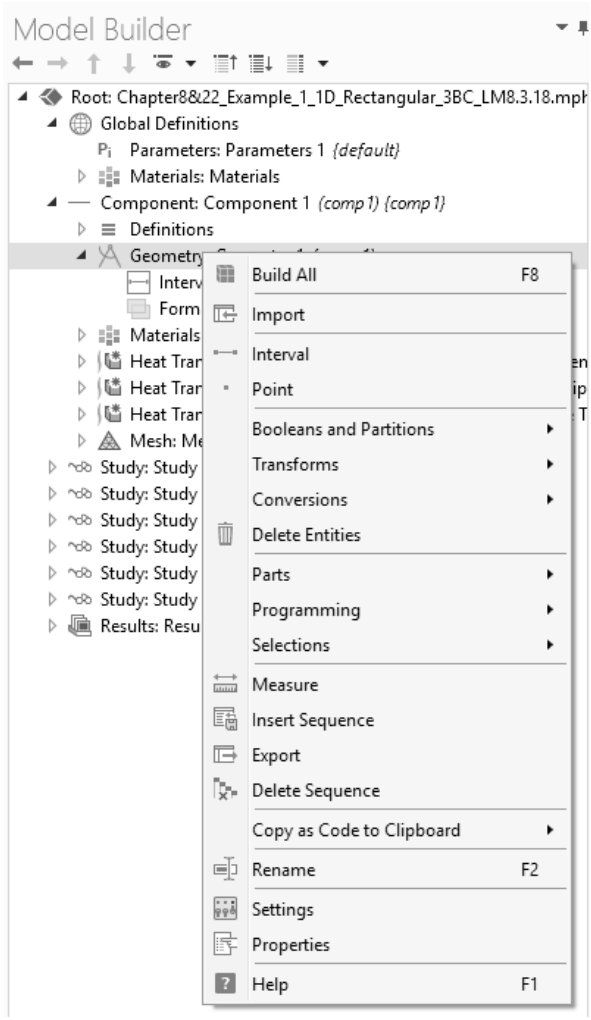


FIGURE 8.17. Geometry creation options available for a 1D model.

A number of methods may be used to create a 1D geometry. It is possible to *Import* the geometry features or subcomponents from a third-party CAD tool (e.g., Solid Edge) or create the geometry by means of built-in geometrical operations. After the geometry creation is completed, it can be updated by applying *Build All*. This feature is specifically helpful when importing the subcomponents.

A collection of operations on geometrical features is available under the *Transforms* menu (Figure 8.18a). When creating a geometry, the user may wish to build a set of features in a pattern, or simply create multiple copies of them with no specific pattern—located at certain coordinates. The *Array* feature is used for the pattern creation, while

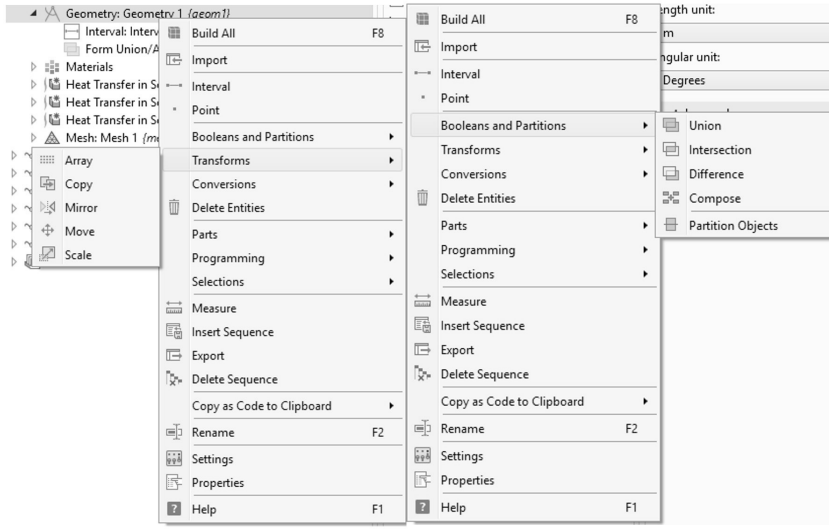


FIGURE 8.18. Geometry creation operations available for a 1D model: (a) *Transforms* (left), (b) *Booleans and Partitions* (right).

the *Copy* feature is applicable when a feature is to be duplicated and moved to any arbitrary location. The individual features or a cluster of them (e.g., a combination of lines making an L-shape) can be created using the *Copy* feature and transferred by the *Move* command to a certain location relative to another feature (e.g., far-left corner of the geometry or where the geometry zeros are set). It is also possible to create the object's reflection (*Mirror*) or change the object's size proportionately (*Scale*).

Booleans and Partitions is a collection of operations such as *Union* (joining the subcomponents); *Intersection* (finding the common feature among components); *Difference* (deducting one subcomponent from the other one—assuming they have common features); *Compose* (adding multiple subcomponents); and *Partition Objects* (dividing the subcomponents into smaller regions—useful when defining mesh regions or regions with different properties such as mechanical, boundary conditions, or contact. The *Boolean and Partitions* operations are available for all geometrical dimensions (Figure 8.18b). The *Conversions* feature makes it possible for the model to be united as a single domain (e.g., a single solid domain consisting of multiple solid sub-domains or a composite domain), split into its subcomponents (e.g., areas, forming composite features), or be divided into pieces by means of partitions. For the latter case, it is possible to maintain the original subcomponents such as points and lines if needed (keep input objects) or define a new set of fully independent objects (Figure 8.19).

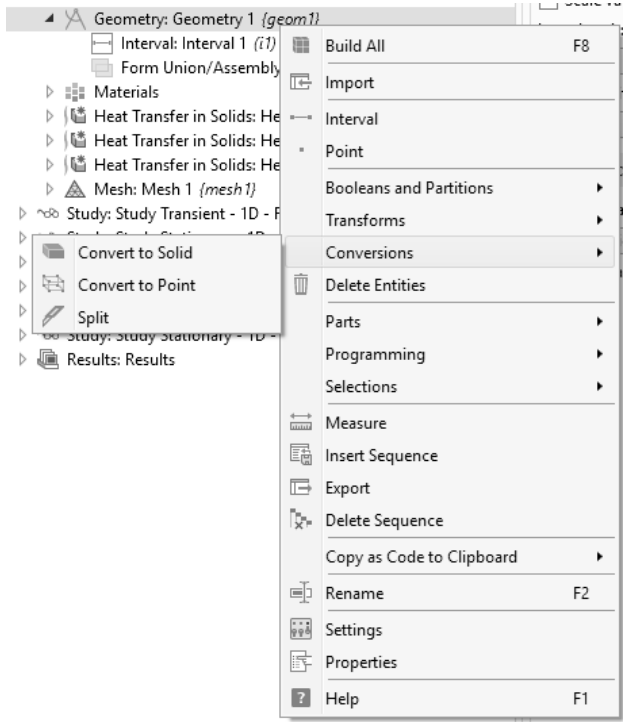


FIGURE 8.19. Geometry creation *Conversions* operations available in a 1D model.

In COMSOL Multiphysics, there is a limited number of built-in geometries available under *Parts* that can be imported into the model (Figure 8.20a). Reviewing the options available, the user finds out that they are categorized by the available physics—*Part Libraries* (Figure 8.20b). Note that COMSOL Multiphysics-related library parts are available with select modules. The user can choose from the library parts or geometries only if the geometry dimension allows for such selections. For example, a 3D geometry may not be importable into a 1D geometry; however, the user may create 3D geometries from 1D sketches defined within a 3D environment.

If the user is a keen programmer, COMSOL Multiphysics provides the opportunity to use this capability in order to generate or manipulate the geometries to some extent. Use of *If-End If* capability makes it possible to create geometry components by clustering them into regions that follow certain conditions—meaning if those coordinate conditions are met, one can create these features (Figure 8.21a). This also allows the user to modify the geometry based on some parameter values. This way the user can do parametric studies that modify the geometry, for example, by defining the number of holes in the plate as a parameter

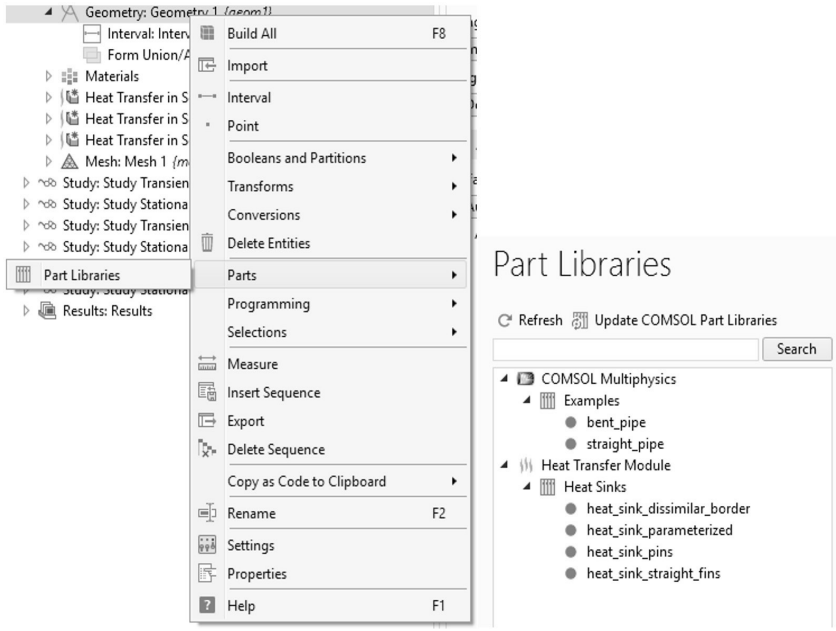


FIGURE 8.20. Geometry creation operations available in a 1D model: (a) *Parts* (left), (b) *Part Libraries* (right).

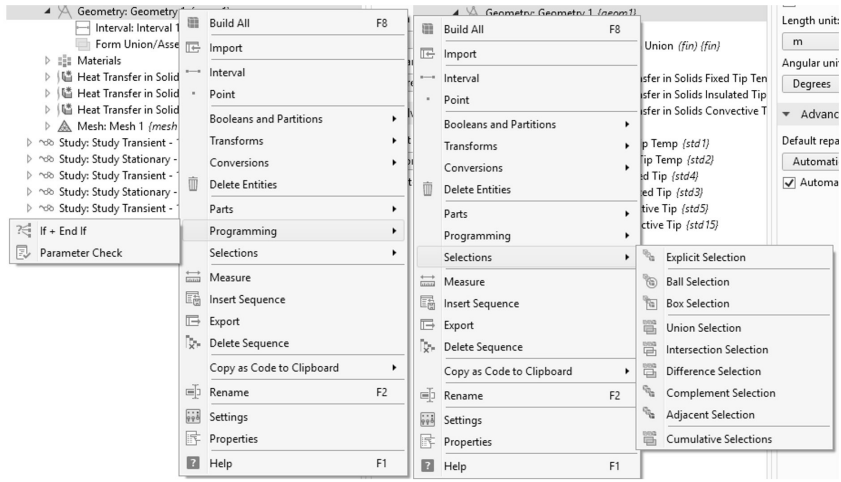


FIGURE 8.21. Geometry creation operations available in a 1D model: (a) *Programming* (left), (b) *Selections* (right).

and then creating a corresponding number of circles to define them depending on the if-then evaluations. The *Selections* menu offers different means to create a selection (Figure 8.21b). This is a collection of identification numbers corresponding to geometric entities which

can then be used in various ways. For example, if you want to define a particular boundary condition on a certain subset of edges, you can create a selection and then reference numbers of that selection when defining this boundary condition. If you change the line segments that go into this selection set, this will change where this boundary condition is applied. Alternatively, you can reference this selection set in several places, making setting up your model more efficient and less prone to errors. Using the operations under the *Selections* menu allows you to make the choice of items that go into this particular selection independent of their identification numbers. For example, if you change the geometry, the numbers assigned automatically to the geometric entities may change, affecting your selection if it is manually created (*Explicit Selection*). Examples of available *Selections* operations are: *Box* (entities fitting within a rectangular area); *Ball* (entities fitting within a spherical volume of a specified radius); *Intersection* (entities that are in common between two or more input selection sets); and *Cumulative* (combination of a number of selections)—Figure 8.21b.

This feature is particularly useful if the user decides to only activate the selection either for display purposes or the operational point of view (e.g., performing Boolean operations or applying boundary conditions to specific boundaries or regions). Imagine you wish to remove certain nuts from the nuts-and-bolts selections that are clustered together. Imagine also that the nuts are made of numerous lines and surfaces, the individual pieces that are challenging to select one by one. The selection feature option makes it possible for these individual items to be chosen efficiently with minimum chance of error. A relatively complex geometry that consists of many intertwined subcomponents, such as entities within other entities, can only be fully accessible if the selected features can be sent to the background (view turned off) so that the hidden parts can become visible (versus hidden). Although there are other display features such as transparent box and wireframe where the interior may be to some extent visible or the surfaces are eliminated, the confusion of dealing with the views remains when dealing with complex geometries, especially in the case of wireframe, where the surface areas are nonexistent and lines (and vertices) may be the only viewable features.

The *Measure* feature is a very useful capability that allows the user to make measurements between and of components such as points, lines, areas, and sizing options. The displayed units depend on the selected system of units. The user can measure the length of a line, area of a surface, or volume of an object. On occasion knowing the

distance between points is extremely helpful; for example, the user may need to move subcomponents by an exact distance to a precise location—a point or intersecting lines. The resultant dimensional values may be copied into a spreadsheet (e.g., Microsoft Excel) or text editor (e.g., WordPad) for further processing. This ability is particularly important since this avoids manual data entry—reducing chance of error and transferring the values accurately, to the full number of decimals available.

It is also possible to make a number of selections (e.g., line, areas, or volumes) and show the cumulative size of the subcomponents. Recall using the BOM feature that was available in some CAD tools such as SolidWorks. A similar idea may be achieved in the FEM tool if the user can proficiently combine the tool's capabilities with their sense of design. Imagine you wish to select the regions of similar materials. Imagine that the geometry has undergone changes after the import. You are able to measure the region using that material before the changes were made by selecting the *Geometry objects* option from the drop-down menu within the *Measure* toolbox appearing in the window. Alternatively, you are able to measure the regions by selecting the *Finalized geometry* option from the same drop-down menu.

You can then make a comparison between the two selections and find out the changes to the material used—assuming you know the density of the material (the volume by the density results in the mass). The importance of the *Measure* feature cannot be overstated. In the following sections, an attempt is made to provide some examples of the use of this feature and possible options to verify and validate them using calculations based on the presented parameters (and variables). So, remember this capability in your work—you may find it will save you a lot of headaches one day!

The geometry may be imported in a sequence of steps from the previously saved or exported files or exported to the chosen formats (binary or text) (Figure 8.22a) and COMSOL Multiphysics versions (Figure 8.22a). Geometries created by the previous versions of the FEM tool can be opened and operated upon within the newer versions of the FEM tool; however, the opposite is not valid. This feature provides you with the means to deal with the incompatibility issues that can arise when transferring data from newer versions of the FEM software to the older ones. The only disadvantage is if the user has an updated FEM tool and wishes to collaborate with an individual who has an older version of the FEM tool for one of a variety of reasons, such as not having access to the new updates—for example, the potential

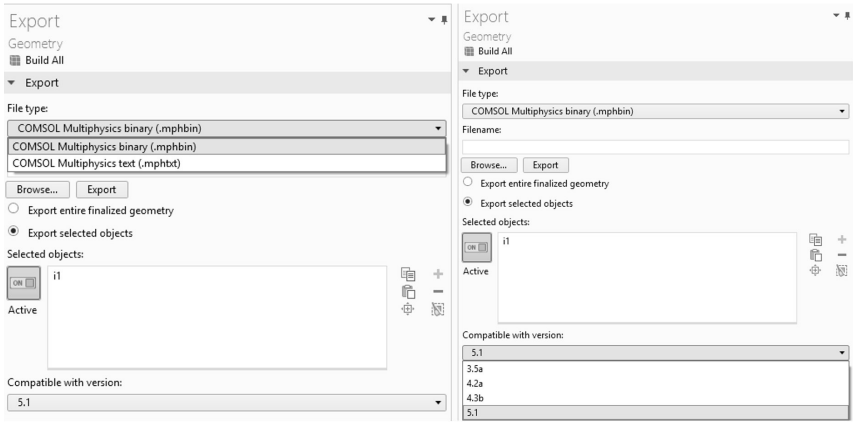


FIGURE 8.22. Geometry export options: (a) *File type* (left), (b) COMSOL Multiphysics *Compatible with version* (right).

collaborator has not renewed the licenses. If this is the scenario, then the user would be able to review the files using the tool's newer versions; however, it would be smart to pay careful attention if they wish to make major changes. The base geometrical kernels are common among the versions; however, the added features, change of interior formulation, or the method of their definitions and presentations might be among the reasons for the older versions not to be capable of interacting with the newer versions. Similar to the scenario that the geometry may be created by means of addition of a sequence of geometry input files or commands, it is also possible to be revised by removing the chosen sequence.

Generally speaking, a good feature of COMSOL Multiphysics geometry creation toolbox is its capability to undo the steps taken in order to include features, delete them as required, or simply deactivate (disable) them. The latter feature is similar to suppressing features in the CAD tools—where the user can choose what geometry-creation steps to include in the displayed or exported geometries. The geometry may be *Copied as Code to Clipboard* (Figure 8.23) with multiple options such as *Create*, *Get*, and *Run*. *Set all* may also be available when the user wishes to impose their own properties. This feature technically allows the user to transfer the nodal data to a code (e.g., MATLAB script language). The data may be copied to a clipboard similar to copying done when using other tools such as Microsoft Excel. *Create* makes a new object based on a new code, *Get* receives the data from a preexisting object and presents it in the form of a code, and *Run* builds a code that can run in the provided sequence(s).

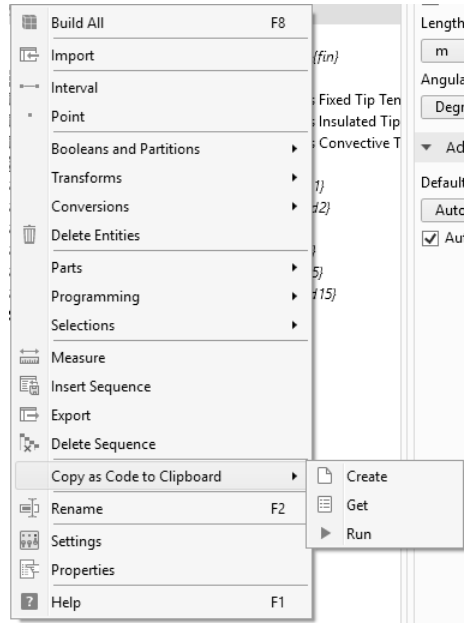


FIGURE 8.23. Geometry creation options available in a 1D model.

Use of *Parameters* (or *Variables* when synchronizing with the CAD tools through the LiveLink feature) is recommended when setting up geometries. *Parameter* may be defined both globally (Figure 8.24a) and locally (Figure 8.24b).

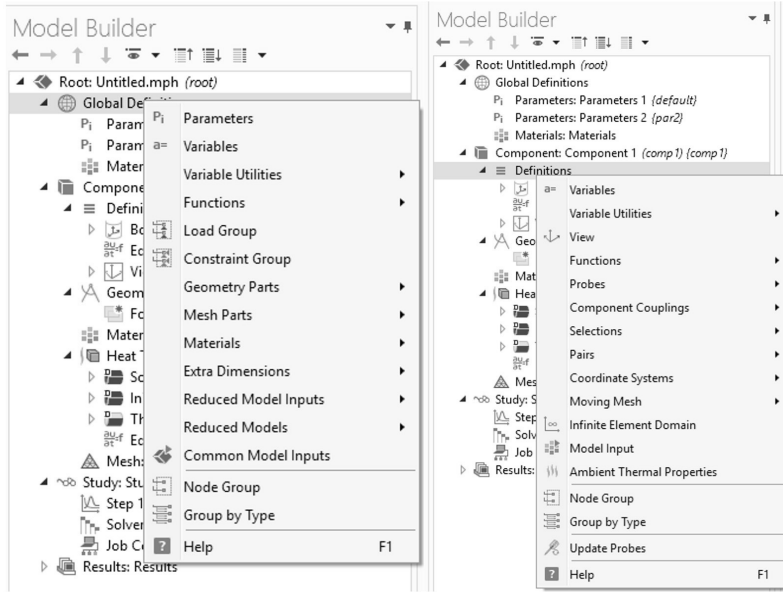


FIGURE 8.24. Setting up parameters: (a) Global (left), (b) Local (right).

8.4 TWO-DIMENSIONAL GEOMETRY

After making the decision as to how to create a 2D fin, either as a separate model file or a new subcomponent within the same model file, the user needs to define the geometry. In general, the geometry may be created either in a dedicated CAD tool such as SolidWorks and imported into COMSOL Multiphysics or by using COMSOL Multiphysics CAD import tool, depending on the modules available to the user. Simple shapes may be created with COMSOL Multiphysics with relative ease. Geometry creation operations and features available for a 2D model are accessible by right-clicking on the *Geometry* node; there you can also access *Virtual Operations* (Figure 8.25).

Some operations are the same as those for 1D; new operations apply to 2D geometry shapes such as circles, polygons, segments, interpolation curves, and parametric curves. Corners can be modified by a *Chamfer* (a linear cut through the corner at an angle) or by a *Fillet* (a circular rounding) (Figure 8.25 and Figure 8.26). *Virtual Operations* shown in Figure 8.25b allows you to make changes to the geometry that are not actual changes; however, they make the program treat the geometry as if the actions specified by these operations have been carried out; for example, two domains can be combined into one (*Composite Domains*) or vertices located at the vicinity of one another within the given tolerance can be merged. It is possible to control the features such as mesh as well.

To illustrate how the mesh control features can be useful, consider the following example. You are interested in modeling the airflow and temperature distribution inside a hangar where aircraft are parked. The hangar is a large rectangular box containing air. The model is to represent the case of an aircraft fire. There are air ducts (inlet and return) located in the hangar ceiling. You need to be able to accurately model the inlet air flow and temperature. For this to happen, the geometry around the duct area should be meshed finely. However, if this fine mesh size is maintained through the entire hangar volume, the number of elements would be very large. To manage the mesh density, mesh control features are defined in which you can select regions in the vicinity of the ceiling ducts to which fine mesh settings can be applied. The rest of the geometry can be meshed using a coarser mesh. In this way you are able to reduce the total number of elements, making better use of time and memory resources. You may even decide to perform a sensitivity analysis to pick the smallest region possible that gives acceptable model accuracy.

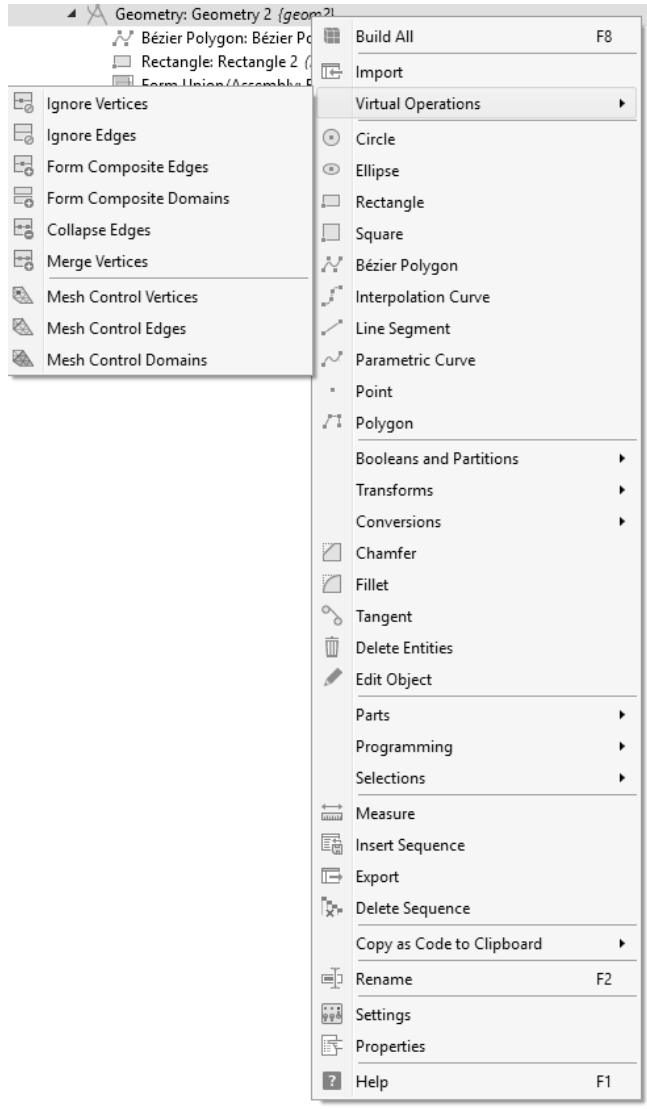


FIGURE 8.25. Geometry creation operations and features available for a 2D model, including *Virtual Operations*.

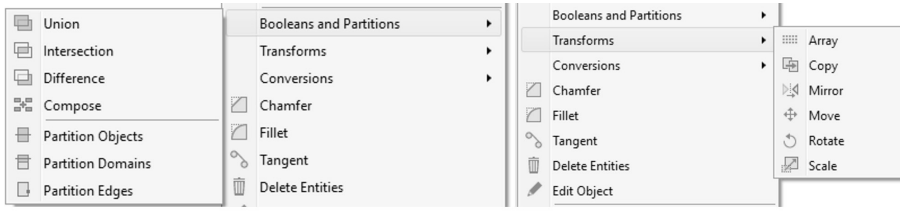


FIGURE 8.26. Geometry creation operations and features available for a 2D model: (a) *Booleans and Partitions* (left), (b) *Transforms* (right).

There may be also cases where the edges are so close to each other that the selected element size does not fit into the small space; this can occur, for example, close to corners defined by very small angles. You may receive an error message for those regions saying that the minimum element size exceeds the size of the region or edge. In this case, *Virtual Operations* makes it possible to unite the closely located faces or edges so that the element size limitations may be avoided (Figure 8.25).

Figure 8.26a is related to the *Booleans and Partitions* features available for a 2D model with the added partitioning capability for domains and edges. Figure 8.26b is related to the *Transforms* operations, which are similar to that of the 1D model with the added *Rotate* feature—the part(s) may be rotated with any order or as clusters. Figure 8.27 is related to the *Conversions* features available for a 2D model. It is similar to that of the 1D geometry with the added *Convert to Curve* operation. As seen, this feature closely follows the geometry dimensions.

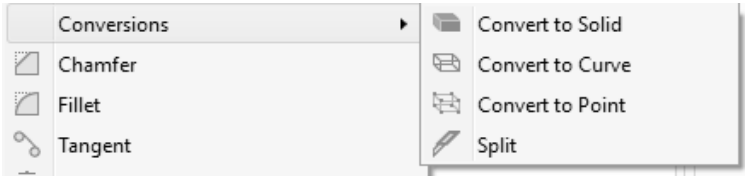


FIGURE 8.27. Geometry creation *Conversions* available for a 2D model.

8.5 THREE-DIMENSIONAL GEOMETRY

Similar to the 2D analysis, the 3D model can be created either by making a new model (*File* > *New*) and following the steps to define the physics and study type, or by building upon the 2D study by adding a new 3D component. Using the latter approach makes it possible to compare the results of 3D and other types of analysis (e.g., 1D) in one plot. For this to happen, the user can select a subset of any solution and compare the corresponding features such as lines and areas. For example, it is possible to select a line in any given space dimensions (1D, 2D, and 3D). The data associated with the selected line may represent the temperature distribution within the solution data set along the given line profile. The line in a 1D scenario may correspond to a line in 2D and 3D. Thus, one can study the effect of

progressing from the 1D to 2D and 3D models. The 3D geometry can either be created by importing a solid geometry that is generated in a CAD software such as CATIA or made directly inside COMSOL Multiphysics. Figure 8.28a shows the 3D features available under geometry when building solid models. The features are divided into eleven sections. The build part is shared among the 1D, 2D, and 3D geometry features. The *Defeaturing and Repair* submenu is presented in Figure 8.28b, where the user can perform deletion operations for fillets, sliver edges, sliver faces, and spikes. The *Cap Faces* capability is helpful if the user wishes to create a volume corresponding to the air contained, for example, in a duct. To help you understand this functionality, imagine filling a plastic bottle with water, closing it with a cap, and placing the bottle inside a freezer. After a while, you will end up with a bottle with solid ice—make sure to use a plastic and not a glass bottle, as the glass may crack with water expansion upon freezing. If you could cut away the bottle, you will find a frozen shape that adopted the interior shape of the bottle. This frozen water volume is the negative of the plastic bottle volume. A similar analogy applies to more complex geometries. For example, analyzing heat transfer-flow for a winding pipe that has different attachments (interior or exterior) is a complex task. Therefore, the specialist may choose to simplify the model, focusing on the flow channel, by eliminating any features, which can be considered to have negligible impact on the flow while modeling larger interior features as walls. By using *Cap Faces* for any openings for this pipe, a complex interior volume can be created and used to model the fluid flow.

There are *Knit to Solid* and *Repair* features at the end of this section that can be used when importing complex geometries with some functionalities similar to creating composite domains mentioned earlier. Selecting the *Repair* feature opens a window that lets the user choose the desired repair tolerance (10^{-6}). It does sometimes work; however, often there will be cases where the geometry cannot be repaired. If you are faced with a problem and cannot find a solution, one approach is to responsibly and ethically use the specialized social platforms (e.g., COMSOL Multiphysics community) to see if someone else can help or has encountered similar problems [173]. This is one example in which the responsibility (collective effort) and accountability (individual effort) become meaningful. This concept will be further discussed in Chapter 12 (“Lean Six Sigma Implementation”).

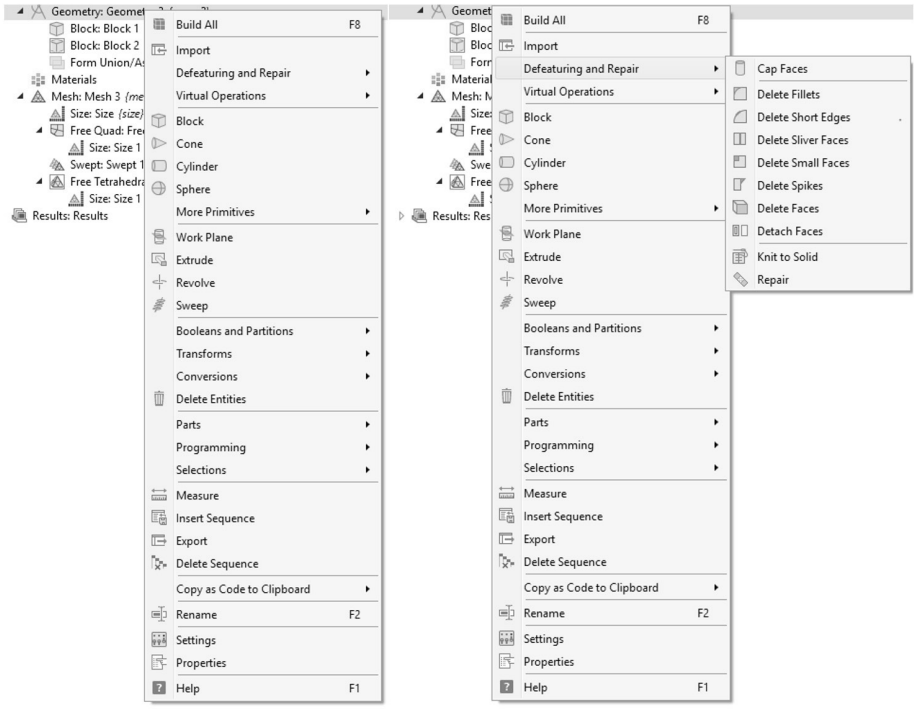


FIGURE 8.28. Geometry creation operations and features available in a 3D model: (a) Geometrical features and operations (left), (b) *Defeaturing and Repair* (right).

Virtual Operations available for 3D geometry are presented in Figure 8.29a. They include all the operations listed for 2D geometry (manipulation of vertices and edges) plus added operations on the domains (3D volumes). Figure 8.29b lists additional geometrical features available under *More Primitives* (in addition to the blocks, cones, cylinders, and spheres shown under the main geometry menu). The CAD Import module makes it possible to create more geometrically complex shapes such as a torus (imagine a donut), ellipsoid, helix, interpolated and parametric curve, and Bezier polygon. Figure 8.30a presents the *Booleans and Partitions* operations for a 3D model with the added capability to partition faces. Figure 8.30b shows the *Selection* features available for a 3D model. It is similar to that of 2D geometry with the added *Cylinder Selection* along with the last portion that is associated with *Color Selections*. The *Transforms* feature is similar to the one presented for 2D. Figure 8.31 shows the *Conversions* feature available for a 3D model. It is similar to that of the 2D model with the added *Convert*

to Surface and *Convert to* COMSOL Multiphysics operations. The latter makes it possible for the geometry domains (or objects) created in Parasolid geometry kernel to be translated to COMSOL Multiphysics geometry kernel, assuming that COMSOL CAD module is employed to import the geometry.

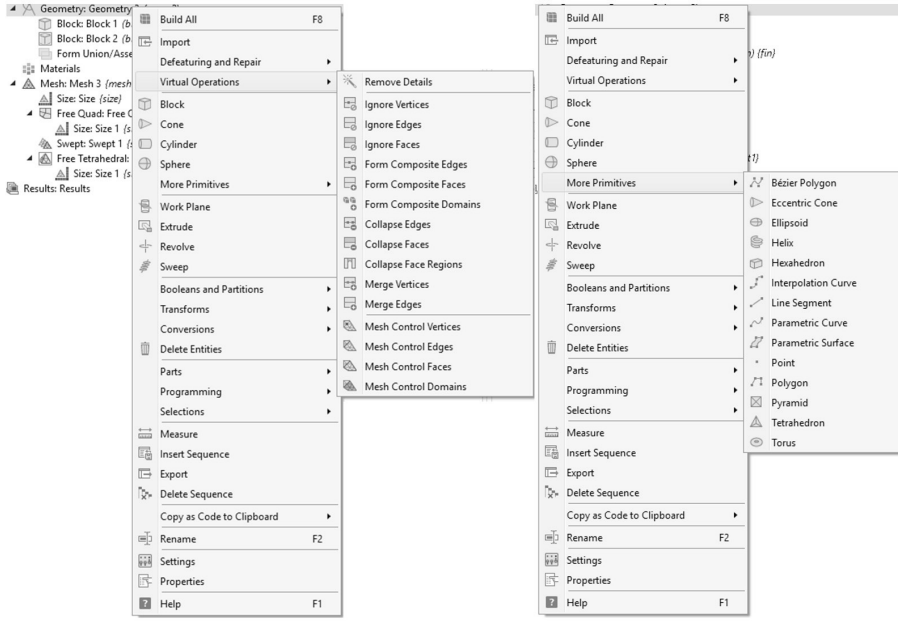


FIGURE 8.29. Geometry creation operations and features available for a 3D model: (a) *Virtual Operations* (left), (b) *More Primitives* (right).

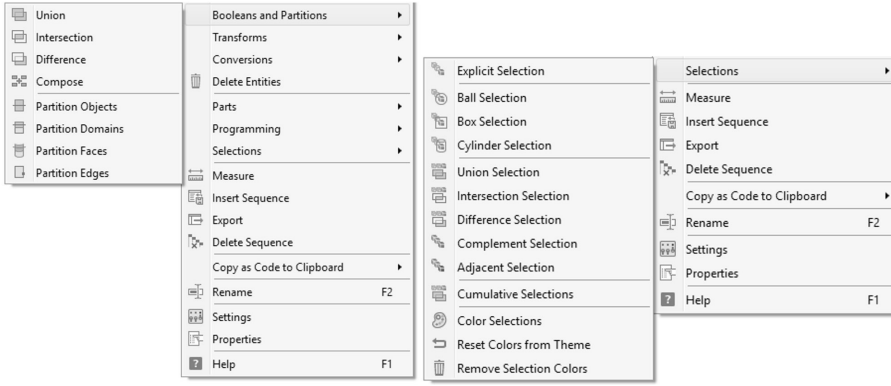


FIGURE 8.30. Geometry creation operations and features available for a 3D model: (a) *Booleans and Partitions* (left), (b) *Selections* (right).

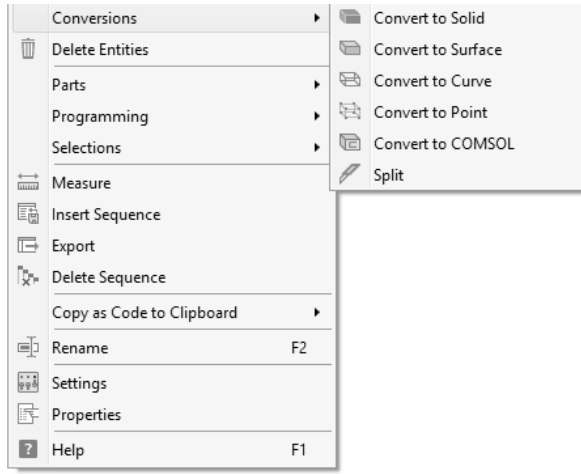


FIGURE 8.31. Geometry creation *Conversions* operations available for a 3D model.

8.6 WORK PLANES

Work planes are used extensively in COMSOL Multiphysics. The first application of work planes is to create 3D shapes by extruding, revolving, or sweeping the 2D profiles sketched on the work plane. The second application is to use the work plane itself to partition (cut) objects.

To sketch in the work plane, select the *Plane Geometry* node after creating the work plane. Right-clicking on this node, you will be presented with all the commands for 2D geometry creation available. It is also possible to incorporate and use inside the work plane the cross section of a 3D solid intersected by the same work plane.

When used for partitioning, work planes can divide the entire geometry or just the selected domains into regions or sub-domains. These separated domains then can be manipulated by means of Boolean expressions such as *Intersect* and *Difference* in order to create new regions. It is also possible to delete or join portions of the divided geometry. Delete Entities is a useful feature that is extensively used in this work. Creating composite features by joining the selected regions of the divided geometry ensures that the domains are treated as one when assigning material properties or meshing. It is possible to maintain or eliminate the boundaries between the input domains or areas and create a seamless region if needed.

To create a new work plane, right-click on the Geometry node (Figure 8.32a). Figure 8.32b shows the settings available when defining a *work plane*. There are four regions within the *Work Plane* settings window: *Plane Definition*, *Local Coordinate System*, *Unite Objects*, and *Selections* of *Resulting Entities*. The first region includes *Plane type*, *Plane*, *Offset type*, and *z*-coordinate options. Figure 8.33a lists available plane types.

The most basic work plane type is *Quick*. It restricts the user to creation of planes parallel to one of the three principal planes of the model's coordinate system. When the plane is created, it has its own local coordinate system (*xw*, *yw*, *zw*), with the *xw* and *yw* axis being in the work plane. When choosing the principal plane parallel to which to create the work plane, one can choose to have the *zw* axis align with either one of the two model plane coordinates, resulting in six options: *x-y, y-z, z-x, y-x, z-y,* and *x-z* planes (Figure 8.33b). For example, if you select the *y-z* plane, a work plane with an *xw* axis parallel to *y*-coordinate and a *yw* axis parallel to *z*-coordinate is created (Figure 8.34). If you chose the *z-y* plane instead, an *xw* axis will be parallel to *z*-coordinate and a *yw* axis will be parallel to *y*-coordinate (Figure 8.34).

Offset type and *z*-coordinate options are selected next (Figure 8.35). The first setting defines where the work plane will be located along its *zw* axis. By selecting the *Offset type*, this location can be defined either as an offset (*Distance*) from the model's coordinates origin (under *z*-coordinate) or by selecting *Through vertex* in the model through which the plane will pass. In the example in Figure 8.34, the *y-z* and *z-y* planes are both located at 20 mm along the *x*-coordinate from the model's origin.

Work planes can be also created using several other, more advanced, methods (Figure 8.33a). For example, the *Vertices* method allows plane creation by selecting three vertices from the model which will then define the plane (as three points are sufficient to define a plane). To locate the work plane, an offset can then be specified along the direction normal to the plane of these three vertices.

Location and orientation of the work plane's *Local Coordinate System* can be defined by means of settings grouped under the relevant heading. The settings under *Origin* and *Local x*-coordinate define the location of the frame's origin and the direction for its *x*-coordinate (Figure 8.36). The *Global* origin option aligns the local frame with the model's frame (except for the offset as defined previously). For example, in Figure 8.34, the *Global* option is used and the local plane origin is located at the point (20, 0, 0) of the model's coordinates. Selecting the

Vertex projection option allows you to define the local origin location by projecting any vertex from the model onto the work plane. Under *Local x-coordinate*, the *x*-coordinate orientation can be kept as aligned with one of the model's axes by using the *Natural* option. For example, in Figure 8.34, *xw* is aligned with the model's *y*-coordinate by this method. Choosing the *Through vertex projection* option will make the *xw* axis pass through the projection of the chosen model vertex. Additionally, the local coordinates can be offset along the local *xw* and *yw* axes and rotated about the local *zw* axis by entering values into the *xw*- and *yw-displacement* as well as the *Rotation* fields (Figure 8.36a).

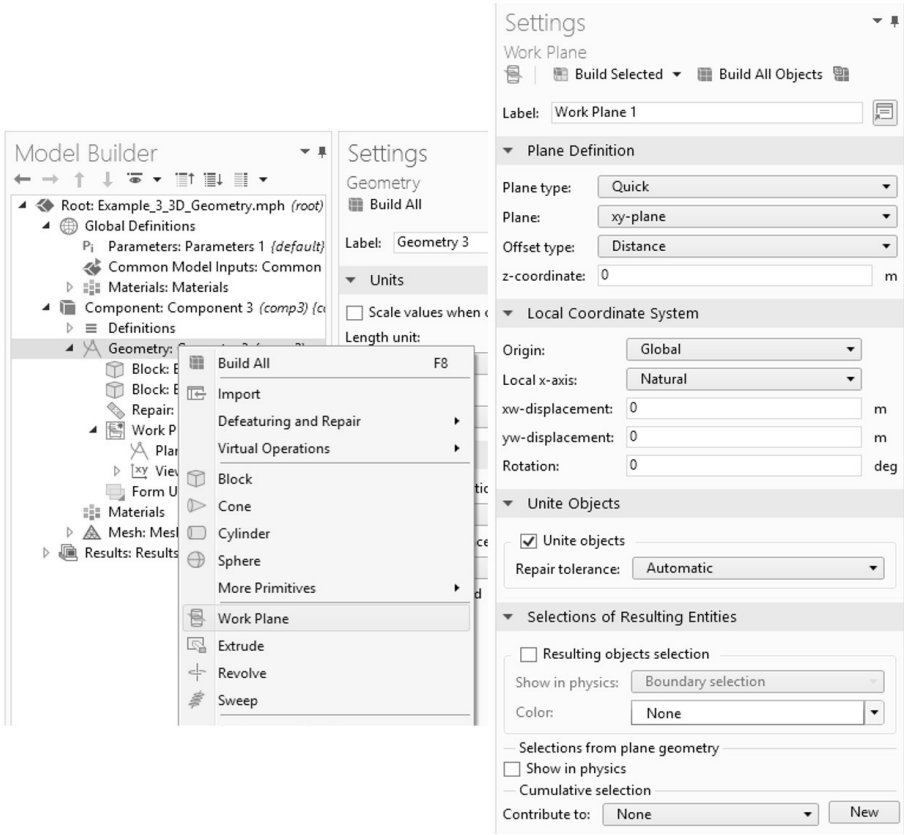


FIGURE 8.32. (a) creating a *Work Plane* from the geometry menu, (b) *Work Plane* selections.

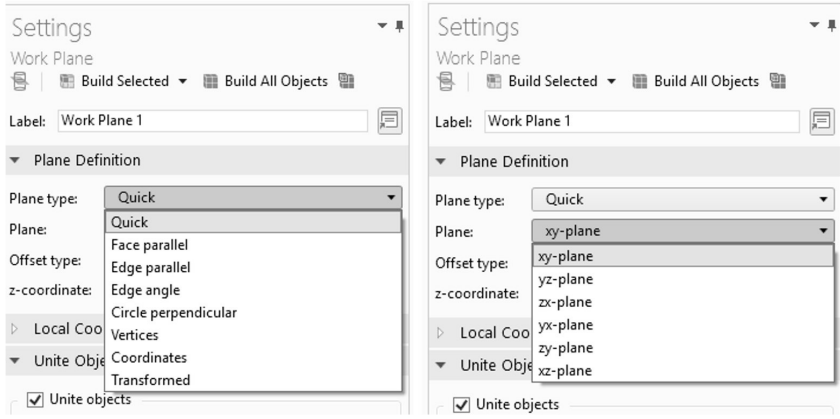


FIGURE 8.33. *Plane Definition* options when creating *Work Plane*: (a) *Plane type* selections, (b) *Plane* selections.

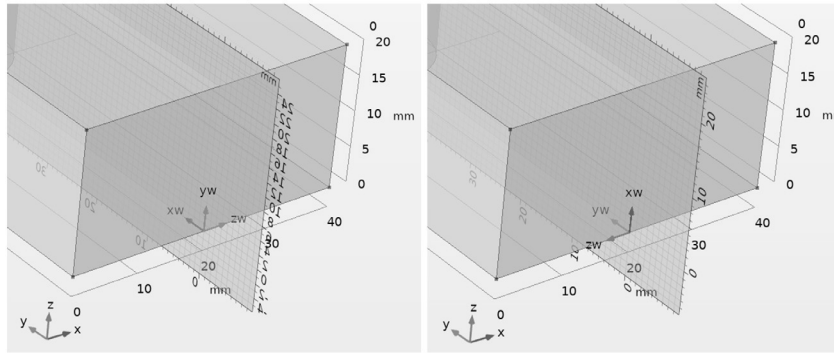


FIGURE 8.34. Two possible orientations of the work plane coordinate system: (a) *y-z* plane, (b) *z-y* plane.

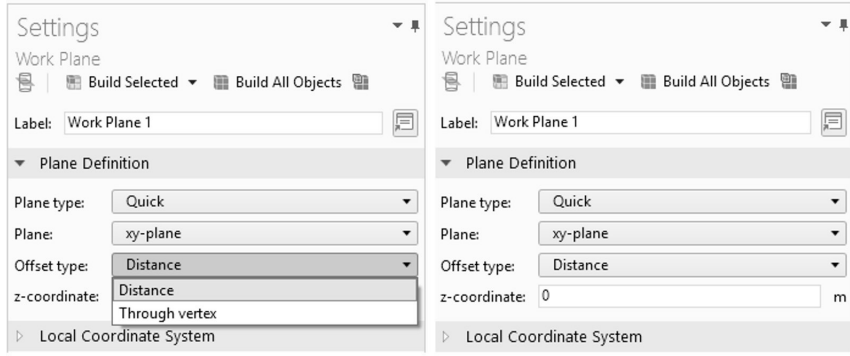


FIGURE 8.35. *Plane Definition* options when creating work planes: (a) *Offset type* selections, (b) *z*-coordinate.

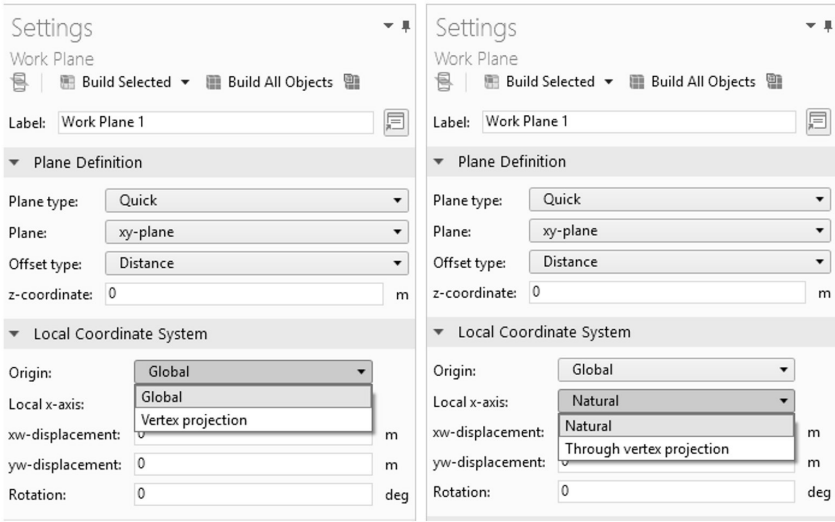


FIGURE 8.36. *Local Coordinate System* options when creating work planes: (a) *Origin* selections, (b) *Local x*-coordinate selections.

CHAPTER 9

EXTENDED SURFACES

The use of extended surfaces as an efficient thermal management method is a common engineering practice. In their design, choice of material is as important as the geometry itself. For example, a heat pipe is a cooling mechanism that has been effectively employed in space applications. Heat pipes consist of a wick structure that directs the heated fluid from the warmer side to the cooler side by means of capillary forces. The heated molecules that are in a gaseous state (vapor) travel upstream to the cool side (condenser), release the surplus heat, and then form liquid, which then travels to the warm end (evaporator) of the pipe, and the cooling cycle repeats.

In some aerospace applications aluminum sheets are used as the heat pipe envelope [174]. The design objective in this case is to ensure that the space nuclear systems are operating within the recommended range of 130 °C to 280 °C. Although aluminum is easily machinable, allowing for manufacture of interior longitudinal grooves to increase surface area of the heat pipe envelope and consequently achieve more efficient heat transfer operating conditions at this elevated process temperature is not as suitable for this material. Therefore, titanium alloy has been suggested as an alternative; though more expensive than aluminum, it has a higher melting point and a higher strength-to-weight ratio. Those characteristics, in addition to its anticorrosive properties, make it a desirable material in aerospace applications. The main challenge in using the titanium is its machinability.

In structural mechanics, extended surfaces are equivalent to cantilever beams or plates that are connected or anchored at one side to another structure. Imagine yourself doing a wall training exercise where your back is against the wall as you hold on to it with your hands raised

and your legs are held (with great effort) straight out, perpendicular to the wall. You may also imagine doing a peacock position—that's where you are balancing your whole stretched-out horizontal body with your hands planted on the ground. In either case, you are now a living cantilever beam. The cantilever design makes it possible for the overhanging structures (e.g., planters, buildings, bridges, and benches) to extend into space without supporting bracings. This is contrary to the structures that are supported at both ends—such as the downward facing dog. On occasions, the structures are built in a cantilever form from two sides, so as to close the gap by joining in the middle. This is the case for cable-stayed bridges, where the two sides eventually meet (e.g., Millau Viaduct bridge in France, whose road surface is 277-meter high above the ground with six 342-meter spans, the highest cable-stayed bridge in Europe) [175].

Cantilever beams are also used in the construction of Micro-Electro-Mechanical Systems (MEMS)—for example, a resonistor (a device with a specific resistance to the passing electric current) and monolithic (a single crystal) resonator. The majority of these structures are made of polymers and silicon or its compound (e.g., silicon carbide). The cantilever transducers make the atomic force microscopy possible, or an array of cantilevers can function as a biosensor in a diagnostic application. Radio frequency filters and resonators also employ this structure [176]. In another MEMS structure, a chemical sensor is created by coating the top surface of a microcantilever beam with a recognition receptor. An immunosensor (a biochemical test to measure molecule concentration is a solution) is thus created that provides the data on an antibody interacting with a specific immunogen (a substance creating the immune response). The beam bends differently relative to a reference microcantilever given different immunogens, and as a result the sensor response varies in the static mode. When operating in the dynamic mode, the variation in the beam resonance frequency shows the concentration of the analyte (a substance whose chemical composition is being measured) [177].

If you walk through a warehouse, you might notice the storage system, which includes vertical columns, bases, arms, and bracings made of structural steel. These are cantilever racks. The connection between the columns are by means of braces. When carrying your tools within the workshop, you may use a cantilever folding tray, which is made of stacked shelves that are hinged to each other (like an accordion) or to the main body of the box enclosing them. The unused shelves can be collapsed if needed. This packing system facilitates organization and accessibility of the stored items. Your makeup carry-on may follow the same mechanism for similar reasons. Enlarge one of the said shelves

of the cantilever folding tray and you have a cantilever chair, where the traditional four legs are absent and the seat is supported by the L-shaped legs. Mart Stam, a Dutch architect, furniture designer, and urban planner, patented the cantilever chair design (Figure 9.1) that was reportedly inspired by the 1926 Tatra T12 two-door saloon car. A cantilever brake is an important mechanical device that assists in safe operation of bicycles. In this design, each arm of the brake is connected to a distinct hinge, located on the seat sides, stay, or fork. Depending on the location of the brake shoe (above or below the pivot pressing against the rim when the arms are pressed toward each other), the brakes are known as first- or second-class lever design systems. These braking systems are particularly favored for mountain bikes where the tires are wide, for they can be designed to have large spaces between the pads and mounts. First-class cantilever brakes are further divided into two sub-categories (i.e., cantilever brakes and direct-pull brakes) and so are the second-class cantilever brakes (i.e., roller cam brakes and U-brakes) [178].

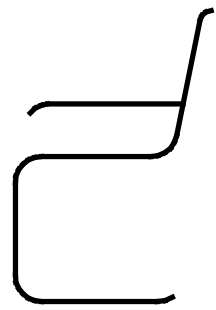


FIGURE 9.1. A cantilever chair (drawings created in Solid Edge).

Modern fixed-wing aircraft also have a cantilever wing structure, introduced by Hugo Junkers, a German aircraft engineer and designer who pioneered the design of all-metal airplanes and wings in 1915. Earlier aircraft wing structures were arranged in biplane or triplane configurations, with two or three stacked parallel wings connected by struts and strings—similar to truss bridges. It was soon discovered that the airflow around one wing adversely affected the other wing and the cross-braces introduced additional drag; therefore, there was a strong motivation to improve the design by minimizing the external bracing. In a typical modern cantilever wing, a large beam (spar) runs through the length of the wing, from the fuselage to the wing tip, closer to its leading edge (at 25 percent of the chord line—an imaginary line connecting the leading and trailing edges). The main spar connects to the fuselage (the main body of the aircraft) and carries the majority of the wing load.

This main spar is connected to the parallel smaller spar located closer to the trailing edge by means of the webs (the web-like structures that reinforce the wing but also support the airfoil shape required to generate lift) and/or the skin (in the case of monocoque or semi-monocoque planes). Since the 1940s, a majority of aircraft wings have used the cantilever design; one notable exception was the Messerschmitt BF 109E, a World War II German fighter that had bracing struts as stabilizers [179,180].

Cantilever beams (extended surfaces), also known as *fins,* are used as one of the primary thermal management measures. Looking at nature, one can recognize the use of these engineering structures in foliage (e.g., conifers), reptiles (e.g., crocodiles and lizards), birds (e.g., altricial birds and chickens), marine mammals (e.g., whales), and other types of mammals (e.g., deer and rabbits). The metabolic heat is transferred to the appendage (extended surface) by the blood flow and is then convected and radiated to the surroundings. One can say that most of the creatures creeping on the ground, having elongated body shapes, are in fact moving fins. Snakes for example have bodies with high surface-to-volume ratio and lose a lot of heat while moving; therefore, they can get cold quickly enough to be adversely affected—which explains their love of sunbathing. Dogs transfer heat by convection and radiation through their tongues. Remember seeing dogs sticking their head outside a moving car, with their mouth open and tongue exposed to the wind? They are just trying to keep cool. Their saliva evaporates, taking away the heat of their body and transferring it to the environment.

Your body parts can also behave like extended surfaces. Your arms and hands, for example, do so after taking a warm shower or lifting weights. Each segment of your limbs may be considered as a cantilever beam connected to the rest of the body at the wrist, elbow, and shoulder joints. You immediately receive feedback when you carry a heavy load or step out of the shower. Perform lateral bicep exercises, gradually increasing the weight, and you feel the tension at the shoulders and also the wrist joints. Make twenty repetitions and your arm is sore for a few days. You also feel the cooling due to water evaporation from your skin on your bare hands while the rest of the towel-covered body remains warm. Your endothermic bio-mechanism (that generates heat on its own) is activated, soon transferring heat to the colder regions of the body (hands in this scenario). Making a sandcastle by excavating the hot sand using your bare fingers also creates similar heating effects. The sand, which has been heated by the Sun's radiation, transfers some of the heat to the fingers and hands. This heat is gradually conducted to the rest of your arm. Your feet are also extended surfaces; make a note of it next time you give them a pedicure, soaking them in a cold basin

of water. Their warmth is almost immediately felt by upper sides of the leg such as your shins. Your cold-blooded (ectotherm) crab friends, whose body temperature rises and falls depending on their surroundings, like to bask in the warmth of their environment to regulate their metabolic functions, which is partially done through their shells acting as extended surfaces.

Stirring sugar in your morning tea not only causes the liquid to cool down due to the endothermic nature of this process—heat is taken away from the surroundings—but also due to the heat loss by conduction into the cold metal spoon. That is why a small teaspoon is preferred to keep the temperature drop to a minimum. Or one can even use a wooden spoon due to its lower conductivity. Another example is a skillet with a large handle placed over fire; the handle does not get as hot as the skillet main body and can be safely handled.

In industry, extended surfaces have been used for thermal management in applications such as Light Emitting Diode (LED) lamps, soldering, and microprocessors. For industrial applications, fins may be designed in a variety of shapes and therefore may be easily adopted to the applications. They can geometrically differ in their cross section and side profile. Rectangular, circular, triangular, tapered, concave, convex, radial, and pin fin shapes may be employed. Fins may be an integral part of the component or assembled to the main structure by methods such as welding, clipping, wiring, and using standoffs and pushpins. The latter approach, where fins are assembled onto the main structure, can enhance thermal management as a retrofit solution. An example of an integrated structure is the oil-cooling system on the Lycoming O-320-E2D engine used, for example, to power the Cessna 172 aircraft. Another example is use of extended surfaces in heritage homes in floor, wall, and ceiling radiators. There are also cases in which fins are combined with other thermal management solutions such as using extended surfaces on heat pipes in thermal energy storage units or space applications, or fan-cooled fins attached to personal computer processors.

On occasion, their connection to the rest of the assembly is strengthened by using an intermediate medium such as specialized thermal tape, grease, or paint that can be used to improve thermal contact between the extended surface and the rest of the structure. In this case, the purpose of using a fin as an add-on geometry (feature) is to thermally manage the excess energy generated due to the exposure of the geometry (assembly) to intense heat sources. This extended surface should therefore meet the design requirements, and its performance is determined by the amount of heat it dissipates from strategic locations—also known as its thermal efficiency. In such cases, geometry variations result in changes to the thermal efficiency. Increasing the

fin surface area, for example, by enlarging the cross section or extending the length, improves the heat transfer to its surroundings. The rate of heat transfer levels off at some point, and the improvement may not justify the added manufacturing cost or extra space requirements. Optimization of the fin is therefore a compromise between the heat transferred and fin size, which is a function of its cross-sectional area and length. This is an important factor, especially in limited spaces and where arrays of fins are to be employed.

The following sections provide case studies of the extended surfaces (fins), which are really just slabs of different shapes. Thermal analysis of some of these geometries is presented in the related publication [1]. The reader is encouraged to not only read this text but also study the model files provided in the accompanying companion disc  . Geometry meshing and applications provided with the geometry models are the bonus features of this book. A brief overview of the provided example applications is given in Chapter 10 ("Geometry Models and Applications"). These examples can serve as stepping-stones—the platforms to step on safely—for the readers who are interested in learning more about the tool. For these case studies, the geometry creation features are presented with a progressively increasing level of complexity, so as to allow the reader to gain confidence in the subject gradually. The purpose of this presentation is to demonstrate how COMSOL Multiphysics tool is employed in order to create and mesh the models. The examples attempt to use as much as possible the geometry creation capabilities that are part of the base COMSOL Multiphysics package, although in a few cases, a dedicated CAD tool (Solid Edge) was used. The Author would like to hear about your experiences and the way you would interact with the material for advancing the field and simultaneously making integrity-based decisions.

9.1 FIN WITH RECTANGULAR CROSS SECTION

An extended surface with a rectangular cross section is shown in Figure 9.2. The 3D geometry is made of a base feature (wall) on the left connected to the extended surface (fin). Dimensions are shown for this example as general parameters, meaning that the reader may change the parameters *a, b,* and *L* as needed to obtain fins of varying proportions. For this exercise, the 1D and 2D geometries are presented and discussed as the initial steps, which can be taken prior to creating the 3D geometry. Initially, a basic design is presented; subsequently, a more complex design is introduced that embeds a rectangular cooling channel into the fin.

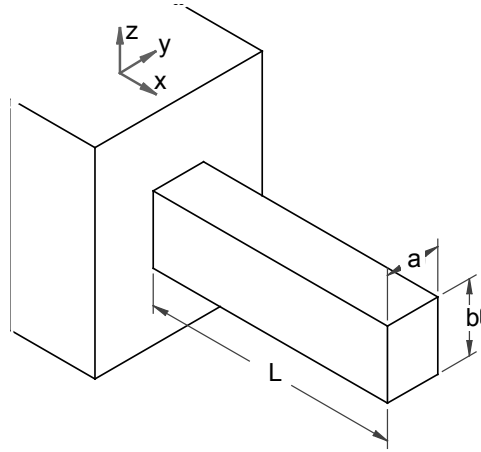


FIGURE 9.2. 3D fin with rectangular profile and rectangular cross section geometry (drawings created in Solid Edge).

9.1.1 One-Dimensional Fin Geometry

Procedures to set up a 1D geometry were discussed in Section 8.3 ("One-Dimensional Geometry"). Figure 9.3 shows settings of global parameters in the *Global Definitions* menu (under the main model setup). This provides flexibility when revising the dimensions (Figure 9.3 and Figure 9.4). The 1D fin in this example is created by using the *Interval* feature with a set length of $L = 0.1\ m$ (Figure 9.5). Metric or Imperial systems of measurement can be employed when creating geometries (Figure 9.4). The reader is to ensure that the units are not applied inconsistently, which may happen if the material properties

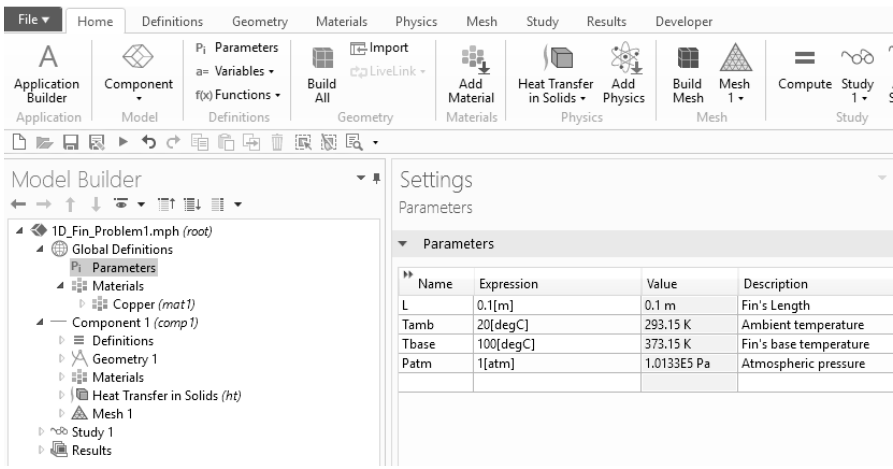


FIGURE 9.3. Parameters used to create 1D fin geometry (global level).

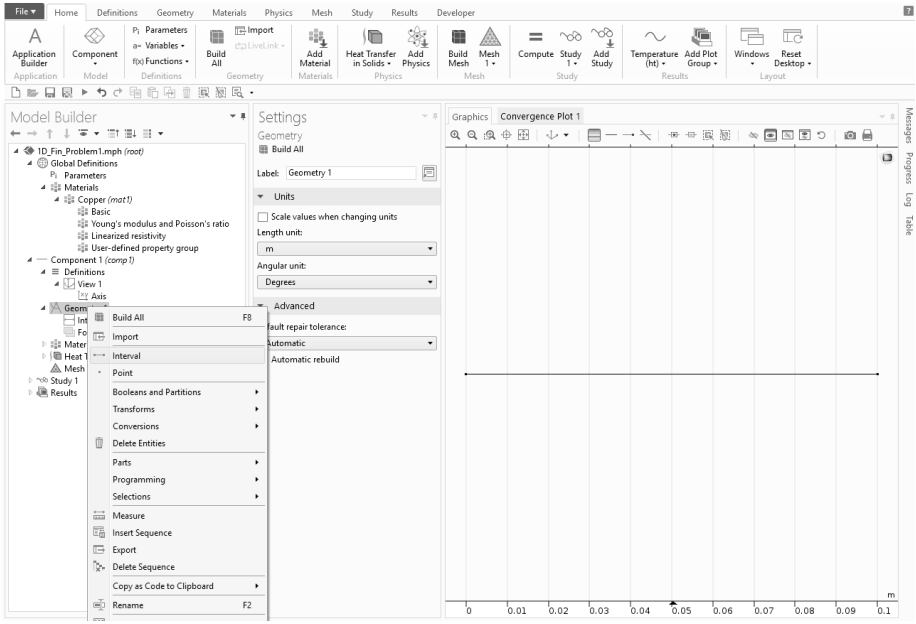


FIGURE 9.4. Geometry tree for 1D fin geometry by using parameters.

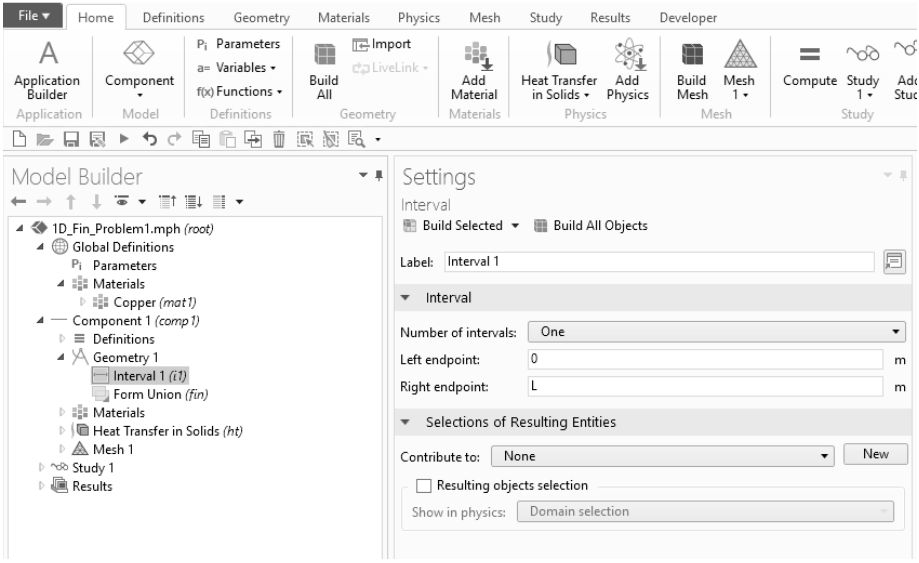


FIGURE 9.5. Setting up geometry parameters for 1D fin geometry (*i1*).

units are not checked carefully or if the dimensions of the imported parts are not consistent with those of the model.

9.1.2 Two-Dimensional Fin Geometry

Section 8.4 ("Two-Dimensional Geometry") provides an introduction to the creation of 2D geometries. To build this 2D model, the user can create a new model by selecting *File* > *New* and following the methodology presented in the previous section to create the 1D fin. Another method is to build upon the 1D example by adding a new 2D component (Figure 9.6). The advantage of the latter technique is that it makes it easier to compare the 1D and 2D results in one plot (for the selected solution subset), which can be useful for more complex problems.

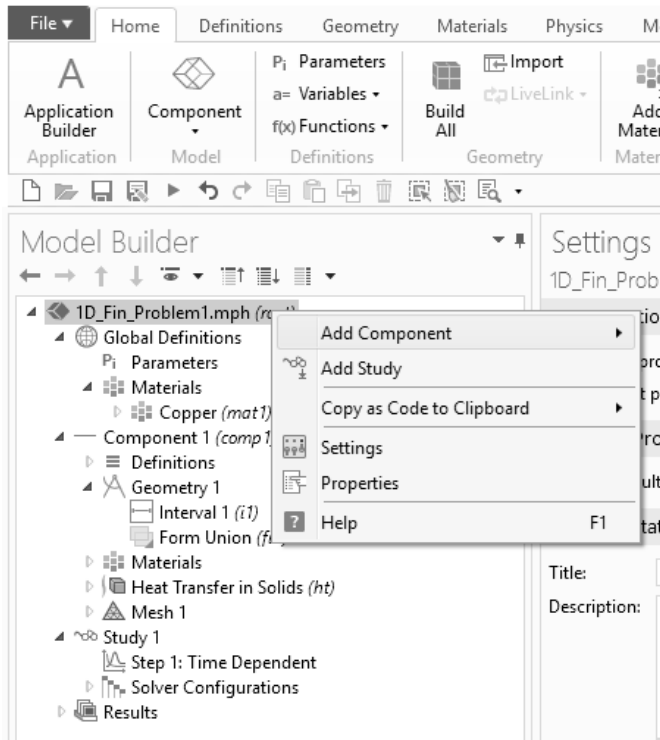


FIGURE 9.6. Adding a new component to the model tree.

The next step is to create the geometry. In this example, the geometry is created using Bezier polygon and rectangle commands, both are part of the core package of COMSOL Multiphysics with the parameters

identified in Figure 9.7. Figure 9.8 shows the settings used for the creation of the polygon and rectangle shapes. These shapes are added to the current component's geometry by right-clicking on its *Geometry* node and selecting the desired shape. The Bezier polygon is created from its segments that are connected to one another, forming open or closed loops. Any number of segments can be added, consisting of *x*- and *y*-coordinate data for each vertex (known as the control point). Segments of linear, quadratic, or cubic shapes are possible. Note that in each linear segment after the first one, the first of the two points listed is equal to the second point of the previous segment. Thus, for example, changing the coordinates of point 2 of the first segment will automatically assign the same values to the point 1 of the second segment. The *Rectangle* command settings include its width and height as well as the position of the rectangle center or corner. Rotation about the center or corner can be also set, positive being in the counterclockwise direction. As new entities are defined within the component's geometry node, their labels (names) are added automatically under the geometry tree (Figure 9.9). There are two names for each entity: (1) the long name is one that you can modify yourself (e.g., *Rectangle 2*), and (2) the short name is set by the system and is the one referenced when any operations are done on these entities (e.g., *r2*).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
L	0.1[m]	0.1 m	fin length
mesh_size	0.0021[m]	0.0021 m	mesh size
n	1	1	multiplier
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
premiter	(th1+th2)/2	0.01 m	average thickness throughout
rho	998.2071[kg/m^3]	998.21 kg/m³	density
rotation	0[deg]	0 rad	rotation angle
th1	0.01[m]	0.01 m	fin thickness, b
th2	0.01[m]	0.01 m	fin width

FIGURE 9.7. Parameters used to create 2D side-rectangular fin geometry (global level).

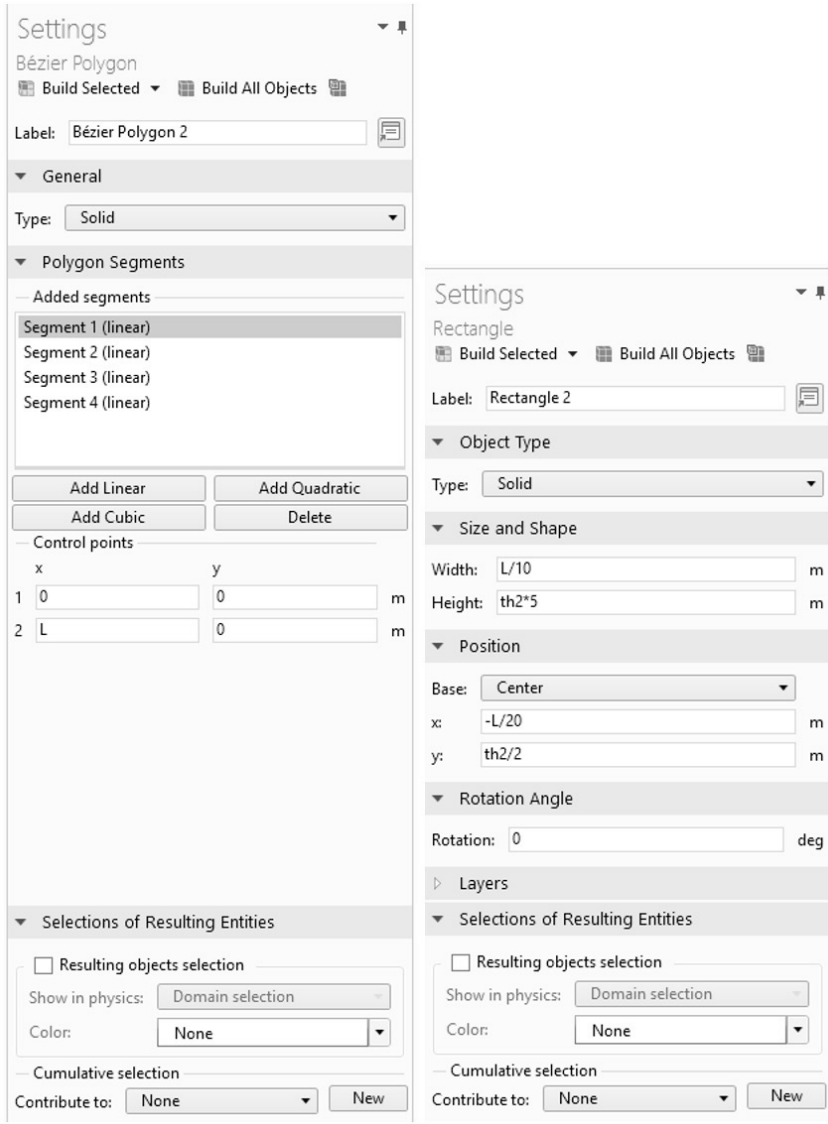


FIGURE 9.8. 2D side-rectangular fin: (a) Creating Bezier polygon (*b2*), (b) Creating rectangle (*r2*).

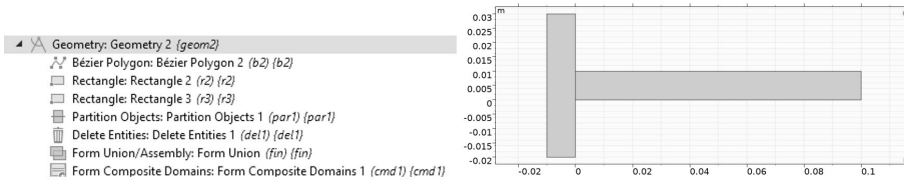


FIGURE 9.9. (a) Geometry building components for 2D side-rectangular fin (left), (b) 2D side-rectangular fin (right).

For this example, it is clear that the domains are symmetrical with respect to the horizontal line passing through the center of the part, parallel to the *x*-coordinate. Therefore, one could directly create the geometry of half of its actual size from the start. However, to demonstrate the use of partitioning, a rectangle (*r*3) is generated with one side that crosses the part along the horizontal symmetry axis. This rectangle (*r*3) is used as the partitioning domain for the actual geometry consisting of the Bezier polygon (*b*2, horizontal fin) and the rectangle (*r*2, vertical wall)—Figure 9.10a.

The highlighted regions in Figure 9.10b are then deleted after being selected in the *Delete Entities* node. Figure 9.11 shows the geometry after deletion, with a horizontal line visible passing through the wall domain; the line was generated by the partitioning. Although the user can work with this geometry in its *present* state to perform any types of analysis using the separate domains created by this line, separate material and mesh would need to be defined for each domain. If such separation is not needed, it is possible to either delete this extra line or combine the two regions using the *Form Composite Domains* feature under *Virtual Operations* in order to build a single domain. The latter step was performed in the example shown, with the resulting geometry taking advantage of the symmetry shown in Figure 9.12a. The meshed geometry is presented in Figure 9.12b. It shows that different mesh types were used for each region: triangular mesh for the wall and rectangular mesh, generated by the mapped mesh technique, for the fin.

As it was stated before, reducing the complexity and size of the geometry, without sacrificing any important information, will make your solution more efficient. When doing this, not only the geometry but also the boundary conditions and any loads must be symmetrical; otherwise, even though the geometry may be symmetrical, some boundary and initial conditions may be missed in certain regions of the geometry, resulting in the physics not being properly captured.

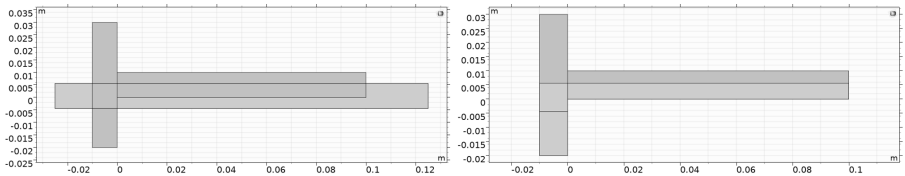


FIGURE 9.10. 2D side-rectangular fin: (a) Creating partitions, (b) Purple-highlighted entities to be deleted.

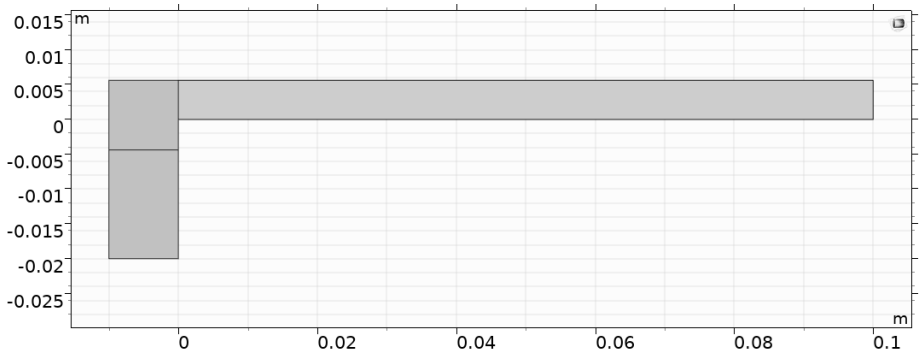


FIGURE 9.11. Creating composite domains for 2D fin with the same thickness throughout the length.

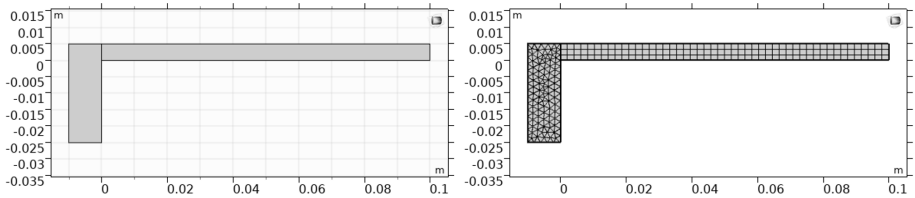


FIGURE 9.12. Creating 2D side-rectangular fin by using parameters: (a) Geometry (left), (b) Meshed geometry (right).

9.1.3 Three-Dimensional Fin Geometry

For the example presented herein, the 3D geometry is created using block entities based on the parameters defined in Figure 9.13. *Block* is part of COMSOL Multiphysics core module geometry modeling. The block size is defined by its three dimensions (width, depth, height), and its position is defined by locating either its corner or center, similar to the rectangle in 2D.

The *Axis* selection determines the direction along which the height dimension of the block will be oriented. The default is set to z-coordinate. The same direction will also serve as the axis about which the block will be rotated if a nonzero rotation angle is entered. The *Coordinate System* setting allows the user to choose another coordinate system within which to work for defining this block. For example, any local work plane coordinate system (xw, yw, zw) defined earlier may be used. Thus, if you have a smaller *Block B* which is attached to a larger *Block A* and then you rotate *Block A*, *Block B* would be reoriented appropriately if you used the coordinate system local to *Block A*.

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	(th1*th2)/2	5E-5 m^2	tip area
L	0.1[m]	0.1 m	fin Length
mesh_size1	0.003[m]	0.003 m	fin mesh size
mesh_size2	0.005[m]	0.005 m	vertical block fin size
n	1	1	multiplier
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
premiter	2*(th1+0.5*th2)	0.03 m	fin perimeter
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
th1	0.01[m]	0.01 m	fin thickness, b
th2	th1	0.01 m	fin width, a

FIGURE 9.13. Parameters used to create 3D side-rectangular fin with rectangular cross section geometry (global level).

Similar to the 2D analysis, one can use symmetry to simplify the solid geometry. In order to achieve this, a cutting plane (symmetry plane) parallel to the *x*-*z* plane is defined along the *y*-coordinate at the midpoint of the fin base (Figure 9.14). This is a work plane whose settings are presented in Figure 9.14. This work plane is defined by the *Plane* (*zx-plane*), *Offset type* (*Distance*), and *y-coordinate* (*th2/2*) settings. Figure 9.15 shows the components of the geometry block along with the highlighted region that is deleted to take advantage of symmetry. Figure 9.16 presents the block definition settings used for this example. They are similar to the 2D rectangle settings, except that the specifications related to the third dimension (along the *z*-coordinate) such as length and coordinate data of the reference point are also added. Figure 9.17a shows the reduced geometry taking advantage of the symmetry. Either of the volumes (to either side of the cutting plane) may be employed. For this example, the left portion of the solid body (away from the viewer) is selected for further analysis.

Meshed geometry is presented in Figure 9.17b, showing automatically generated tetrahedral mesh used for the wall and a mesh made up of regular brick-shaped elements built using mapped and swept mesh features. If there are constant boundary conditions at the interface (connecting areas) of the two solids such as constant temperature, heat

flux, or any other conditions that isolate the fin from the vertical wall, under careful consideration; the analyst may separate the physics and regions and treat them as individual domains.

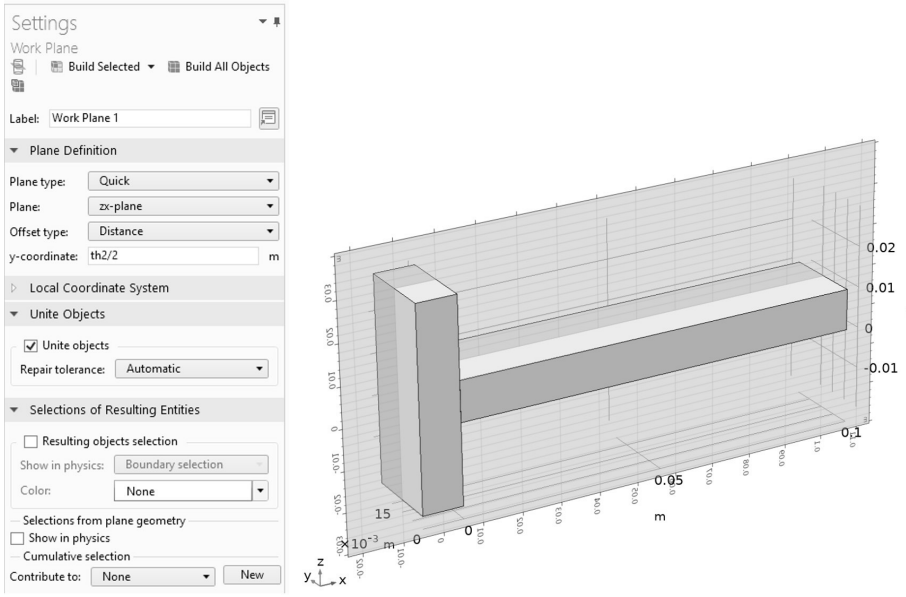


FIGURE 9.14. Creating work plane for 3D side-rectangular fin with rectangular cross section (*wp1, z-x* work plane, $y = th_2/2$).

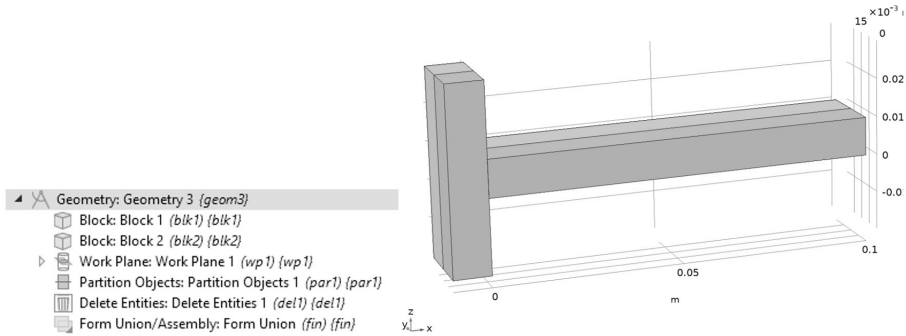


FIGURE 9.15. (a) Geometry building components for 3D side-rectangular fin with rectangular cross section (left), (b) Purple-highlighted entities to be deleted (right), (*wp1*, *z-x* work plane, $y = th_2/2$).

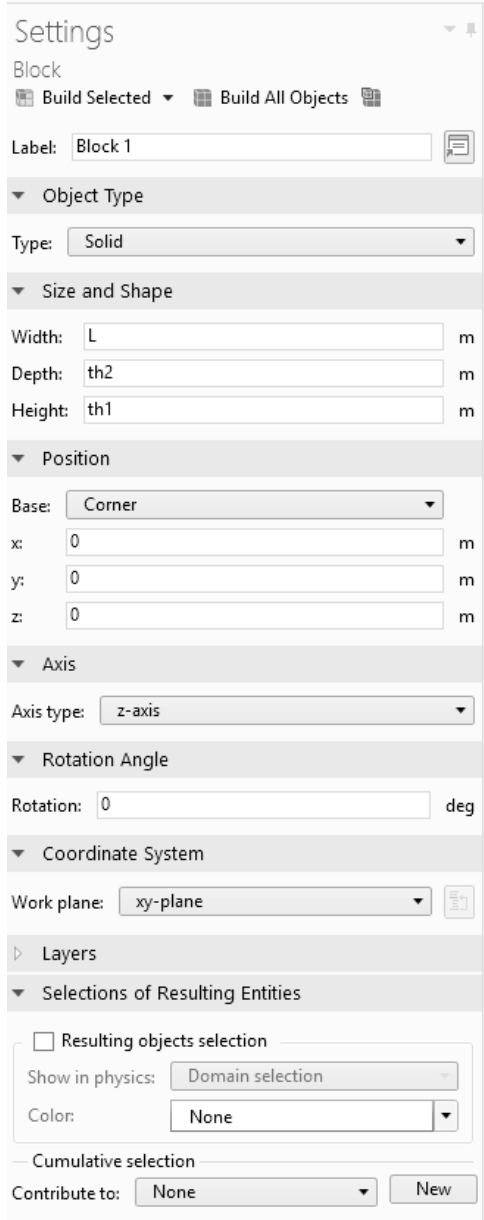


FIGURE 9.16. Creating block for 3D side-rectangular fin with rectangular cross section (*blk1*).

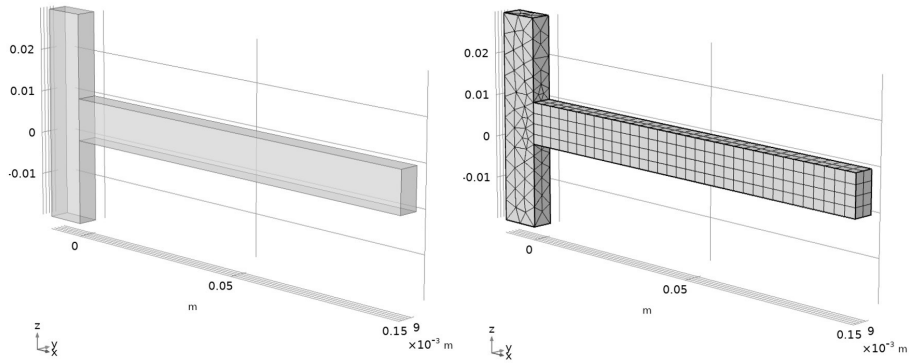


FIGURE 9.17. Creating 3D side-rectangular fin with rectangular cross section by using parameters: (a) Geometry (left), (b) Meshed geometry (right).

Recall that one way to verify that the calculations were done correctly when defining a formula using parameters is to use the *Measure* feature introduced earlier. You can select the entities such as areas (e.g., fin tip area highlighted in Figure 9.18 is 5×10^{-5} m^2) and perimeters (e.g., fin perimeter highlighted in Figure 9.19 is 0.03 m). Note that these values equal those calculated by formulae shown in Figure 9.13.

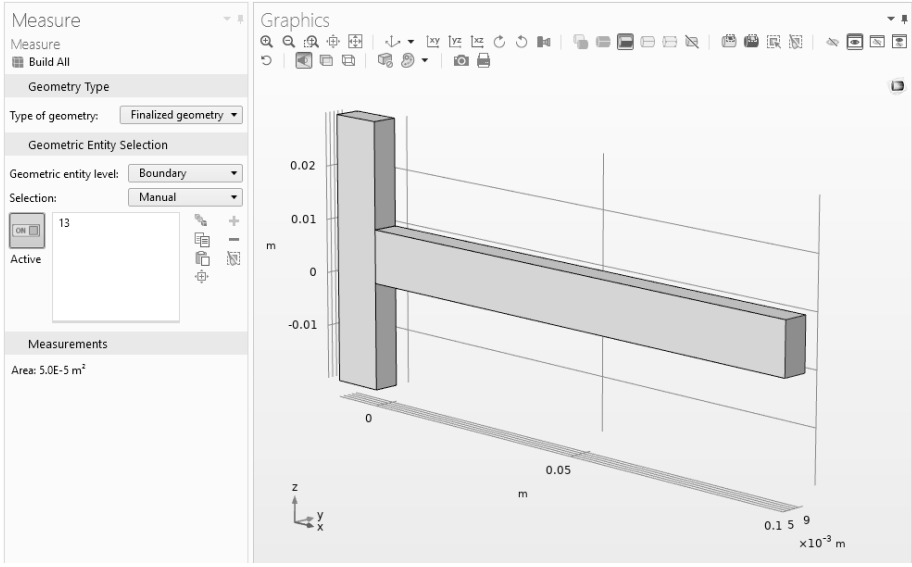


FIGURE 9.18. Measuring tip area for 3D side-rectangular fin with rectangular cross section.

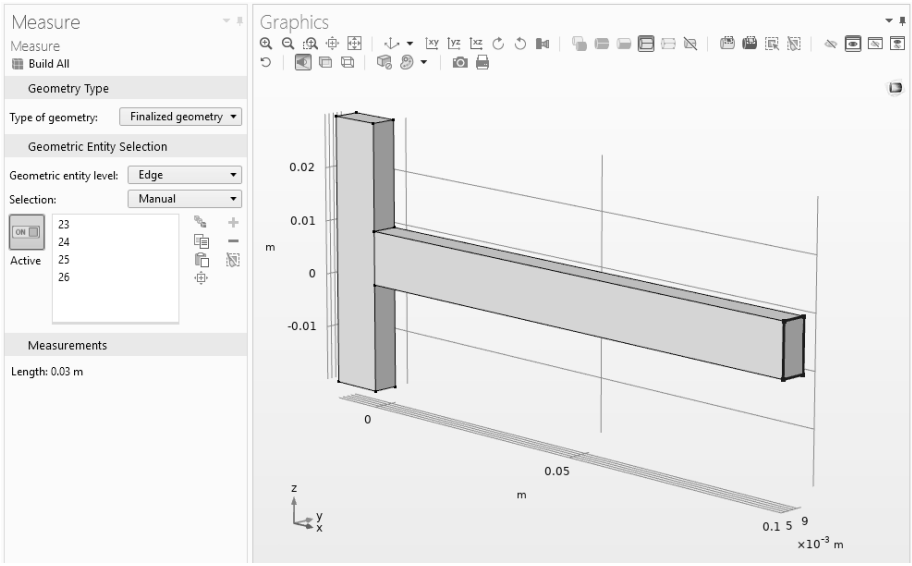


FIGURE 9.19. Measuring tip perimeter for 3D side-rectangular fin with rectangular cross section.

9.1.4 3D Fin with Central Channel

This example builds on the preceding one of the 3D rectangular fin by adding a rectangular channel of 2.5 mm height and 10 mm depth that passes through the fin's center. This channel may be used; for example, to transport a cooling liquid within a heat pipe. The channel geometry is added by a block (*blk5*). Note that in order to model fluid flow through the channel, a solid geometry needs to be created for it—one cannot mesh an empty space! After creating a geometry model for the channel, one can assign to it solid properties and boundary conditions such as flow or load.

The geometry is created using the parameters defined in Figure 9.20. Similar to the preceding example with no central channel, the symmetry plane can be used to simplify the solid geometry (Figure 9.21). In order to achieve this, a cutting plane parallel to the *x*-*z* plane is defined along the *y*-coordinate at the midpoint of the fin base—similar to what was presented in Figure 9.14. Half of the geometry to either side of the cutting plane may be employed. For this example, when facing the wall, the left portion of the solid body is selected for further analysis (Figure 9.22b). The meshed geometry is presented in Figure 9.23b. The fin and the wall are meshed with automatically generated tetrahedral elements. Also, where a central flow channel is added, additional elements forming layers around the channel boundary are created. This is to accurately model the fluid-solid interaction in the vicinity of the channel walls. When performing analysis, the user can introduce a thin layer of material between any two solid parts

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	(th1*th2)/16	1.25E-5 m^2	tip area
area_ch	(th1*th2)/16	1.25E-5 m^2	channel area
area_end	th1*th2*0.5	1E-4 m^2	fin end area
area_tip	area_end-area_ch	8.75E-5 m^2	fin tip area
channel_depth	th2/2	0.01 m	channel depth
channel_height	th1/4	0.0025 m	channel height
L	0.1[m]	0.1 m	fin's Length
mesh_size	0.05[m]	0.05 m	mesh size
n	1	1	multiplier
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
perimeter_ch	2*(th1+th2)/4	0.015 m	channel perimeter
perimeter_tip	premiter_end+perimeter_ch-0.25*th2	0.05 m	fin tip perimeter
premiter_end	(2*(th1+0.5*th2))	0.04 m	fin end perimeter
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
th1	0.01[m]	0.01 m	fin thickness, b
th2	0.02[m]	0.02 m	fin width, a

FIGURE 9.20. Parameters used to create 3D side-rectangular fin with central channel geometry (rectangular cross section) (global level).

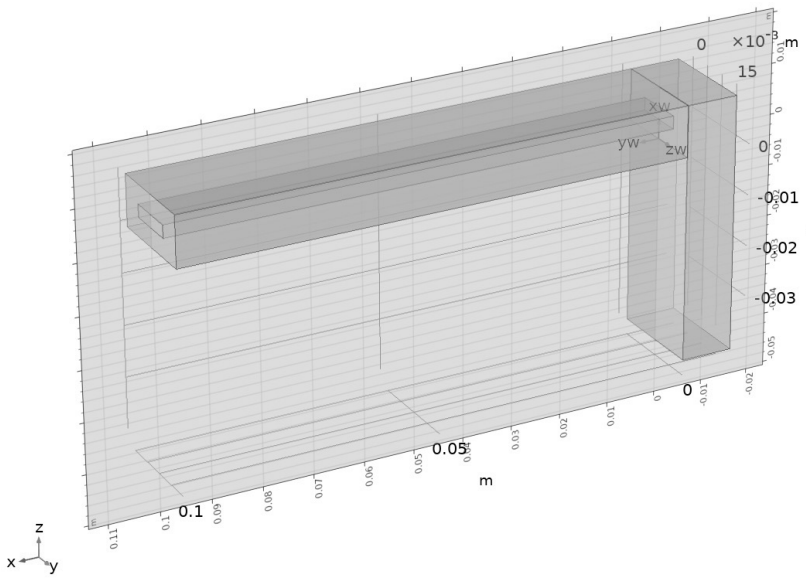


FIGURE 9.21. Creating work plane for 3D side-rectangular fin with central channel (rectangular cross section) (*wp1, z-x* work plane, $y = th_2/2$).

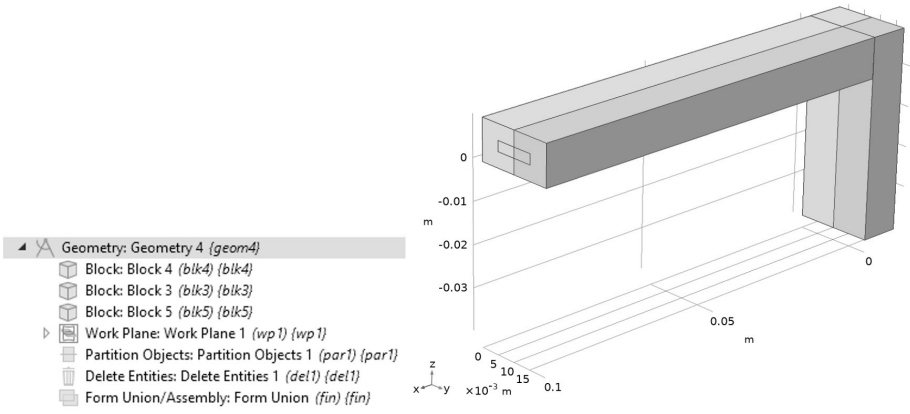


FIGURE 9.22. (a) Geometry building components for 3D side-rectangular fin with central channel (rectangular cross section) (left), (b) Purple-highlighted entities to be deleted (right), (*wp1*, *z-x* work plane, $y = th_2/2$).

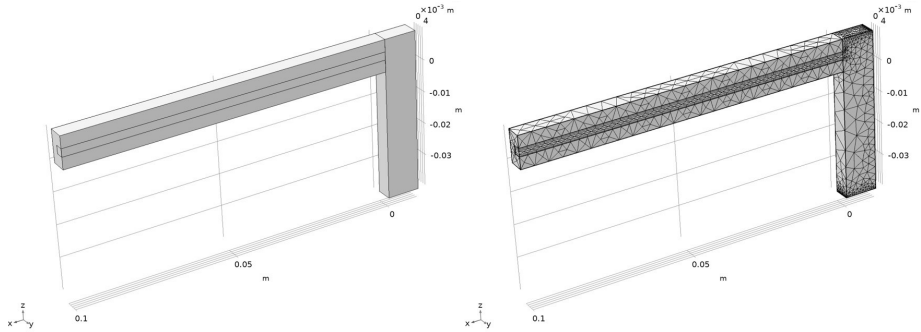


FIGURE 9.23. Creating 3D side-rectangular fin with central channel (rectangular cross section) by using parameters: (a) Geometry (left), (b) Meshed geometry (right).

in order to model thermal or mechanical contact. In this example, the thin layer can be the interface of the exterior duct and interior channel. This is particularly helpful if there is surface roughness at the interface or materials are different, causing a thermal gradient that is sharp enough to need accurate representation by introducing an intermediate surface.

Two measures are provided, one for area (the area of the C-shape fin tip excluding the central channel) and one for perimeter (the profile length or perimeter of the fin tip excluding the cooling channel). The two should be consistent with the ones measured on the model (Figure 9.24 and Figure 9.25).

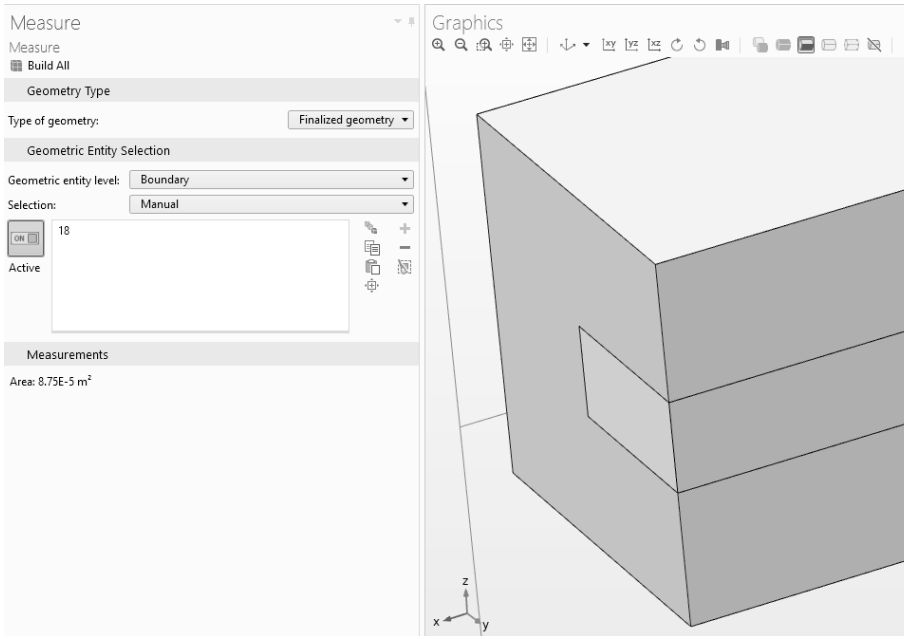


FIGURE 9.24. Measuring tip area for 3D side-rectangular fin with central channel (rectangular cross section).

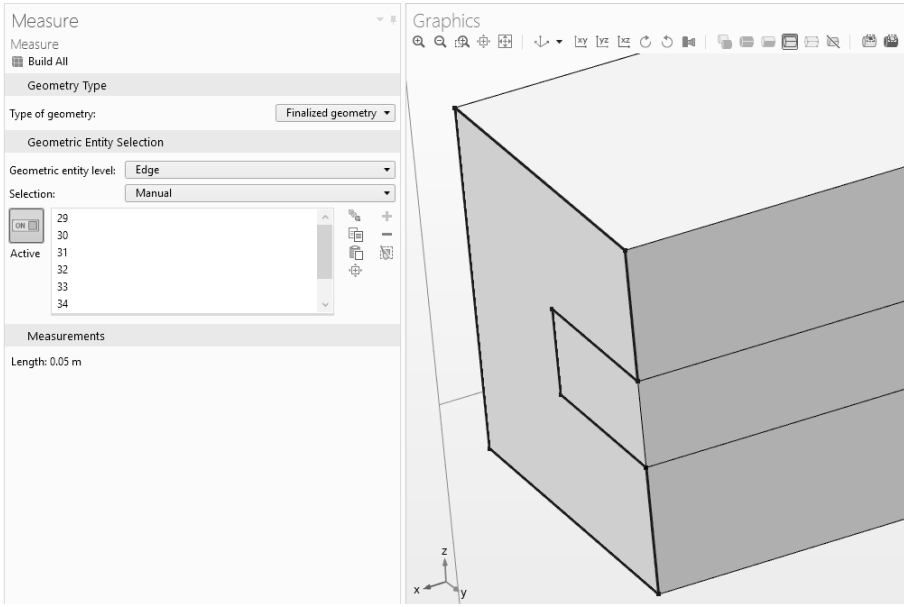


FIGURE 9.25. Measuring tip perimeter for 3D side-rectangular fin with central channel (rectangular cross section).

9.2 CYLINDRICAL FIN

This section focuses on another form of extended surfaces (or fins)—a cylinder (Figure 9.26). First, a basic design is presented. Next, a more complex design is introduced that embeds a rectangular cooling channel into the cylindrical fin. A cylindrical fin with finned cooling channel (which may be employed in aerospace applications) concludes the section.

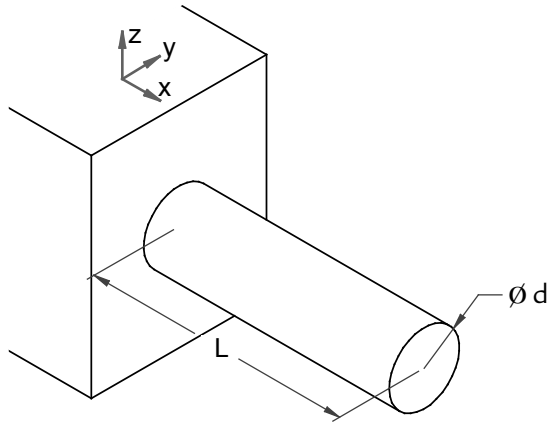


FIGURE 9.26. 3D Cylindrical fin geometry (drawings created in Solid Edge).

9.2.1 3D Cylindrical Fin with No Modifications

Steps taken to set up the 3D model are similar to those of creating the previous geometries. The user needs to create a new model or component on the upper tree, add a 3D component, define physics, and decide which analysis type should be performed. The user may either import the 3D geometry using a dedicated CAD tool or create it using the built-in geometry-creation features within the core COMSOL Multiphysics application. The geometry for this analysis is created using the latter approach with parameters shown in Figure 9.27. Figure 9.28 shows a block representing the surface and a cylinder of 10 mm diameter and 100 mm length comprising the model geometry. To take advantage of the symmetry, first a work plane is created, which is parallel to *x*-*z* plane and that passes through the model center at $y = th_1/2$ (Figure 9.28). This plane is then used to partition the model into two halves (Figure 9.29). Figure 9.30 shows the settings used to create the cylinder (*cyl1*) as part of the geometry construction sequence shown in Figure 9.29a. After partitioning for symmetry, the left half is kept, and the right half is discarded to create

the final geometry (Figure 9.31a). Mesh is shown in (Figure 9.31b). The FEM tool built-in measurement feature is employed to provide the fin tip area and perimeter to confirm correctness of the geometry dimensions. It is seen that the area (Figure 9.32a) and perimeter (Figure 9.32b) measurements are consistent with the calculated ones in the expressions shown in Figure 9.27.

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	0.5*3.1415*(th1/2)^2	3.9269E-5 m^2	fin tip area
L	0.1[m]	0.1 m	fin length
mesh_size	0.0025[m]	0.0025 m	mesh size
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
premiter	(0.5*3.1415*th1)+th1	0.025708 m	fin tip perimeter
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
th1	0.01[m]	0.01 m	fin diameter, d
wall_length	0.05[m]	0.05 m	wall length

FIGURE 9.27. Parameters used to create 3D cylindrical fin geometry (global level).

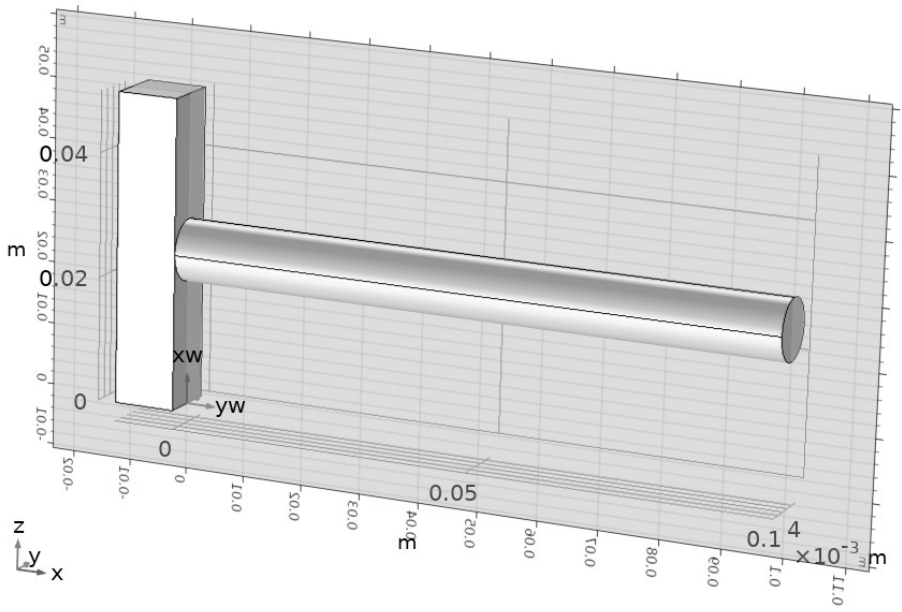


FIGURE 9.28. Work plane used to partition the 3D cylindrical fin (*wp1, z-x* work plane, $y = th_1/2$).

FIGURE 9.29. (a) Geometry building components for 3D cylindrical fin (left), (b) Purple-highlighted entities to be deleted (right), (*wp1*, *z-x* work plane, $y = th_r/2$).

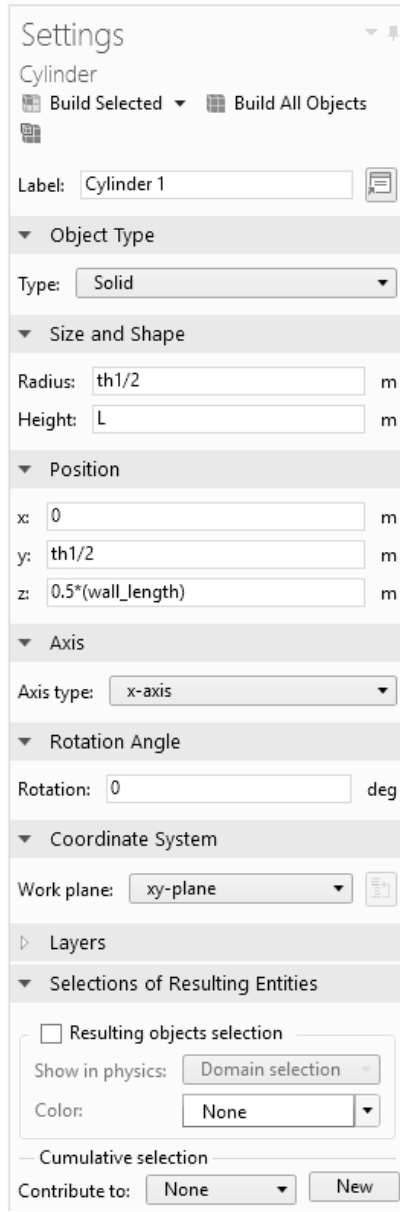


FIGURE 9.30. Cylinder creation settings for 3D cylindrical fin (*cyl1*).

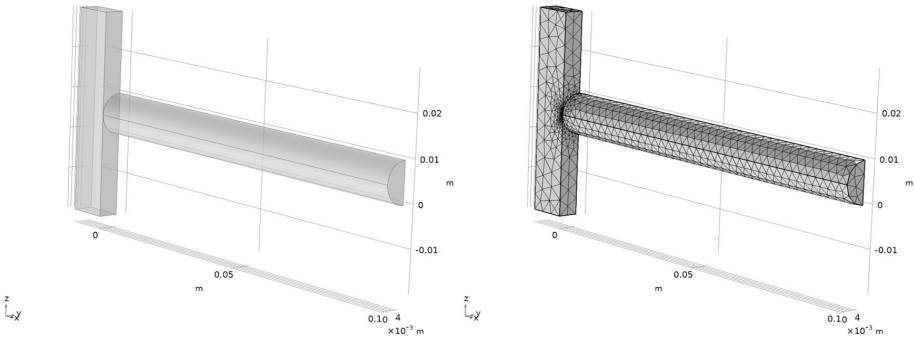


FIGURE 9.31. Creating 3D cylindrical fin: (a) Geometry (left), (b) Meshed geometry (right).

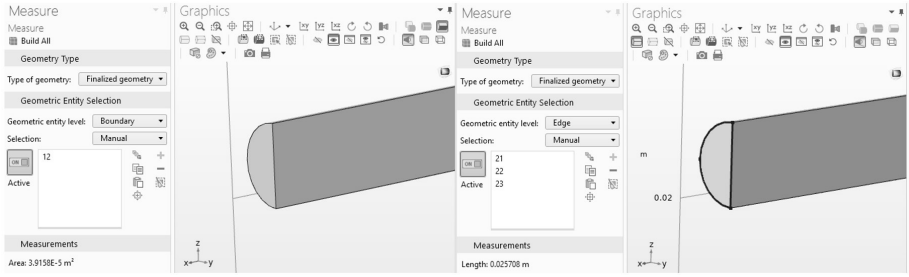


FIGURE 9.32. Measuring tip features for 3D cylindrical fin: (a) Fin tip area (left), (b) Fin tip perimeter (right).

9.2.2 3D Cylindrical Fin with Central Channel

Parameters to build this geometry are presented in Figure 9.33. The geometry for this analysis is similar to that of the cylindrical fin with an additional central rectangular channel (2.5×5 mm^2). Block creation settings (*blk5*) for the central channel are shown in Figure 9.34. A vertical work plane shown in Figure 9.35 is used to partition the geometry into two symmetrical halves. This vertical work plane is parallel to the *x*-*z* plane and passes through the model center at $y = th_1/2$ (Figure 9.35), and it partitions the geometry as shown in Figure 9.36. The partitioned domains are shown in the same figure as well with the highlighted regions to be deleted due to the symmetry. Figure 9.37 and Figure 9.38 present the finalized geometry and the mesh. Note that the mesh size is different for the vertical wall and horizontal fin embedding the central channel. The FEM tool built-in measurement feature is employed to

provide the fin tip area and perimeter for additional sources of confirmation (Figure 9.39). It is seen that the area (Figure 9.39a) and perimeter (Figure 9.39b) measurements approximately equal the values obtained via the expressions shown in Figure 9.33.

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	(0.5*3.1415*(th1/2)^2)-(0.25*th1)^2	3.3019E-5 m^2	fin tip area
channel_depth	th1/2	0.005 m	channel depth
channel_height	th1/4	0.0025 m	channel height
L	0.1[m]	0.1 m	fin length
mesh_size	0.0025[m]	0.0025 m	mesh size
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
premiter	0.5*(2*3.1415*th1/2)+(th1-0.25*th1)+3*0.25*th1	0.030708 m	fin tip perimeter
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
th1	0.01[m]	0.01 m	fin diameter, d
wall_length	0.05[m]	0.05 m	wall length

FIGURE 9.33. Parameters used to create 3D cylindrical fin with central channel geometry.

The central channel is represented as a solid part (separate volume) within the cylinder to which different properties can be applied. The central channel embedded in the geometry can be used as a cooling channel, and different fluids such as air, water, oil, and ethylene glycol be used as the cooling media. Different types of oils are used in cooling applications, for example, in aerospace and electronics fields. The oil type may even vary in one application depending on the lifecycle of the component; for example, the oil used for operating a new aircraft engine is a mineral oil, which is a replacement for the oil mixed with an anticorrosive agent when the engine was first manufactured and stored. The oil that is used after twenty-five to fifty hours of engine operation is an ash-less dispersant oil. Each have their own thermo-physical properties as well as dynamic viscosity that represents the dependency of the shear stress on the gradient of the velocity of the coolant moving tangentially with respect to the part to be cooled. Engine oil fulfills multiple functional needs: cooling, lubricating, sealing, and cleaning.

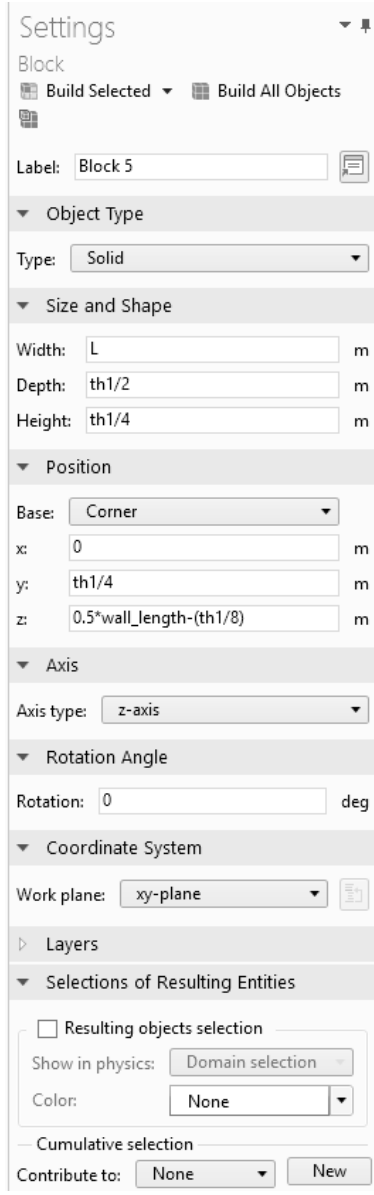


FIGURE 9.34. Creating central rectangular channel for 3D cylindrical fin with central channel (*blk5*).

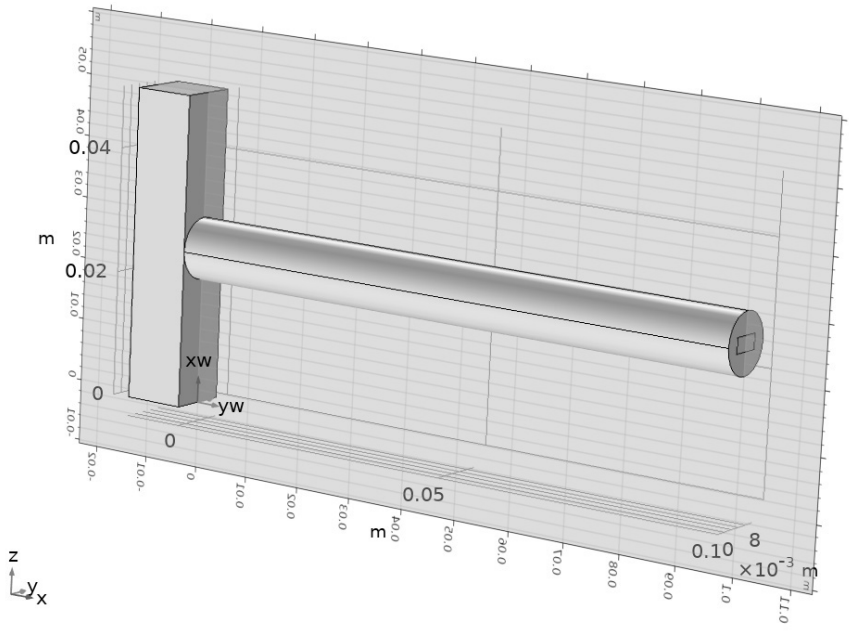


FIGURE 9.35. Work plane used to partition 3D cylindrical fin with rectangular central channel (*wp2*, *z-x* work plane, $y = th_1/2$).

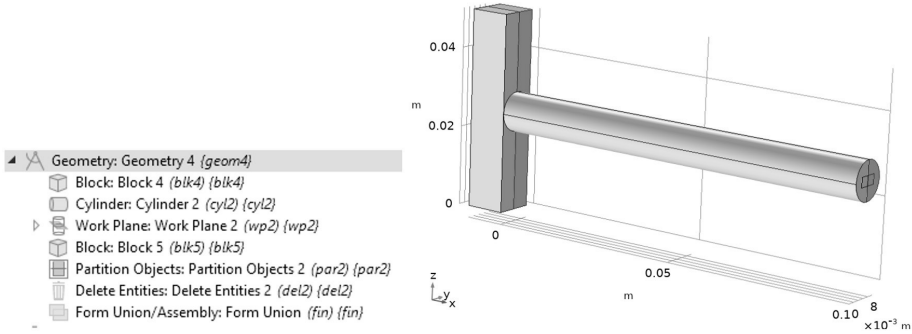


FIGURE 9.36. (a) Geometry building components for 3D cylindrical fin with rectangular central channel (left), (b) Purple-highlighted entities to be deleted (right), (*wp2*, *z-x* work plane, $y = th_1/2$).

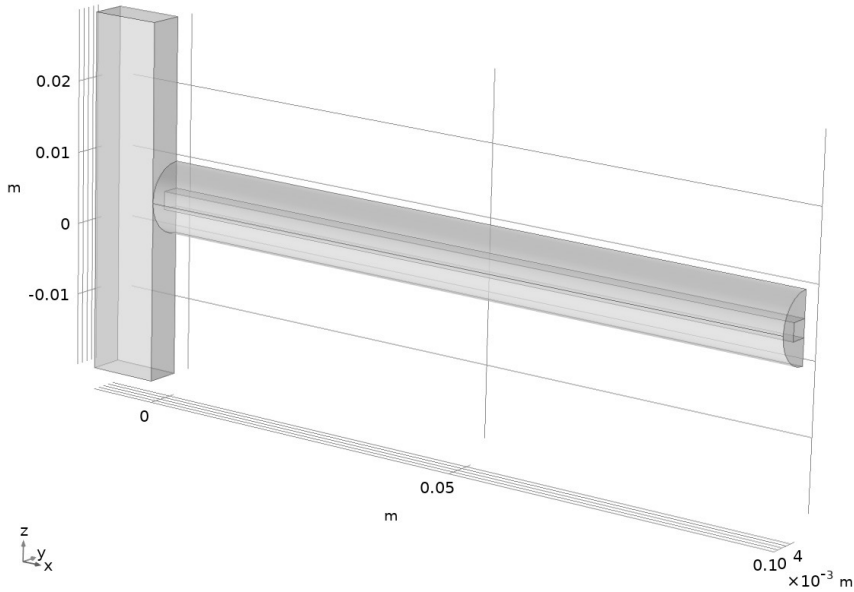


FIGURE 9.37. Creating geometry for 3D cylindrical fin with rectangular central channel.

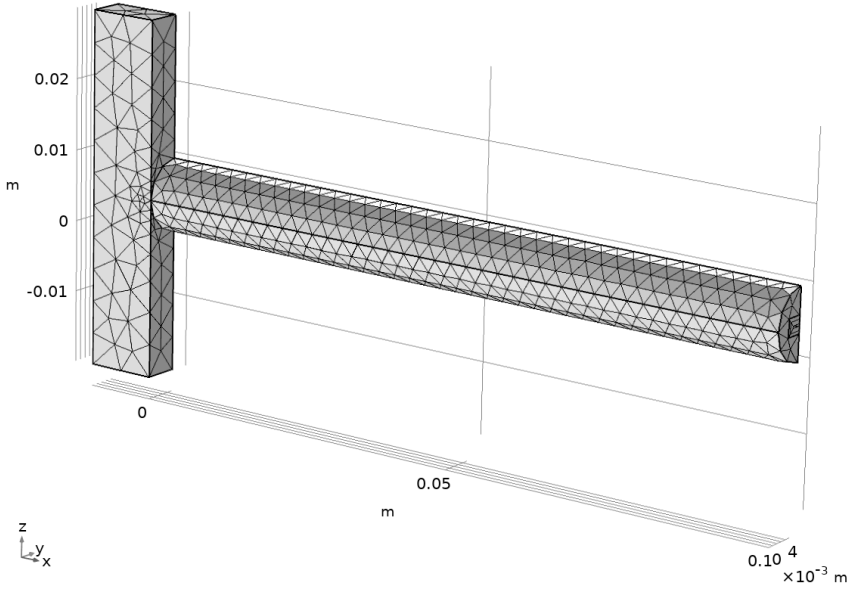


FIGURE 9.38. Creating mesh for 3D cylindrical fin with rectangular central channel.

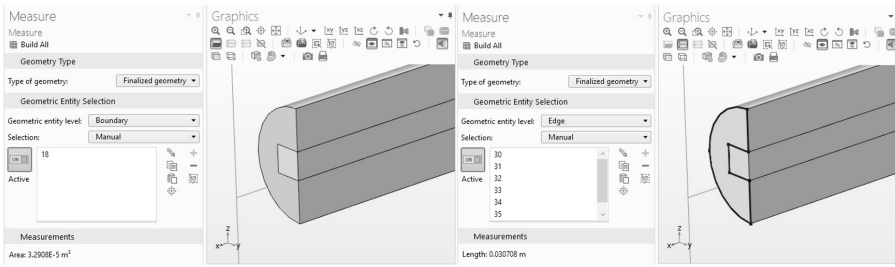


FIGURE 9.39. Measuring tip features for 3D cylindrical fin with rectangular central channel: (a) Fin tip area (left), (b) Fin tip perimeter (right).

9.2.3 3D Cylindrical Fin with Finned Central Channel

The following case study presents a modified heat pipe that consists of a cylindrical fin which embeds a finned central channel. Geometrical parameters are shown in Figure 9.40. The geometry for this analysis is cylindrical, as in the previous case, but with a round finned central channel (Figure 9.41). To create the rectangular grooves around the circular channel, the most efficient approach is to use a circular pattern CAD command (Figure 9.41). While the user is able to create rectangular patterns using COMSOL Multiphysics equipped with the geometry importer module, the circular pattern command is not available. Thus, the finned tube in this case study is created in Solid Edge (a mid-level CAD tool by Siemens PLM)—Figure 9.41. This example illustrates that, for complex geometries, a CAD tool may need to be employed. In addition to setting up the geometry, the model definition is completed by selecting the physics (e.g., *Conjugate Heat Transfer*), and the analysis type (e.g., *Time Dependent*).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
L	0.1[m]	0.1 m	fin length
mesh_size	0.05[m]	0.05 m	mesh size
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
rho	998.2071[kg/m^3]	998.21 kg/m³	density
th1	0.01[m]	0.01 m	fin diameter, d
wall_length	0.05[m]	0.05 m	wall length
z_displacement	0.005+0.5*(wall_length)-0.005	0.025 m	displacement in z direction

FIGURE 9.40. Parameters used to create 3D cylindrical fin with finned central channel geometry (global level).

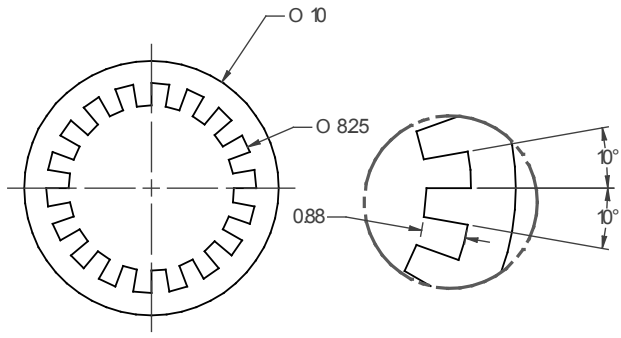


FIGURE 9.41. Dimensions of finned central channel cross section geometry (created in Solid Edge).

Note that when importing geometry into the model, you need to ensure that volumes are imported intact (meaning that the surfaces forming them are closed entities) so that they are distinguishable when setting up the components and assigning materials to them. The geometry is partitioned using the work plane presented in Figure 9.42. Figure 9.43 shows the geometry sequence as well as the geometry partitioned using the defined work plane.

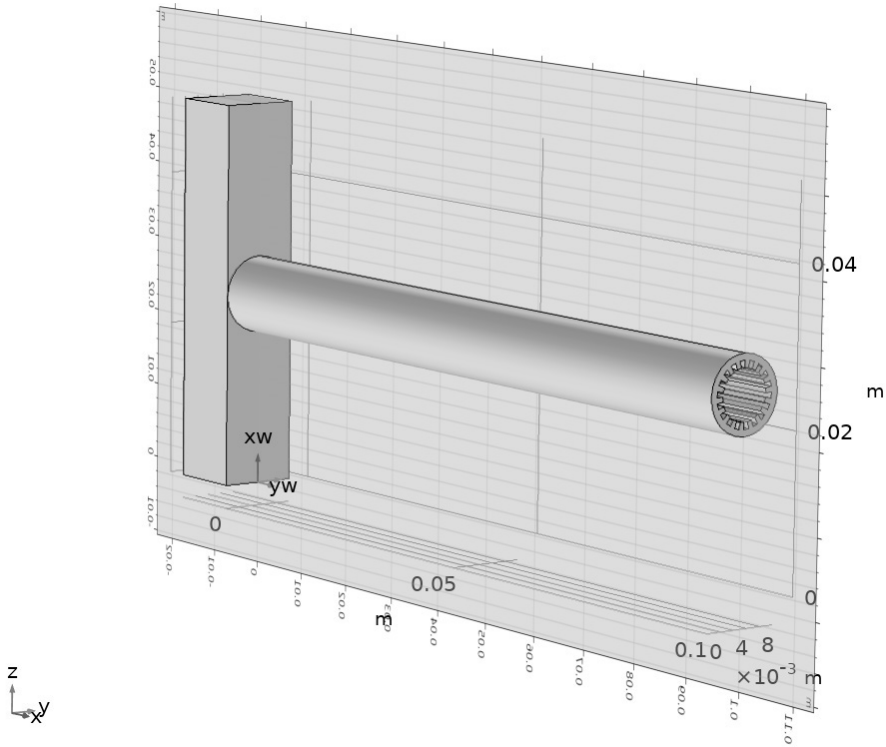


FIGURE 9.42. Work plane used to partition 3D cylindrical fin with finned central channel (*wp1, z-x* work plane, $y = th_1/2$).

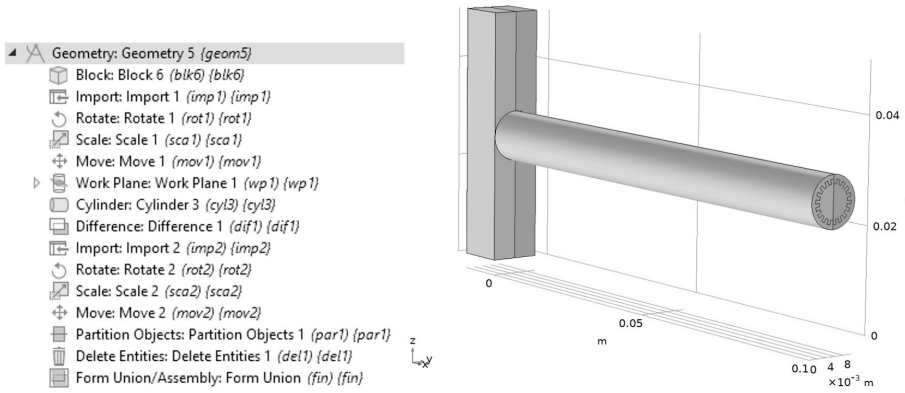


FIGURE 9.43. (a) Geometry building components for 3D cylindrical fin with finned central channel (left), (b) Purple-highlighted entities to be deleted (right), (*wp1, z-x* work plane, $y = th_1/2$).

The geometry sequence starts with the creation of the wall block (*blk6*). The hollow-finned cylinder component, created in Solid Edge (Figure 9.41), is imported into the program (*imp1*) in an orientation and location that does not align correctly with the model's coordinates, and its dimensions also need some adjustment. Therefore, after import, it is rotated by 90 degrees (*rot1*), scaled (*sca1*) with the *x*, *y*, and *z* scaling factors of 1, 0.5, and 0.5 to obtain a 100-mm length and 10-mm external diameter, and then moved (*mov1*) by means of a displacement vector (*x,y,z*) = (*0,0.005,0.0025*) m to connect it to the wall. After creation of the partitioning work plane (*wp1*), a new cylinder is created (*cyl3*) with the external diameter of 10 mm that coincides with the location of the imported fin. A difference volume (*dif1*) of this cylinder (*cyl3*) and the imported one (*imp1*) creates a volume that fills the hollow part of the imported fin. This newly created interior volume is then merged (while keeping the interior boundaries) with the imported hollow cylinder, and therefore two independent domains (interior and exterior) are created.

Figure 9.44 and Figure 9.45 present the finalized geometry and meshing. Note that the mesh distribution and size vary for the vertical wall and horizontal fin embedding the finned central channel. If a flow analysis is to be performed, the fin interior surface area in contact with the liquid flowing inside the channel will be meshed by multiple layers of boundary elements. The FEM tool built-in measurement feature is employed to provide the fin tip area and perimeter as an additional confirmation (Figure 9.40). It is seen that the area (Figure 9.46a) and perimeter (Figure 9.46b) measurements should be consistent with the associated calculated ones when using parametric studies. The reader is encouraged to calculate these two values.

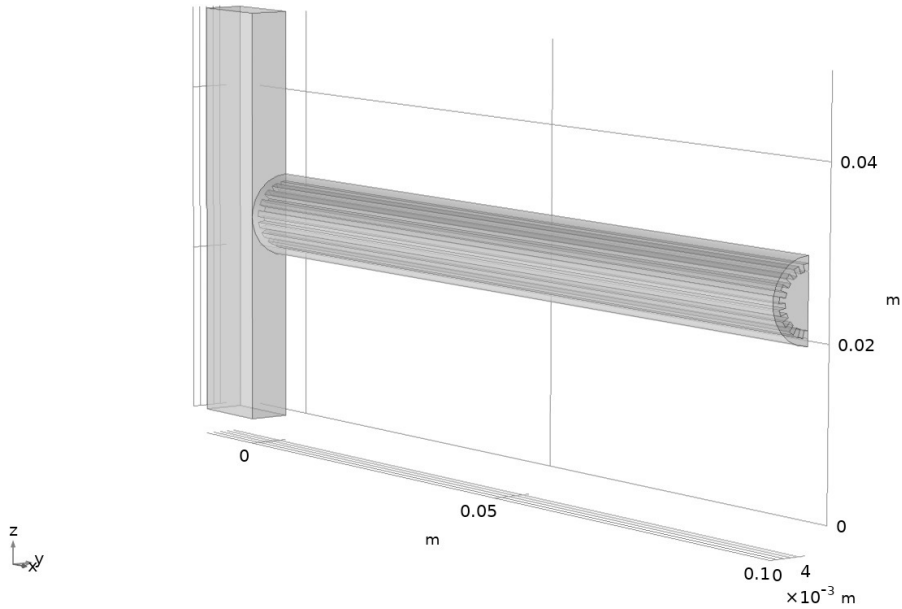


FIGURE 9.44. Creating geometry for 3D cylindrical fin with finned central channel.

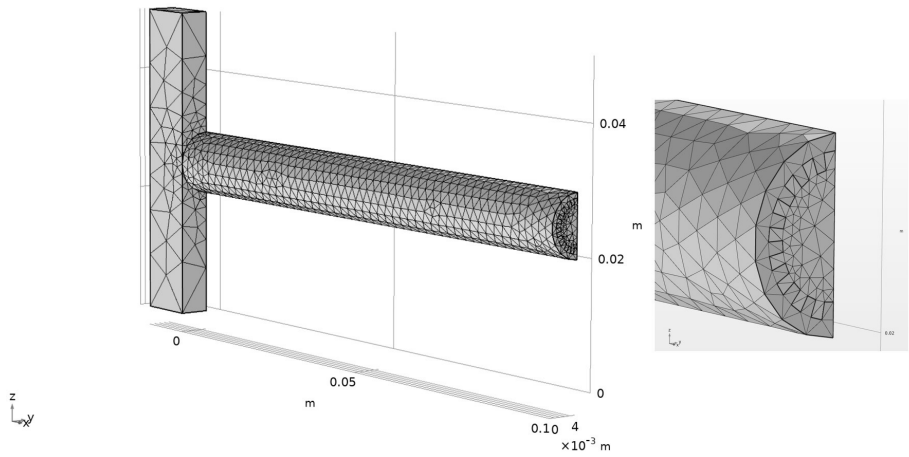


FIGURE 9.45. (a) Creating mesh for 3D cylindrical fin with finned central channel (left), (b) Mesh close-up (right).

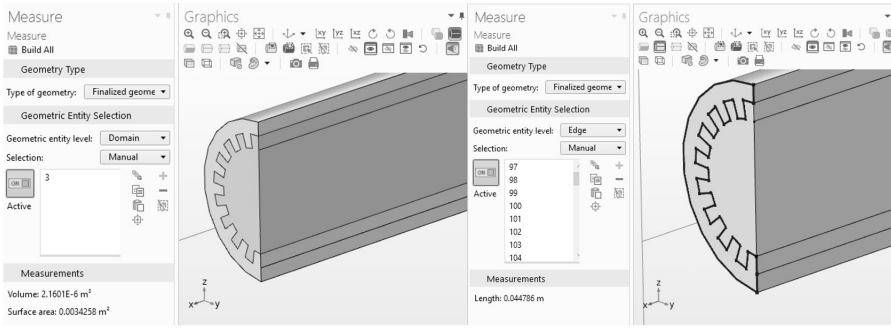


FIGURE 9.46. Measuring tip features for 3D cylindrical fin with finned central channel: (a) Fin tip area (left), (b) Fin tip perimeter (right).

9.3 SIDE-RECTANGULAR FIN WITH TRIANGULAR CROSS SECTION

In this model, a fin with a side-rectangular longitudinal shape and a triangular cross-section profile is created (Figure 9.47). The 3D geometry is made of a base feature (wall) to which the fin is perpendicularly attached.

Dimensions are provided for this example as parameters L, a, and b, so that the reader may change them as needed (Figure 9.48). The geometry is created in COMSOL Multiphysics core application. One way to create this geometry is to: (a) create a block; (b) create a work plane, starting with an x-z work plane and rotating it 45 degrees about the x-coordinate; and (c) partition the block volume and delete the top half (*Delete Entities*). Another method to create the work plane is to select the *Through vertex* plane type and then specify three points on the block (*blk6*) geometry through which the plane must pass (Figure 9.49). This method is employed when creating this part. This results in the inclined work plane shown in Figure 9.50. Figure 9.51 shows the geometry tree and the part partitioned by the work plane before the highlighted portion is deleted. An alternative approach to creating this fin is to draw a triangular profile on the y-z work plane and then extrude it along the x-coordinate by the length of the fin. The geometry and the mesh are shown in Figure 9.52. The FEM tool built-in measurement feature is employed to provide the fin tip area and perimeter for additional confirmation (Figure 9.53). It is seen that the area (Figure 9.53a) and perimeter (Figure 9.53b) measurements are consistent with the calculated values shown in Figure 9.48.

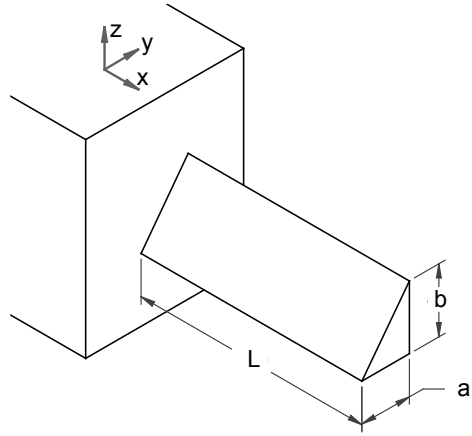


FIGURE 9.47. 3D rectangular fin with triangular cross section geometry (drawings created in Solid Edge).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	(th1*th2)/2	5E-5 m^2	fin tip area
L	0.1[m]	0.1 m	fin length
mesh_size	0.002[m]	0.002 m	mesh size
n	1	1	multiplier
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
premiter	((th1*th1+th2*th2)^0.5)+th1+th2	0.034142 m	fin tip perimeter
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
th1	0.01[m]	0.01 m	fin thickness, a
th2	0.01[m]	0.01 m	fin width, b

FIGURE 9.48. Parameters used to create 3D side-rectangular fin with triangular cross section geometry (global level).

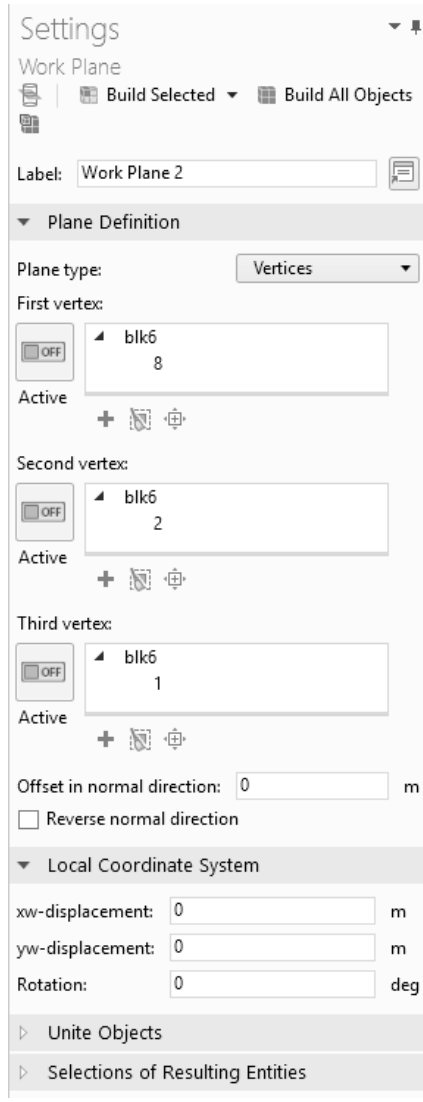


FIGURE 9.49. Creating work plane using vertices (*wp2*).

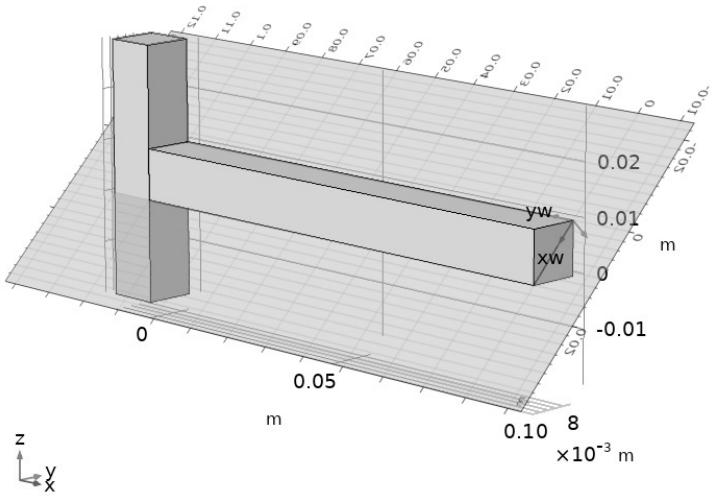


FIGURE 9.50. Creating work plane for 3D side-rectangular fin with triangular cross section (*wp2,* vertex work plane).

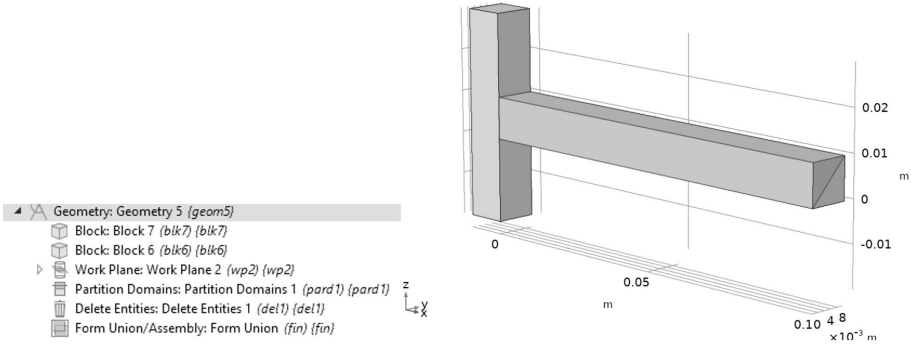


FIGURE 9.51. (a) Geometry building components for 3D side-rectangular fin with triangular cross section (left), (b) Purple-highlighted entities to be deleted (right), (*wp2*, vertex work plane).

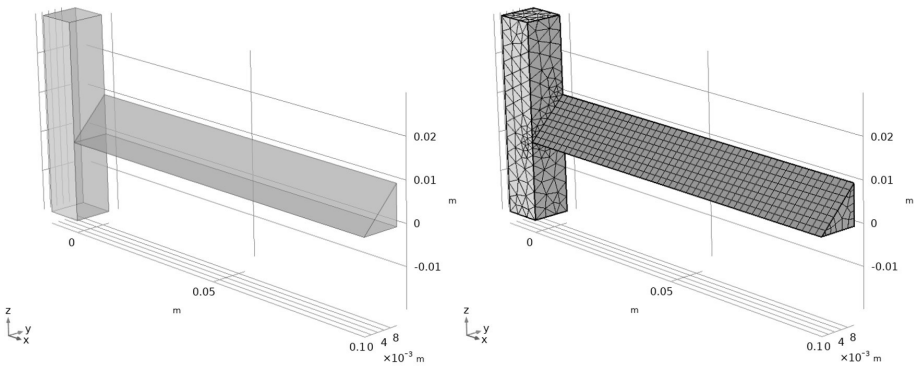


FIGURE 9.52. Creating 3D side-rectangular fin with triangular cross section: (a) Geometry (left), (b) Meshed geometry (right).

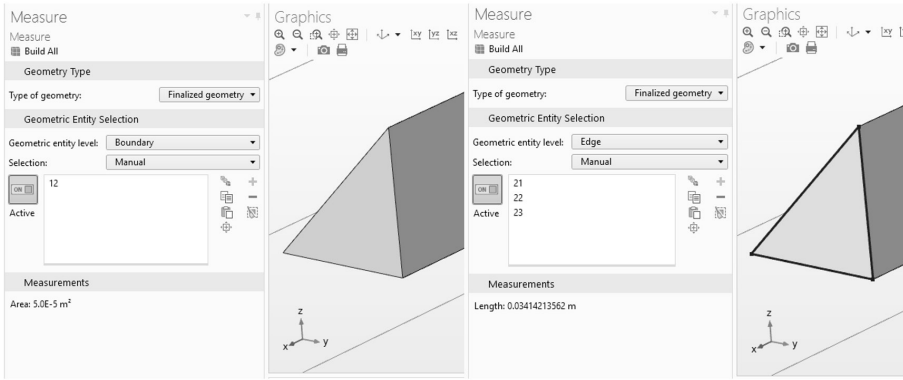


FIGURE 9.53. Measuring fin tip area for 3D side-rectangular fin with triangular cross section: (a) Fin tip area (left), (b) Fin tip perimeter (right).

9.4 SIDE-TRIANGULAR FIN WITH RECTANGULAR CROSS SECTION

In this example, instead of having a constant rectangular cross section along the fin length, the cross-section height is linearly reduced, reaching zero at the tip. This produces the side-triangular profile fin with rectangular cross section. This geometry is created in COMSOL Multiphysics core application tools (Figure 9.54). Parameters to create the geometry are presented in Figure 9.55. Two approaches can be used to create this geometry: extrusion of the sketched profile or partitioning with work planes. In the former approach, a triangular profile is drawn on the x-z work plane and then extruded along the y-coordinate by the depth of the fin. In the second approach, a work plane is created using *Through vertex*. It starts with the x-y work plane, which is then rotated by 22.5 degrees about the y-coordinate. Another work plane is then created using the same method and then rotated by −22.5 degrees about the y-coordinate. An alternative method to generate the work plane, which was used to create this part, is to use the *Through vertex* type and then select three points which are located on the desired plane—a point at the triangle tip and two points each at the upper and lower corners for each work plane, where the base meets the vertical wall (Figure 9.56). This step is repeated to create the second work plane. The triangle tip points can be created in a separate command (*pt1* and *pt2* in Figure 9.57). The volume is then partitioned using the defined work planes, and the top and bottom halves of the 3D fin are deleted. Figure 9.57 presents the geometry tree and the part including the partitioned domains that are deleted, highlighted in purple. Figure 9.58 presents the geometry and

the selected mesh. A combination of tetrahedral elements (vertical wall) and hexahedral elements (horizontal fin) are used to mesh this geometry. The mesh distribution varies with size and shape (tetrahedron versus cuboid). The FEM tool built-in measurement feature is employed to provide the fin top surface area and perimeter to confirm the methodology. It is seen that the area (Figure 9.59) and perimeter (Figure 9.60) measurements are consistent with the calculated values shown in Figure 9.55.

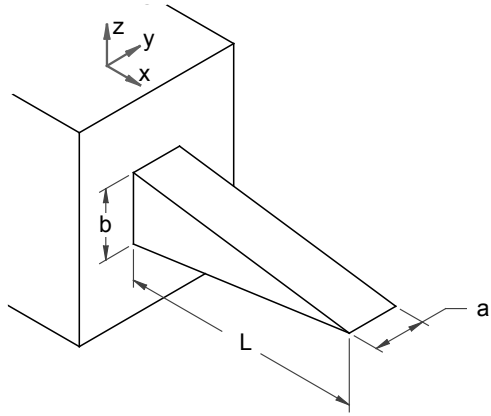


FIGURE 9.54. 3D side-triangular fin with rectangular cross section geometry (drawings created in Solid Edge).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	th1*L	0.001 m²	top surface area
L	0.1[m]	0.1 m	fin length
mesh_size	0.0025[m]	0.0025 m	mesh size
n	1	1	multiplier
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
premiter	2*(th1+L)	0.22 m	top surface perimeter
rho	998.2071[kg/m^3]	998.21 kg/m³	density
th1	0.01[m]	0.01 m	fin thickness, b
th2	th1	0.01 m	fin depth, a

FIGURE 9.55. Parameters used to create 3D side-triangular fin with rectangular cross section geometry (global level).

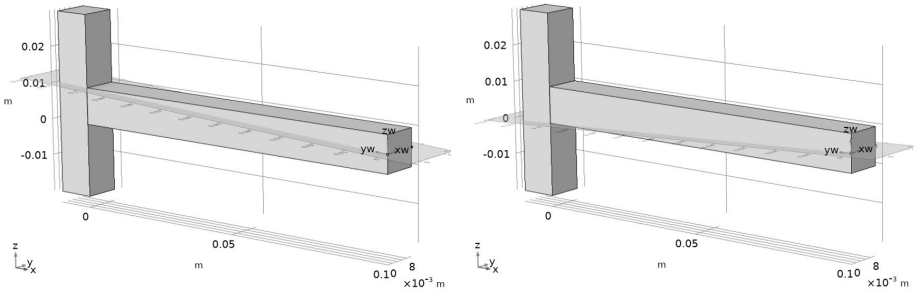


FIGURE 9.56. Creating work planes for 3D side-triangular fin with rectangular cross section using points: (a) *wp3* work plane (left), (b) *wp4* work plane (right).

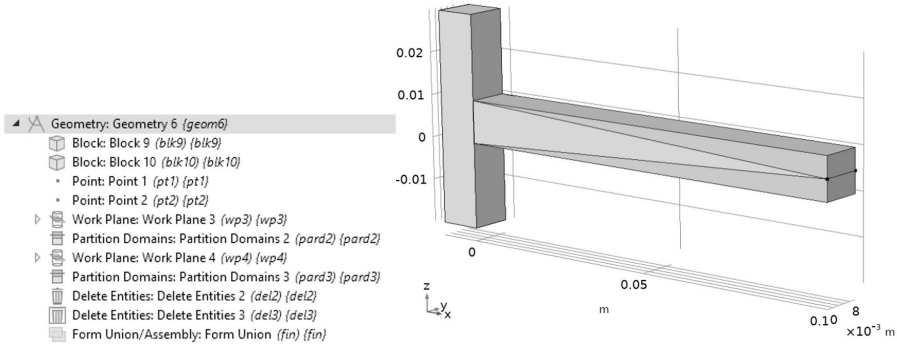


FIGURE 9.57. (a) Geometry building components for 3D side-rectangular fin with rectangular cross section (left), (b) Purple-highlighted entities to be deleted (right), (*wp3* and *wp4*, point work planes).

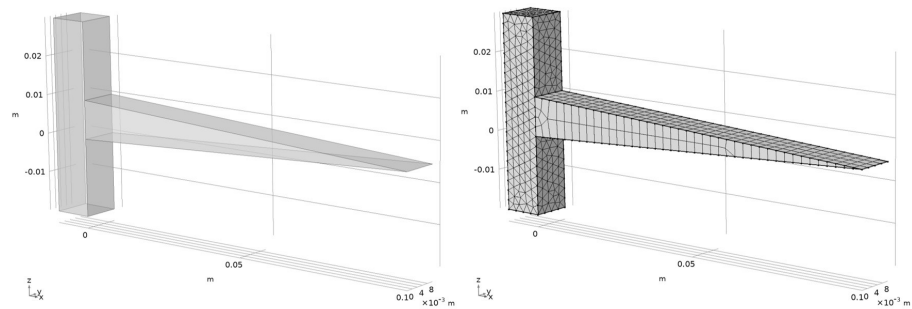


FIGURE 9.58. Creating 3D side-triangular fin with rectangular cross section: (a) Geometry (left), (b) Meshed geometry (right).

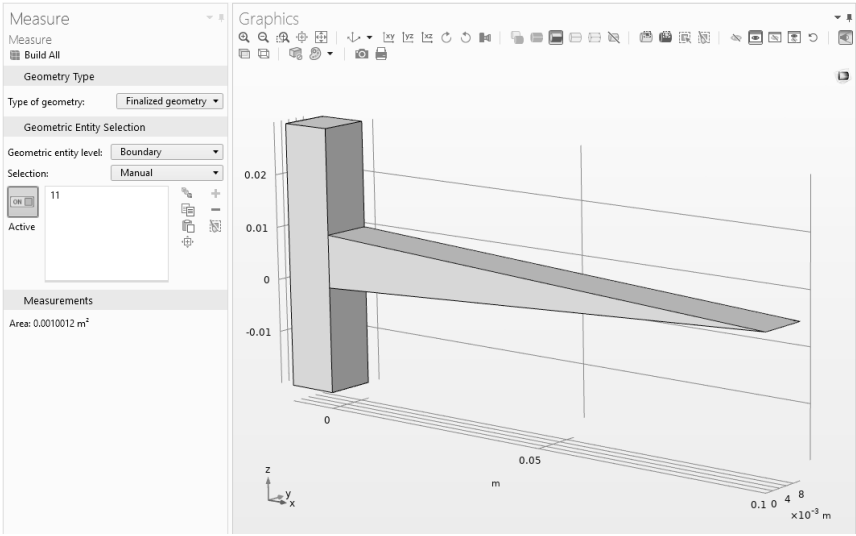


FIGURE 9.59. Measuring fin top surface area for 3D side-triangular fin with rectangular cross section.

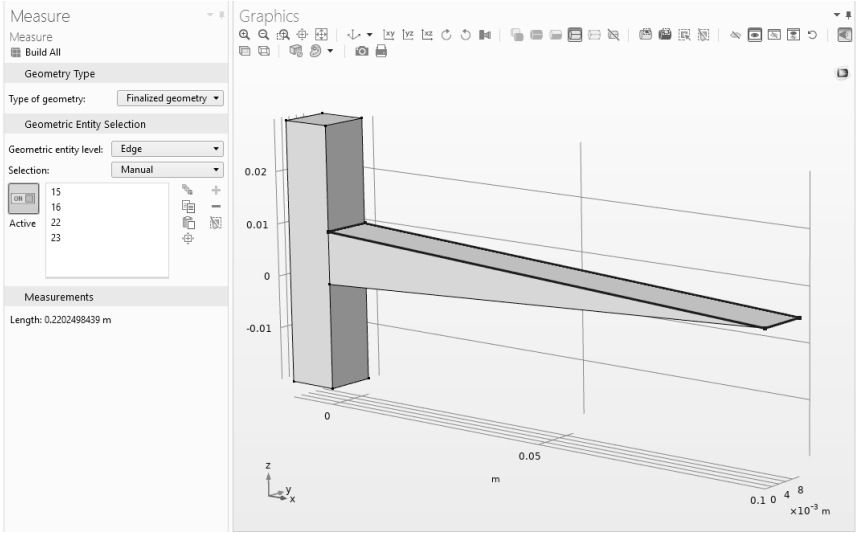


FIGURE 9.60. Measuring fin top surface perimeter for 3D side-triangular fin with rectangular cross section.

9.5 SIDE-CONCAVE FIN WITH RECTANGULAR CROSS SECTION

This fin model replaces the linearly changing cross-section height of the previous model with one that tapers off non-linearly to a zero height at the tip, with a concave curvature. Thus, a 3D side-concave fin with

a rectangular cross section is produced in this study (Figure 9.61). The main challenge in creating this geometry is the definition of the curved surface. Even though this shape is relatively complex, it could still be created in COMSOL Multiphysics CAD import tool. Parameters used to create this geometry are shown in Figure 9.62. The geometry is built by first creating two blocks, one for the wall and another as the base shape for the fin. Parametric surfaces are then defined that correspond to upper and lower concave surfaces; these surfaces are used as partitioning work planes (Figure 9.63 and Figure 9.64). The concave quadratic curves used are defined by the equation: $z = \pm(delta0/2) \times \left(x/bb\right)^2$, where the parameters *delta0* and *bb* are defined in Figure 9.62. The upper and lower solid entities remaining after the partition step are deleted (Figure 9.65). The geometry components along with the highlighted regions, which are to be deleted, are presented in Figure 9.65. Geometry and mesh are shown in Figure 9.66. The horizontal part (fin) is meshed with hexahedral elements while the vertical part (wall) is meshed with tetrahedral ones (Figure 9.66b). The FEM tool integrated measurement feature is employed to provide the fin top surface area and perimeter to compare the data estimated by the tool and that of the calculations (Figure 9.67 and Figure 9.68). Note that the area of a rectangle has been calculated as an approximation while neglecting the curvature. In this case, approximation is close as the curvature is small. The Author recommends the reader to employ geometrical parameters presented in Figure 9.62 and calculate the associated area and perimeter, comparing them with Figure 9.67 and Figure 9.68, respectively.

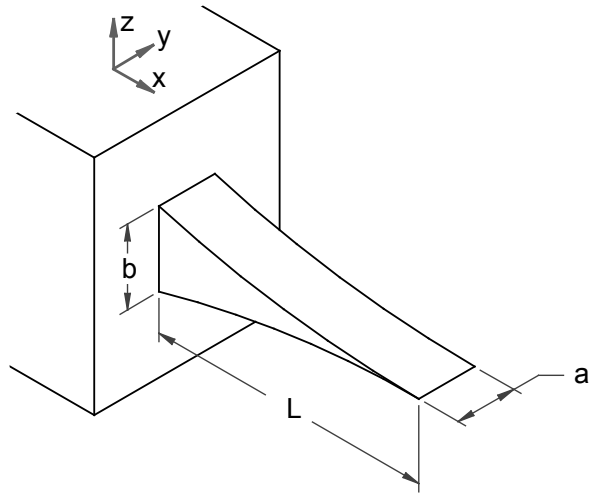


FIGURE 9.61. 3D side-concave fin with rectangular cross section geometry (drawings created in Solid Edge).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	th2*L	5E-4 m^2	fin top surfcae area
bb	1[m]	1 m	coefficient in ellipse
delta0	1[m]	1 m	coefficient in ellipse
L	0.1[m]	0.1 m	fin length
mesh_size	0.01[m]	0.01 m	mesh size
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
perimeter	2*(th2+L)	0.21 m	fin top surfcae perimeter
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
th1	0.01[m]	0.01 m	fin thickness, b
th2	th1/2	0.005 m	fin width, a

FIGURE 9.62. Parameters used to create 3D side-concave fin with rectangular cross section geometry (global level).

9.6 SIDE-CONVEX FIN WITH RECTANGULAR CROSS SECTION

In this fin model, the nonlinear variation of the rectangular cross-section height creates a convex curvature of the top and bottom fin surfaces to produce the 3D side-convex fin with rectangular cross section (Figure 9.69). The geometry for this analysis is similar to that of the side-concave fin with rectangular cross section presented in Figure 9.61, and so it was also created in COMSOL Multiphysics CAD import tool.

Geometrical parameters to create this geometry are shown in Figure 9.70. Following similar steps to the previous fin model, the 3D geometry is created by first defining blocks to represent the elongated rectangular fin and the wall and then defining parametric surfaces used to partition the rectangular fin block (Figure 9.71 and Figure 9.72). The difference here lies in the formula used to define the curvature in the *Expressions z* field of the parametric surface definition. The convex curves are defined by: $z = \pm(delta01/2) \times \left(x/bb1\right)^{1/2}$, where the values for *delta01* and *bb1* are given in Figure 9.70. The geometry components and subcomponents and highlighted domains, which designate entities selected for deletion, are presented in Figure 9.73.

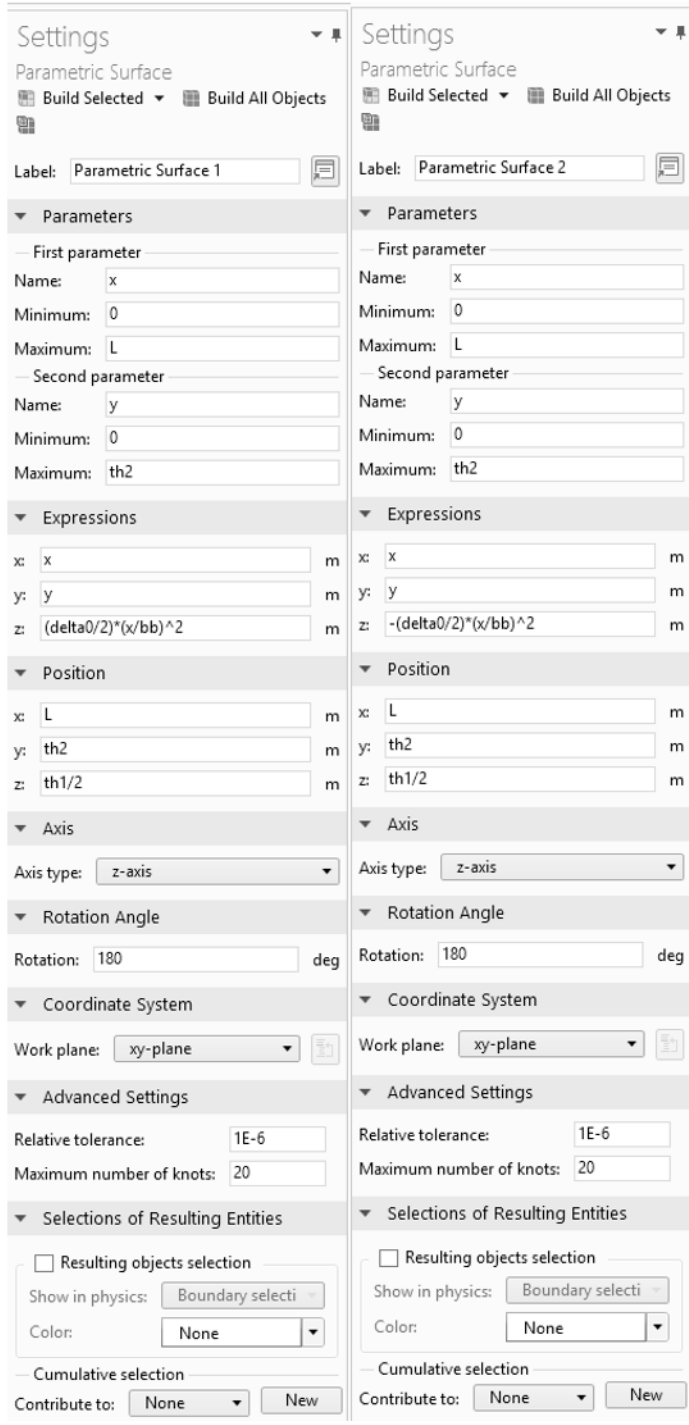


FIGURE 9.63. Defining parametric surfaces for 3D side-concave fin with rectangular cross section (*x-y* work plane, along *z*-coordinate): (a) *ps1* work plane (left), (b) *ps2* work plane (right).

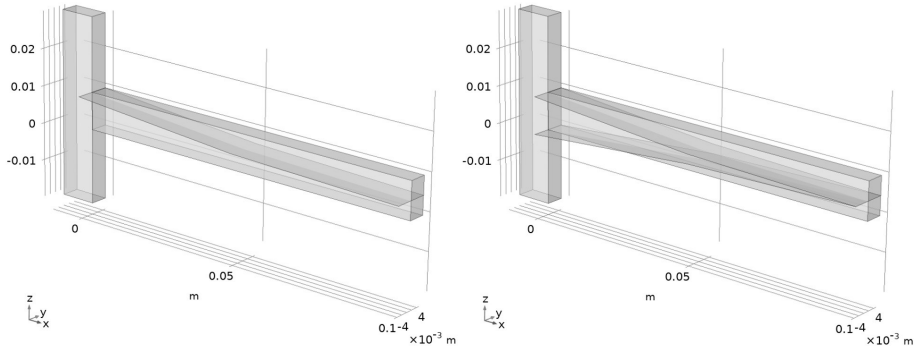


FIGURE 9.64. Creating parametric surfaces for 3D side-concave fin with rectangular cross section (*x-y* work plane, along *z*-coordinate): (a) *ps1* work plane (left), (b) *ps2* work plane (right).

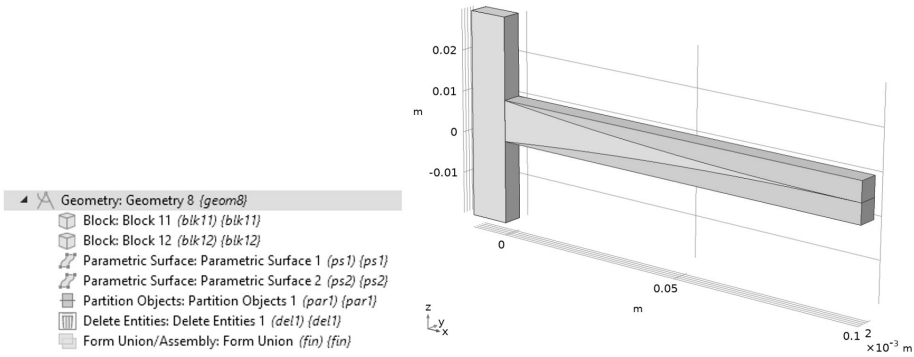


FIGURE 9.65. (a) Geometry tree for 3D side-concave fin with rectangular cross section (left), (b) Purple-highlighted entities to be deleted (right) (*ps1* and *ps2* through *x-y* work plane, along *z*-coordinate).

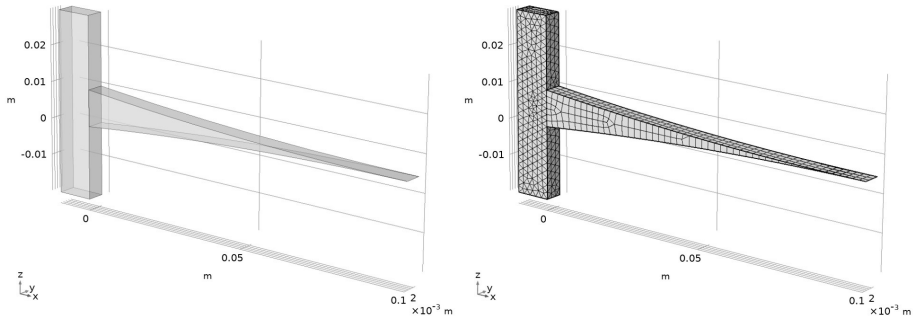


FIGURE 9.66. Creating 3D side-concave fin with rectangular cross section: (a) Geometry (left), (b) Meshed geometry (right).

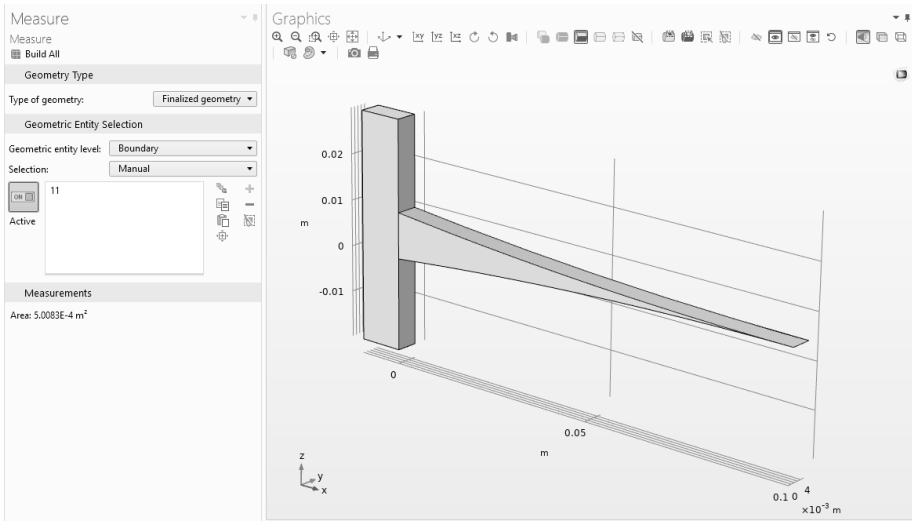


FIGURE 9.67. Measuring fin top surface area for 3D side-concave fin with rectangular cross section.

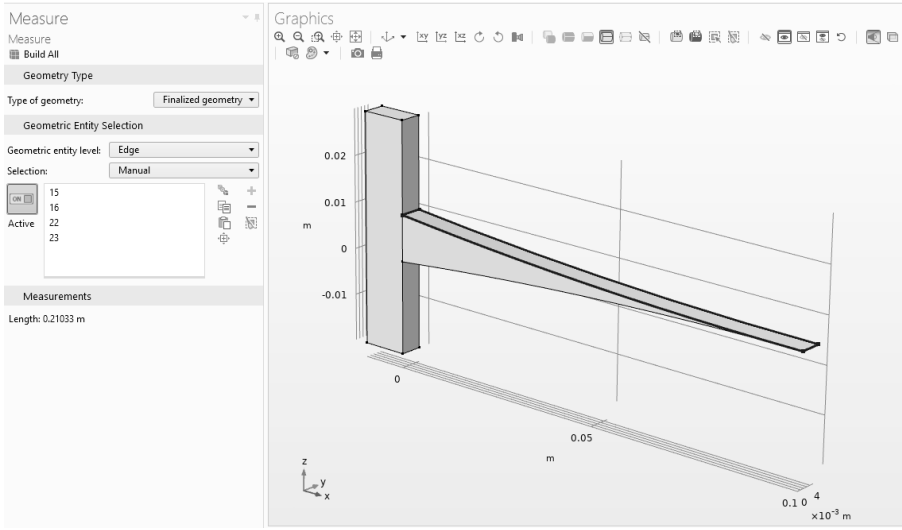


FIGURE 9.68. Measuring fin top surface perimeter for 3D side-concave fin with rectangular cross section.

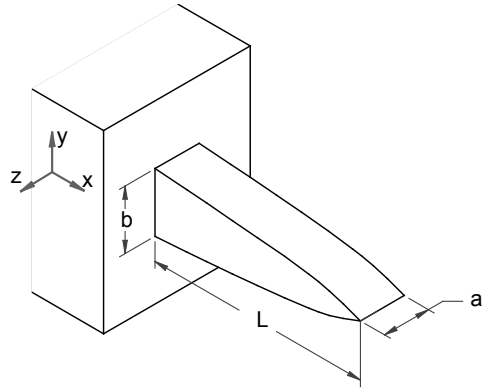


FIGURE 9.69. 3D side-convex fin with rectangular cross section geometry (drawings created in Solid Edge).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	th2*L	5E-4 m^2	top surface are
bb1	0.5[m]	0.5 m	constant
delta0	1[m]	1 m	coefficient in ellipse
delta01	delta0/44.44	0.022502 m	constant
L	0.1[m]	0.1 m	fin length
mesh_size	0.0025[m]	0.0025 m	mesh size
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
perimeter	2*(th2+L)	0.21 m	top surfcae perimeter
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
th1	0.01[m]	0.01 m	fin thickness, b
th2	th1/2	0.005 m	fin depth, a
wall_length	0.05[m]	0.05 m	wall_length

FIGURE 9.70. Parameters used to create 3D side-convex fin with rectangular cross section geometry (global level).

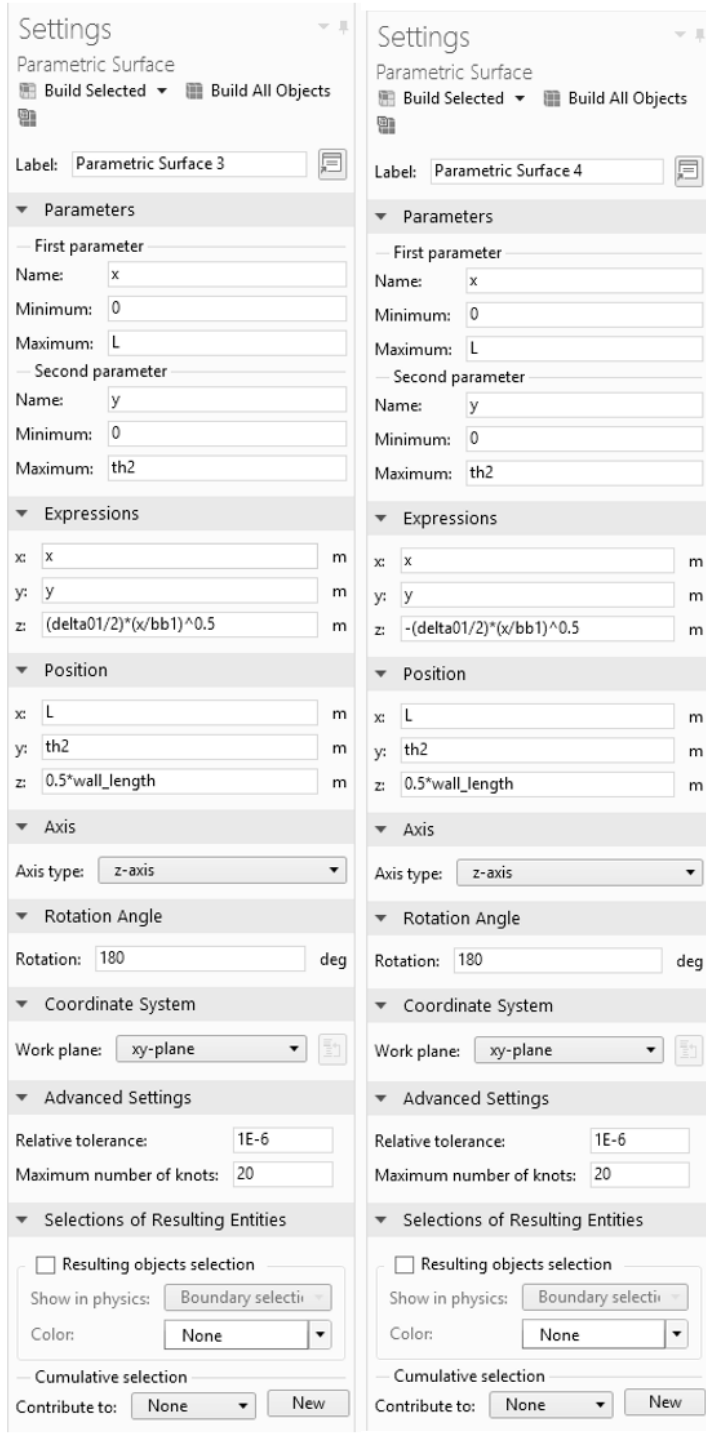


FIGURE 9.71. Defining parametric surfaces for 3D side-convex fin with rectangular cross section (*x-y* work plane, along *z*-coordinate): (a) *ps3* work plane (left), (b) *ps4* work plane (right).

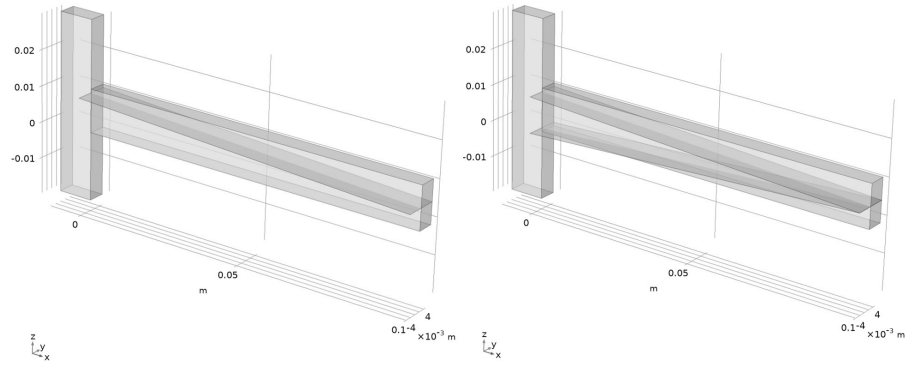


FIGURE 9.72. Creating parametric surfaces for 3D side-convex fin with rectangular cross section (*x-y* work plane, along *z*-coordinate): (a) *ps3* work plane (left), (b) *ps4* work plane (right).

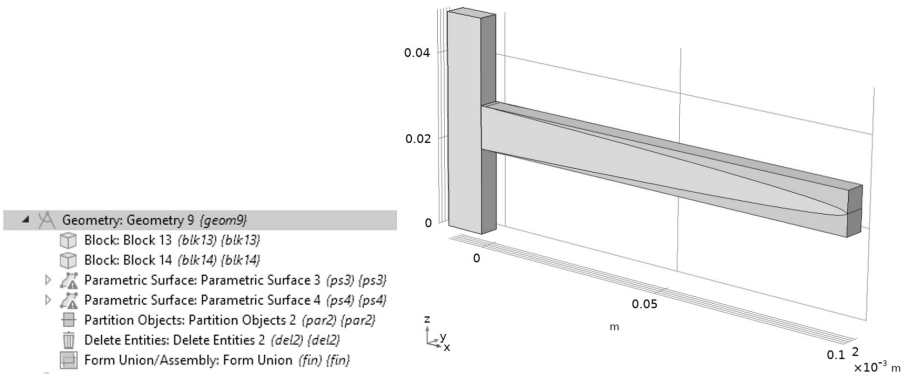


FIGURE 9.73. (a) Geometry building components for 3D side-convex fin with rectangular cross section (left), (b) Purple-highlighted entities to be deleted (right) (*ps3* and *ps4* through *x-y* work plane, along *z*-coordinate).

Geometry and meshed geometry are shown in Figure 9.74. The horizontal part (fin) is meshed with hexahedral elements while the vertical part (wall) is meshed with tetrahedral elements, using different mesh sizes that can be defined as appropriate. The fin top surface area and perimeter measurements are carried out (Figure 9.75 and Figure 9.76). The user may use the geometrical parameters presented in Figure 9.70 to calculate these two values for comparison.

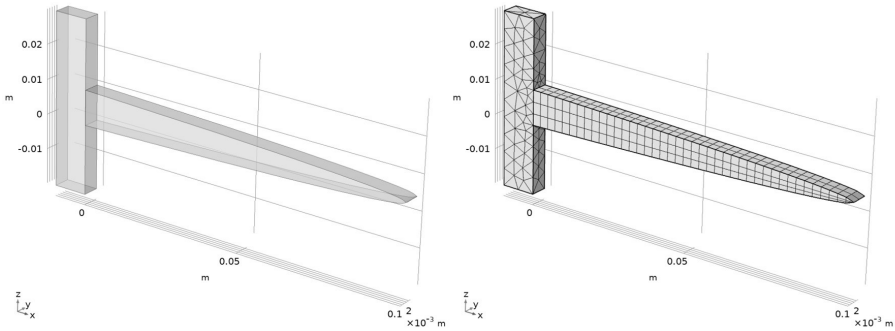


FIGURE 9.74. Creating 3D side-convex fin with rectangular cross section: (a) Geometry (left), (b) Meshed geometry (right).

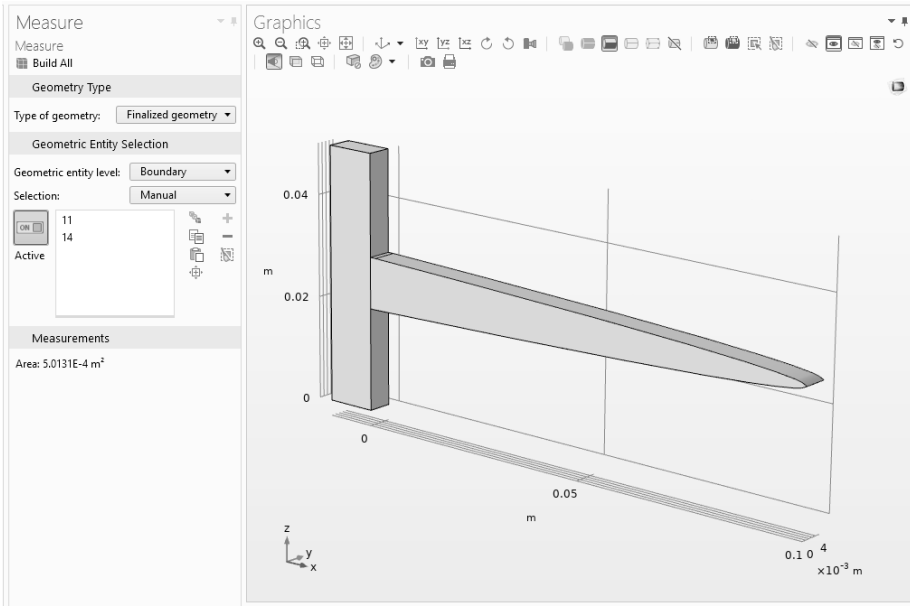


FIGURE 9.75. Measuring fin top surface area for 3D side-convex fin with rectangular cross section.

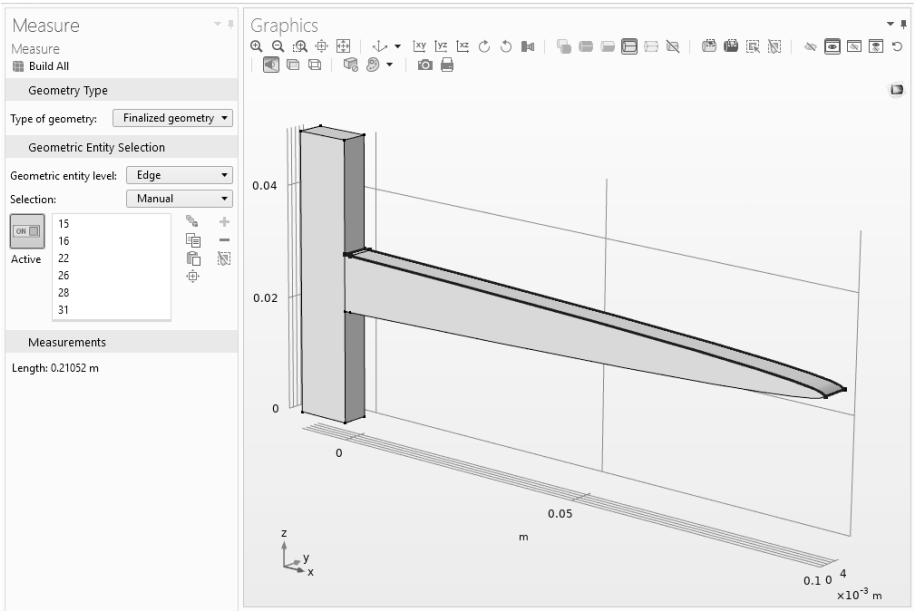


FIGURE 9.76. Measuring fin top surface perimeter for 3D side-convex fin with rectangular cross section.

9.7 SIDE-CONCAVE-TRAPEZOIDAL FIN WITH RECTANGULAR CROSS SECTION

This fin example is largely similar to the side-concave fin geometry, except its narrow end (distant from the wall) has a nonzero height (Figure 9.77). The geometrical parameters to create this geometry are shown in Figure 9.78. The geometry is built by first defining blocks to represent the elongated rectangular fin and the wall. Then, parametric surfaces are introduced to represent the upper and lower concave faces; these surfaces are used as partitioning work planes (Figure 9.79 and Figure 9.80). The concave curves are defined by a quadratic equation: $z = \pm\left(delta02/2\right)\times\left(x/bb2\right)^2$, where the values for *delta02* and *bb2* are given in Figure 9.78. The two work planes are offset to create the trapezoid shape (offset being defined by *Position z* in Figure 9.79). The highlighted upper and lower solid entities remaining after partition are deleted (Figure 9.81). The finished geometry and mesh are shown in Figure 9.82. The horizontal part (fin) is meshed with hexahedral elements while the vertical part (wall) is meshed with tetrahedral ones. The FEM tool measurement feature is used to obtain fin top surface area and perimeter (Figure 9.83 and Figure 9.84). The user may employ the geometrical parameters presented in Figure 9.78 and calculate these two values for verification.

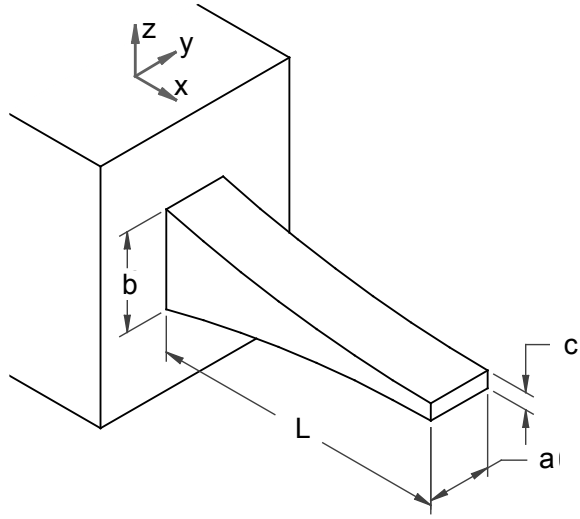


FIGURE 9.77. 3D side-concave-trapezoidal fin with rectangular cross section geometry (drawings created in Solid Edge).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	th2*L	5E-4 m^2	fin top surface area
bb	1[m]	1 m	coefficient in ellipse
bb1	0.5[m]	0.5 m	constant
bb2	bb/2	0.5 m	constant
delta0	1[m]	1 m	coefficient in ellipse
delta01	delta0/44.44	0.022502 m	constant
delta02	delta0-0.85	0.15 m	constant
fin_tip_height	0.004[m]	0.004 m	fin tip height, c
L	0.1[m]	0.1 m	fin length
mesh_size	0.0025[m]	0.0025 m	mesh size
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
perimeter	2*(th1+L)	0.22 m	fin top surface perimeter
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
th1	0.01[m]	0.01 m	fin thickness, b
th2	th1/2	0.005 m	fin depth, a
wall_length	0.05[m]	0.05 m	wall length

FIGURE 9.78. Parameters used to create 3D side-concave-trapezoidal fin with rectangular cross section geometry (global level).

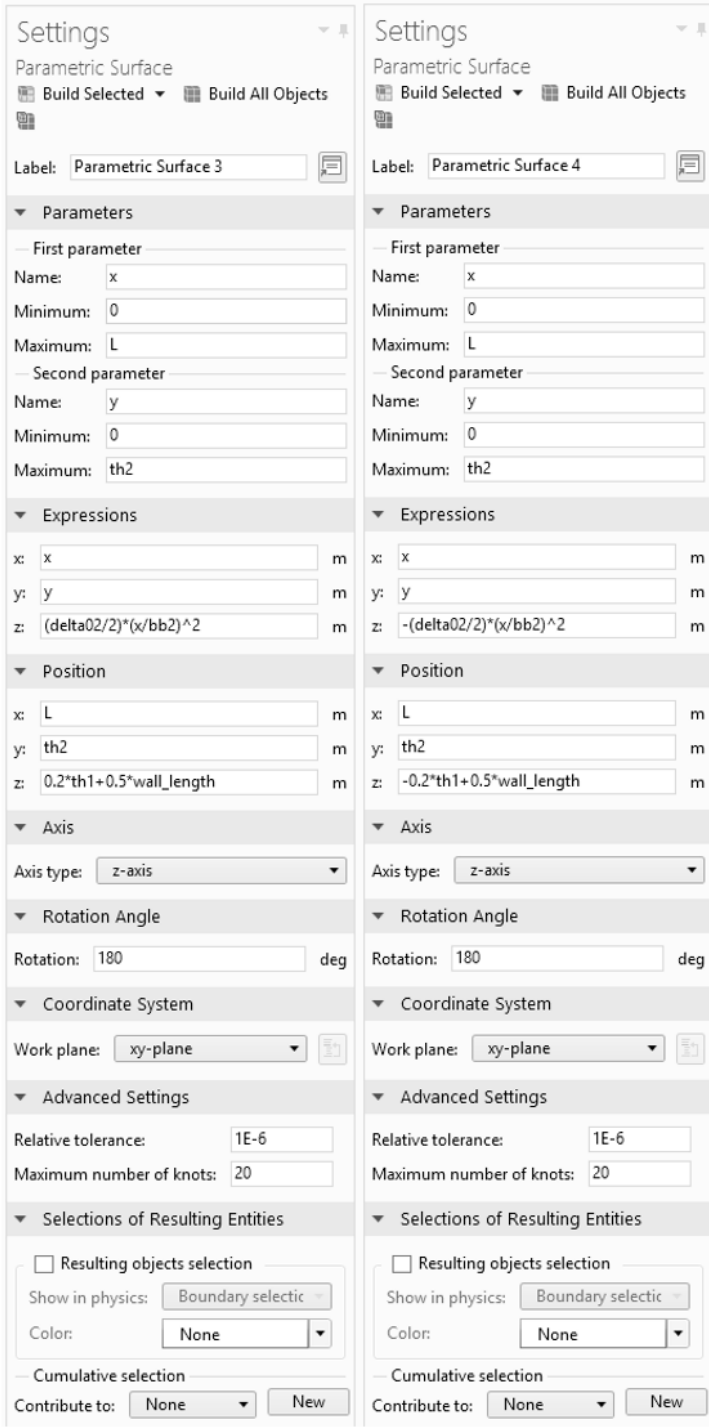


FIGURE 9.79. Defining parametric surfaces for 3D side-concave-trapezoidal fin with rectangular cross section (*x-y* work plane, along *z*-coordinate): (a) *ps3* work plane (left), (b) *ps4* work plane (right).

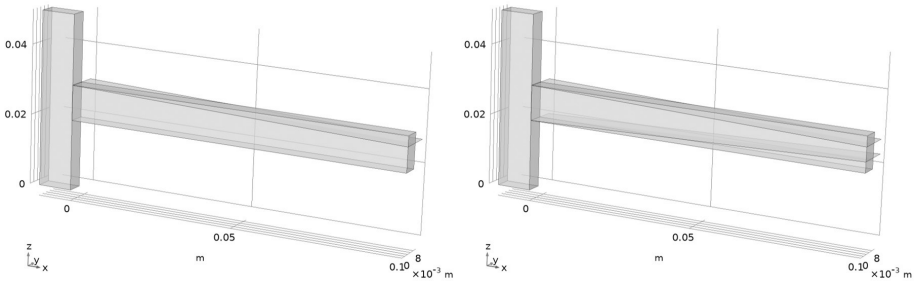


FIGURE 9.80. Creating parametric surfaces for 3D side-concave-trapezoidal fin with rectangular cross section (*x-y* work plane, along *z*-coordinate): (a) *ps3* work plane (left), (b) *ps4* work plane (right).

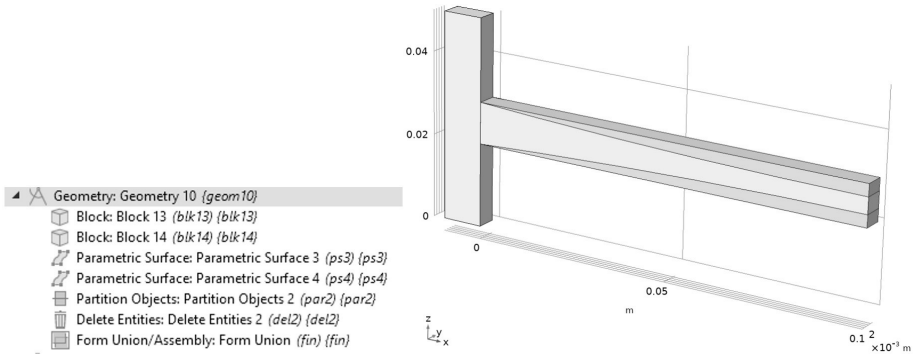


FIGURE 9.81. (a) Geometry building components for 3D side-concave-trapezoidal fin with rectangular cross section (left), (b) Purple-highlighted entities to be deleted (right) (*ps3* and *ps4* through *x-y* work plane, along *z*-coordinate).

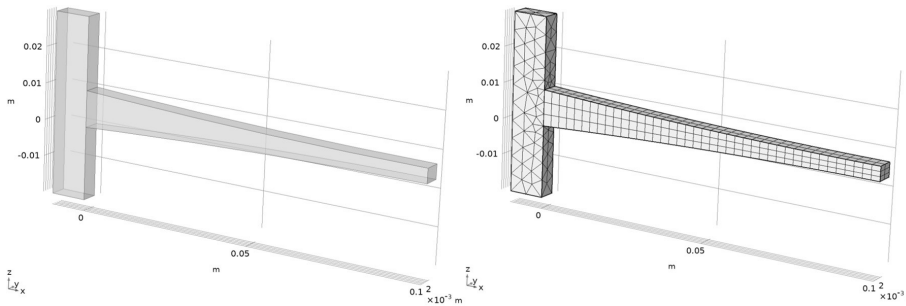


FIGURE 9.82. Creating 3D side-concave-trapezoidal fin with rectangular cross section: (a) Geometry (left), (b) Meshed geometry (right).

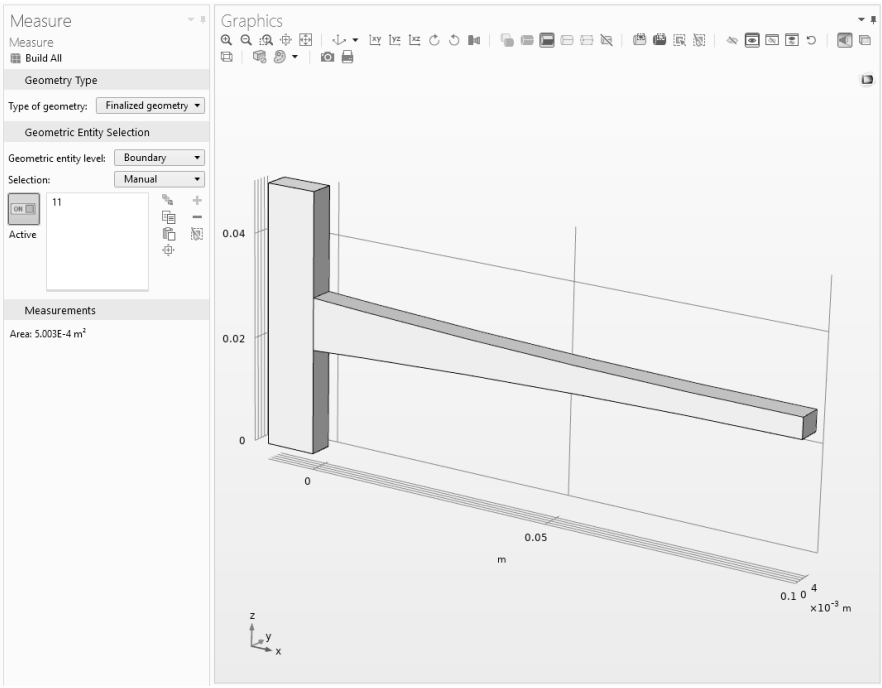


FIGURE 9.83. Measuring fin top surface area for 3D side-concave-trapezoidal fin with rectangular cross section.

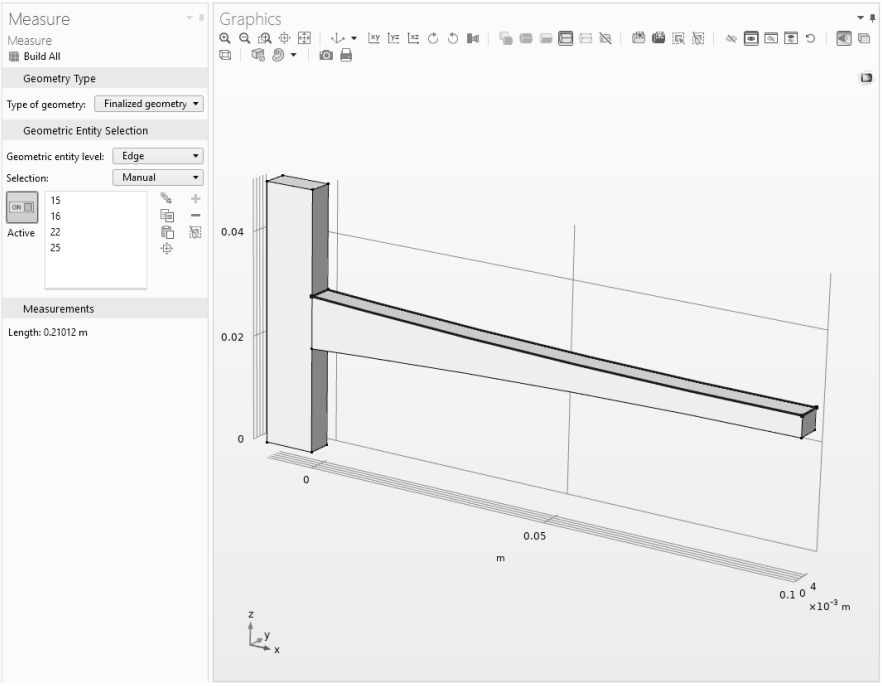


FIGURE 9.84. Measuring fin top surface perimeter for 3D side-concave-trapezoidal fin with rectangular cross section.

9.8 PIN FIN WITH CIRCULAR CROSS SECTION

The fin in this study is characterized by a circular cross section that tapers linearly to a point, producing a conical shape (Figure 9.85). Parameters used to create this geometry are shown in Figure 9.86. The 3D geometry is created in COMSOL Multiphysics internal modeling capabilities—a block for the vertical wall and a cone (Figure 9.87) for the horizontal fin (Figure 9.88). The *z*-*x* work plane, offset along *y*-coordinate, creates the plane of symmetry to partition the geometry (Figure 9.89). The geometry components and subcomponents are presented in Figure 9.88. Figure 9.90 shows the finalized geometry and the meshing. The conical fin is meshed with hexahedral elements while the wall block is meshed with tetrahedral elements—each volume was assigned different a mesh size setting. Measurements made of the fin top surface area and perimeter (Figure 9.91 and Figure 9.92) may be compared against calculations made from the geometrical parameters presented in Figure 9.86.

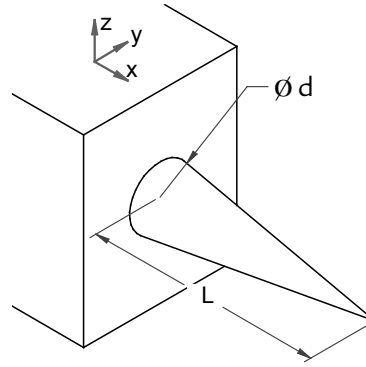


FIGURE 9.85. 3D pin fin with circular cross section geometry (drawings created in Solid Edge).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
L	0.1[m]	0.1 m	fin length
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
th1	0.01[m]	0.01 m	fin top diameter, d2
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
mesh_size	0.0025[m]	0.0025 m	mesh size
wall_length	0.05[m]	0.05 m	wall length
th2	th1	0.01 m	fin bottom diameter, d1

FIGURE 9.86. Parameters used to create 3D pin fin with circular cross section geometry (global level).

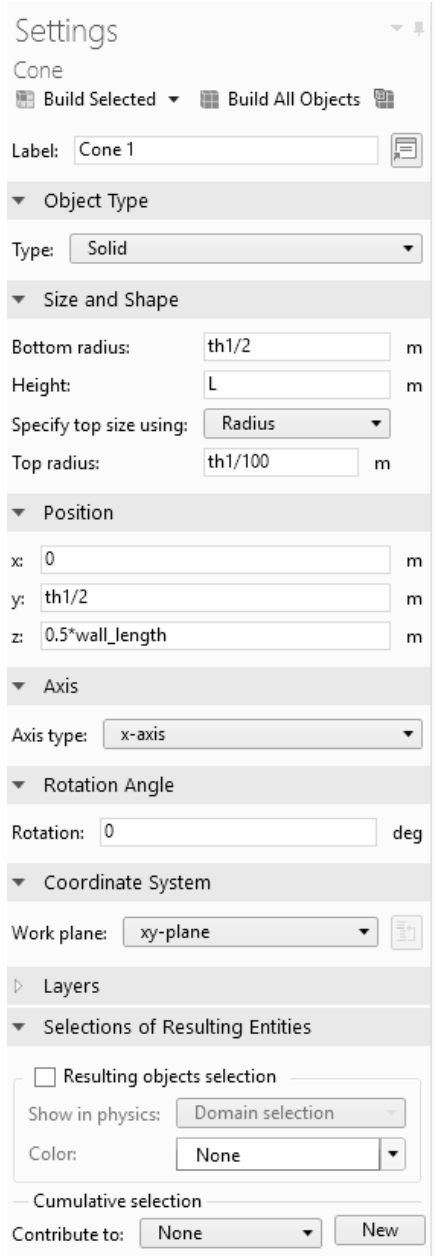


FIGURE 9.87. Cone creation settings for 3D pin fin (*cone1*).

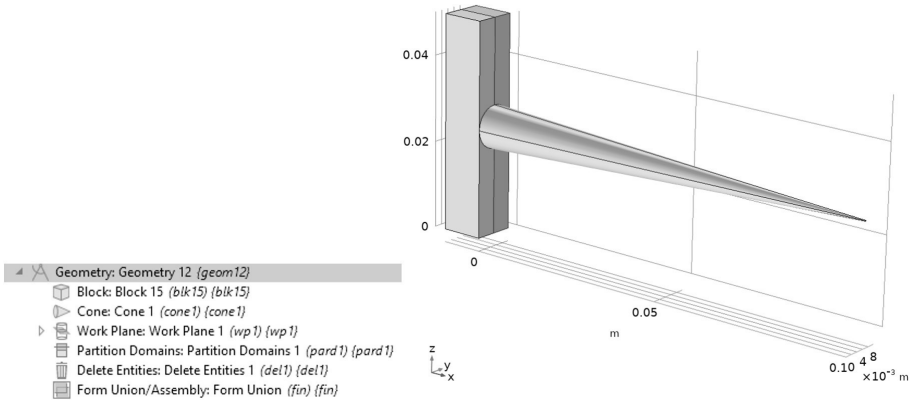


FIGURE 9.88. (a) Geometry building components for 3D pin fin (left), (b) Purple-highlighted entities to be deleted (right).

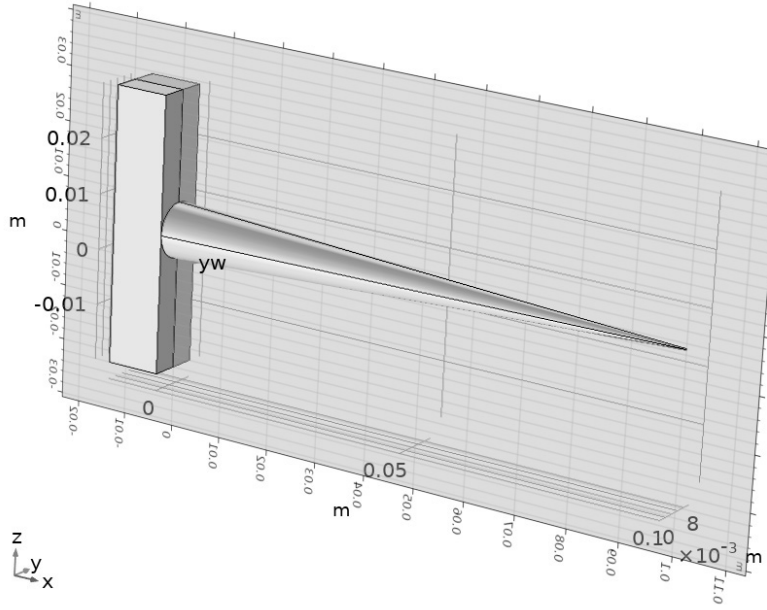


FIGURE 9.89. Creating work plane to partition 3D pin fin with circular cross section (*z-x* work plane, $y = th_1/2$).

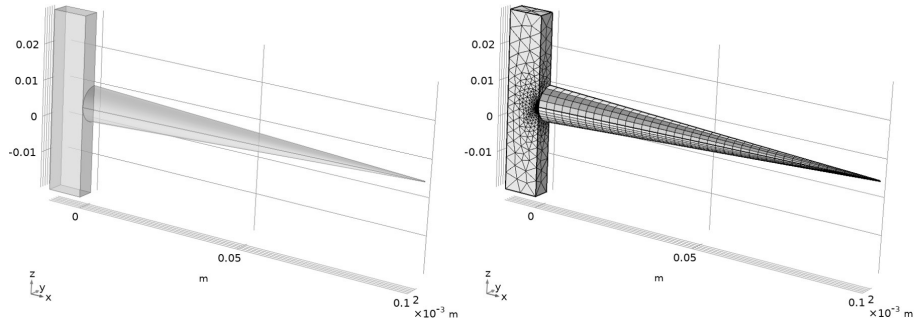


FIGURE 9.90. Creating 3D pin fin with circular cross section: (a) Geometry (left), (b) Meshed geometry (right).

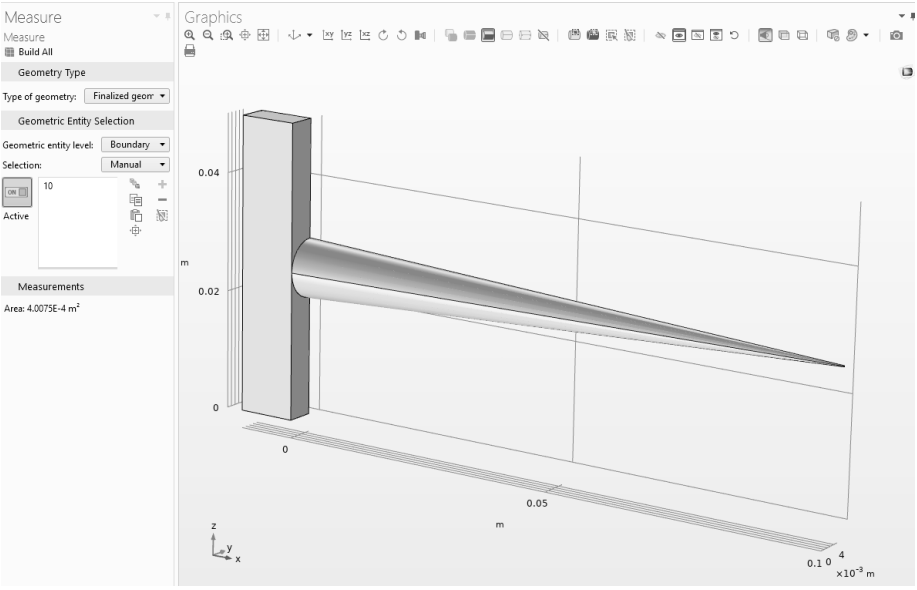


FIGURE 9.91. Measuring fin top surface area for 3D pin fin with circular cross section.

9.9 RADIAL FIN WITH HYPERBOLIC PROFILE

This study looks at a flattened, half-disk (radial) fin that tapers from base to tip following a hyperbolic profile (Figure 9.93). The figure shows a sectioned view of the fin (half of it removed) to more clearly display the surface curvature. Parameters used to create the geometry are presented in Figure 9.94.

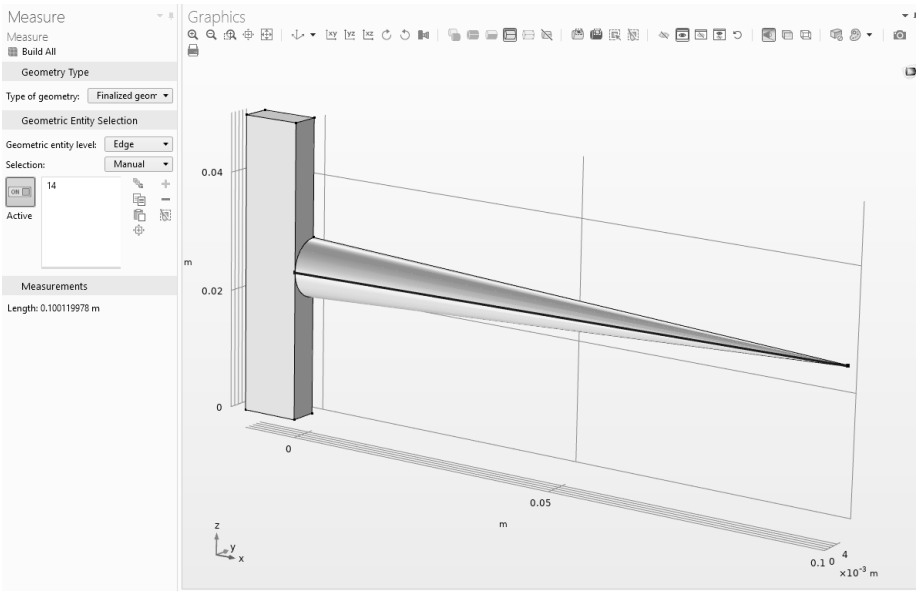


FIGURE 9.92. Measuring fin top surface perimeter for 3D pin fin with circular cross section.

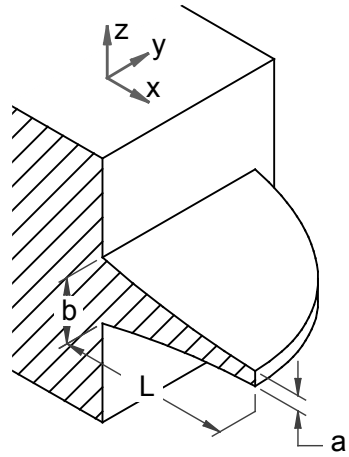


FIGURE 9.93. 3D radial fin with hyperbolic profile geometry (drawings created in Solid Edge).

The 3D geometry of the fin is created in COMSOL Multiphysics CAD import tool by applying the following steps (Figure 9.95):

(1) create a block (*blk15*) to represent the elongated rectangular wall;
(2) create a sphere (*sph1*) of radius L (Figure 9.96);
(3) define parametric surfaces to represent the upper and lower concave surfaces, which are used as partitioning work planes (*ps7* and *ps8* work planes in Figure 9.97 and Figure 9.98). These surfaces

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	3.1415*L*L/4	0.0078538 m^2	inlet area
bb	1[m]	1 m	coefficient in ellipse
bb1	0.5[m]	0.5 m	constant
bb2	bb/2	0.5 m	constant
bb3	bb/2	0.5 m	constant
d_box	2*L	0.2 m	box dimension
delta0	1[m]	1 m	coefficient in ellipse
delta01	delta0/44.44	0.022502 m	constant
delta02	delta0-0.85	0.15 m	constant
delta03	delta0-0.975	0.025 m	constant
fin_tip_height1	0.004[m]	0.004 m	fin tip height, a1
fin_tip_height2	0.01[m]	0.01 m	fin tip height, a2
L	0.1[m]	0.1 m	fin length
mesh_size	0.01[m]	0.01 m	mesh size
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
premiter	(2*3.1415*L/4)+2*L	0.35708 m	inlet perimeter
rho	998.2071[kg/m^3]	998.21 kg/m^3	density
th1	0.01[m]	0.01 m	fin thickness, b

FIGURE 9.94. Parameters used to create 3D radial fin with hyperbolic profile geometry (global level).

are made along the *z*-coordinate on the *x*-*y* work plane, with a rotation angle of zero degrees;

(4) delete the upper and lower solid entities remaining after partition (Figure 9.99); and

(5) define additional work planes in order to take advantage of the symmetry (Figure 9.100 and Figure 9.101). These work planes are used to divide the solid into four parts—one quarter kept (Figure 9.102).

The final geometry and mesh are shown in Figure 9.102. The radial fin is meshed with hexahedral elements while the vertical column (wall) is meshed with tetrahedral elements. The top fin surface is selected as a mapped mesh area with the results being swept to the entire extended surface body. The fin top surface area and perimeter measurements can be compared to calculations based on the parameter values (Figure 9.94). It is seen that the area (Figure 9.103) and perimeter (Figure 9.104) measurements are consistent with the corresponding calculated ones by parametric studies shown in Figure 9.94. The minor difference seen is due to the curvature introduced to the side profile.

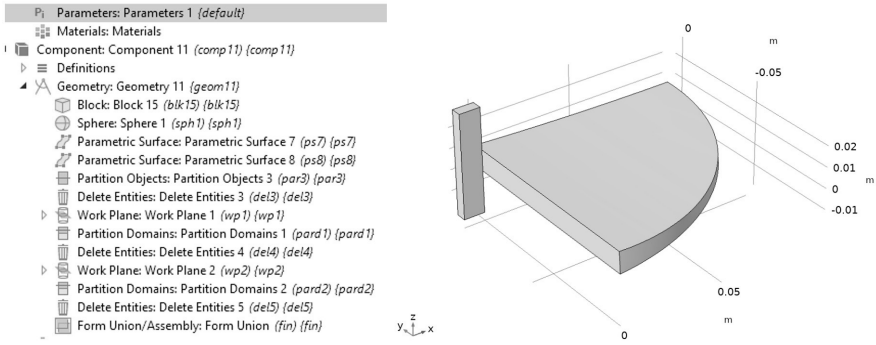


FIGURE 9.95. (a) Geometry building components for 3D radial fin with hyperbolic profile (left), (b) Purple-highlighted entities to be deleted (right).

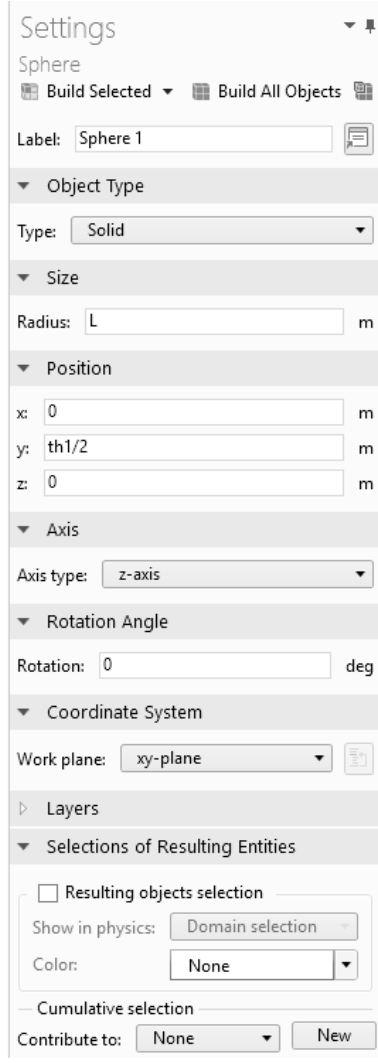


FIGURE 9.96. Sphere creation settings for 3D radial fin with hyperbolic profile (*sph1*).

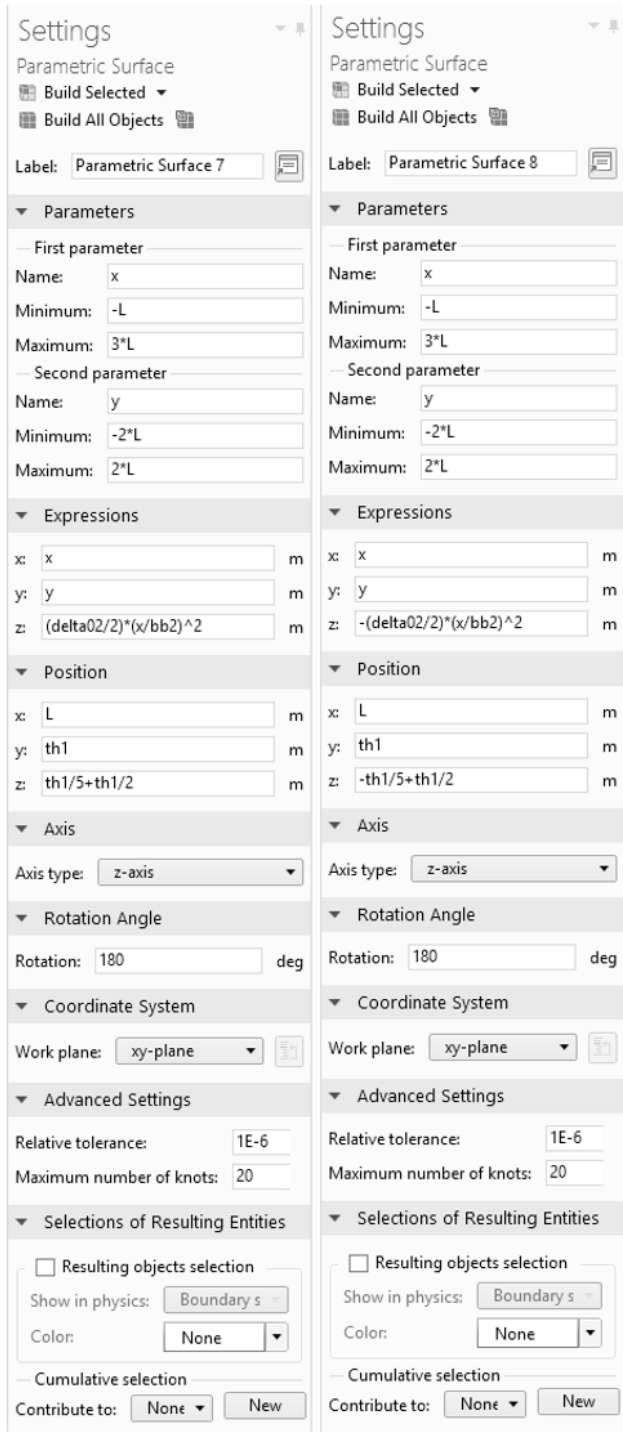


FIGURE 9.97. Defining parametric surfaces for 3D radial fin with hyperbolic profile (*x-y* work plane, along *z*-coordinate): (a) *ps7* work plane (left), (b) *ps8* work plane (right).

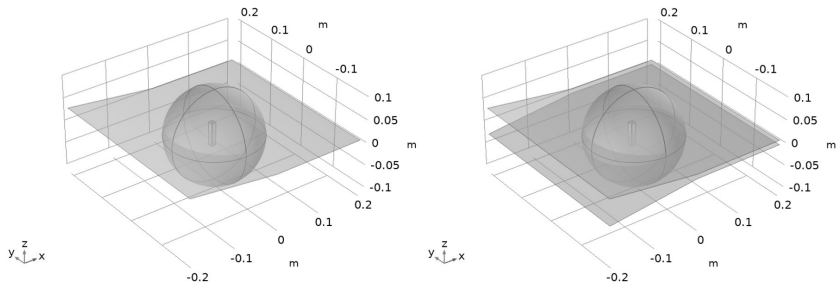


FIGURE 9.98. Creating parametric surfaces for 3D radial fin with hyperbolic profile (*x-y* work plane, along *z*-coordinate): (a) *ps7* work plane (left), (b) *ps8* work plane (right).

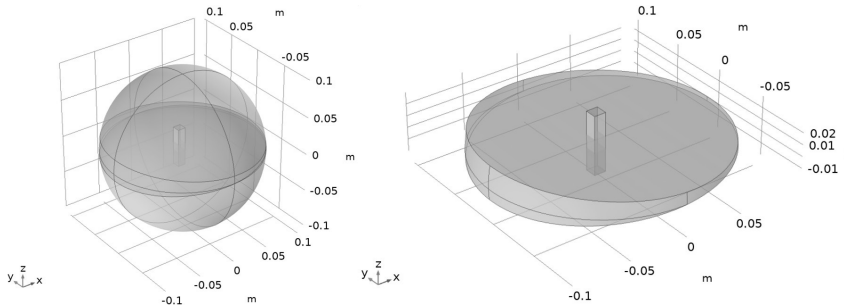


FIGURE 9.99. Processing 3D radial fin with hyperbolic profile (*x-y* work plane, along *z*-coordinate): (a) Partitioning using *ps7* and *ps8* work planes (left), (b) Deleting entities (right).

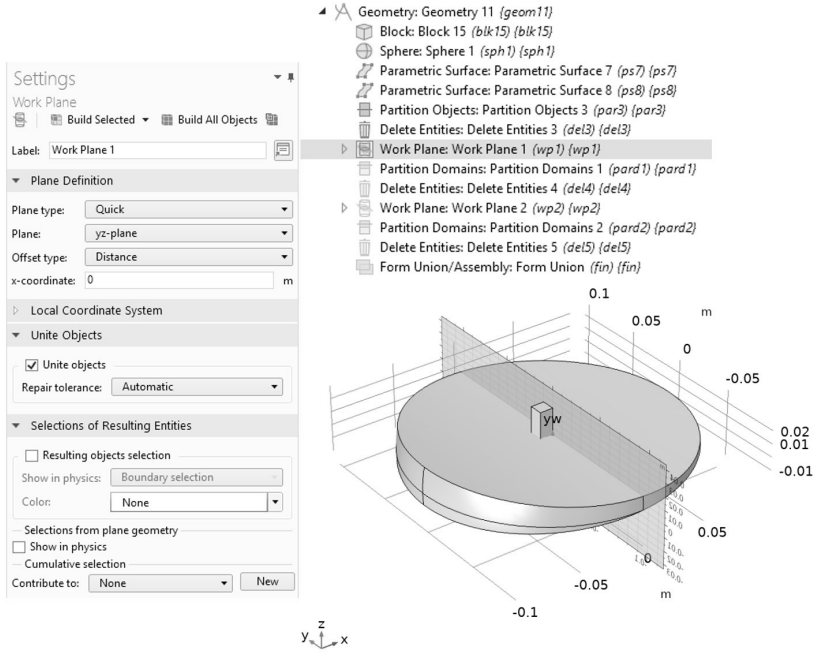


FIGURE 9.100. 3D radial fin with hyperbolic profile (a) Creating work plane (*wp1*, *y-z* work plane, *x = 0*) (left), (b) Geometry building components (top right), (c) Creating work plane to partition the domain (bottom right).

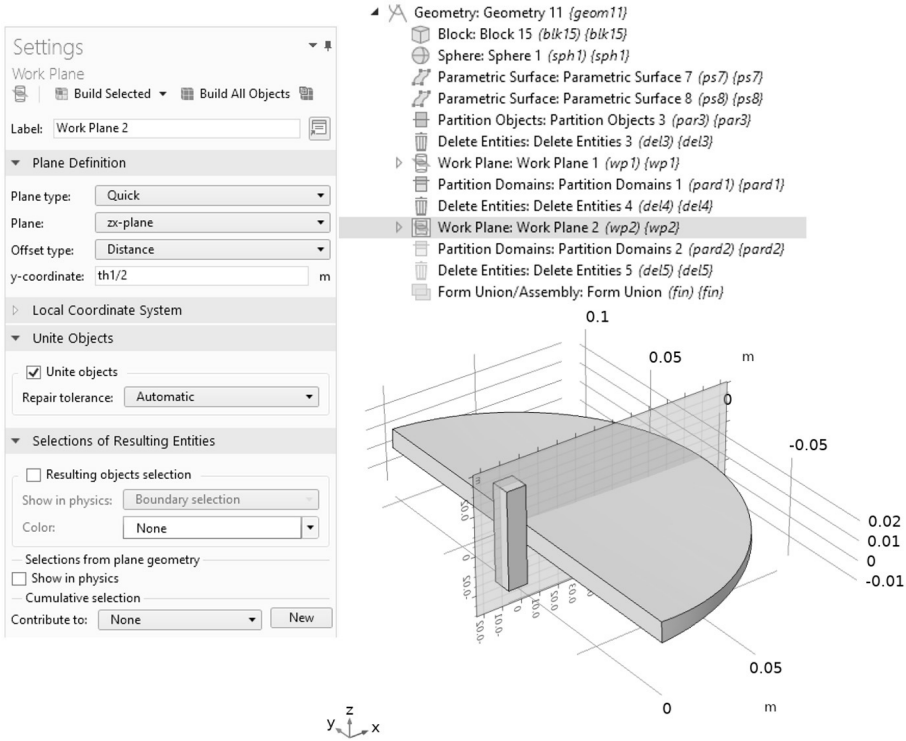


FIGURE 9.101. 3D radial fin with hyperbolic profile (a) Creating work plane (*wp2*, *z-x* work plane, $y = th_1/2$) (left), (b) Geometry building components (top right), (c) Creating work plane to partition the domain (bottom right).

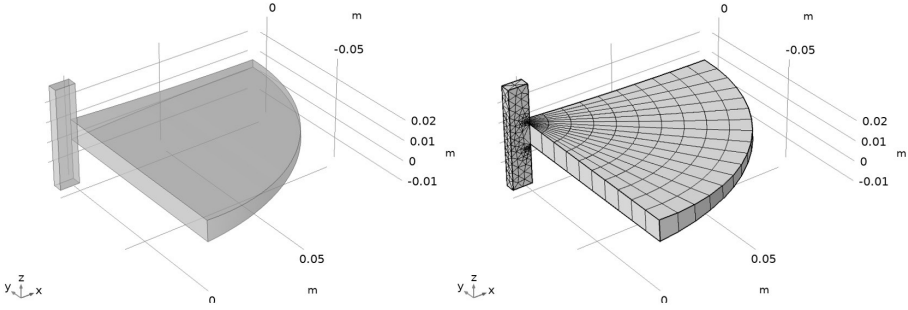


FIGURE 9.102. 3D radial fin with hyperbolic profile: (a) Geometry (left), (b) Meshed geometry (right).

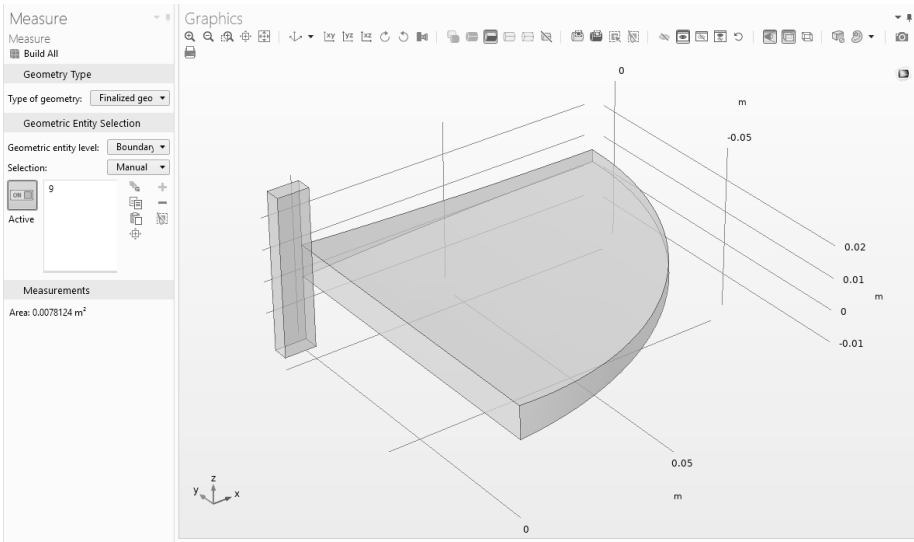


FIGURE 9.103. Measuring radial fin top surface area for 3D radial fin with hyperbolic profile.

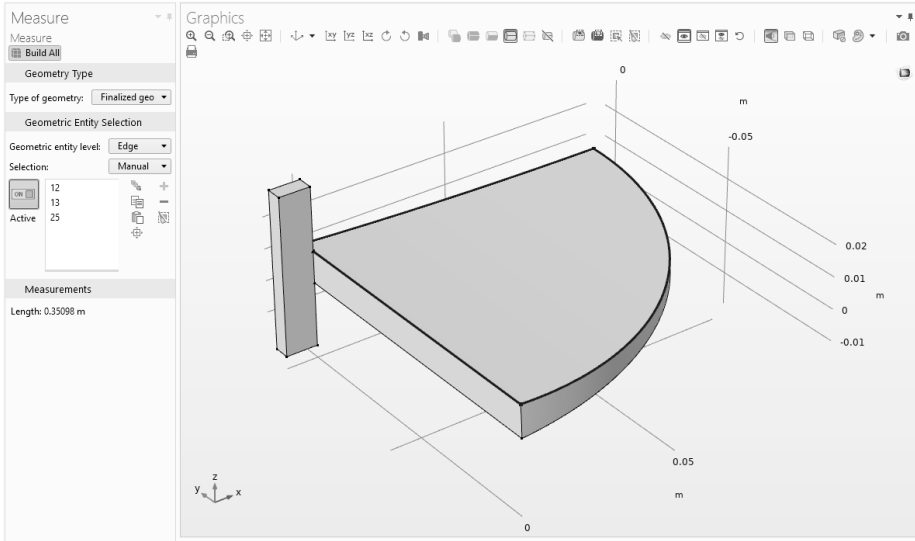


FIGURE 9.104. Measuring radial fin top surface perimeter for 3D radial fin with hyperbolic profile.

9.10 WEBBED RADIAL FIN WITH HYPERBOLIC PROFILE

This study takes the radial fin with hyperbolic profile created in the previous section and adds extra features to it in the form of pass-through holes that create a webbing structure (Figure 9.105). As in the previous

section, only half of the model is shown in the figure (i.e., a sectioned view) to highlight the surface curvature. Parameters used to create the geometry are presented in Figure 9.106.

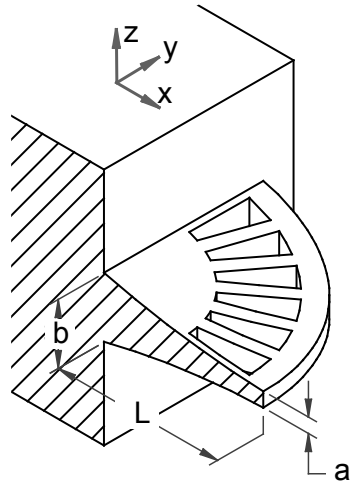


FIGURE 9.105. 3D webbed radial fin with hyperbolic profile geometry (drawings created in Solid Edge).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
area	3.1415*L*L/4	0.0078538 m²	fin top surface area (complete)
bb	1[m]	1 m	coefficient in ellipse
bb1	0.5[m]	0.5 m	constant
bb2	bb/2	0.5 m	constant
bb3	bb/2	0.5 m	constant
delta0	1[m]	1 m	coefficient in ellipse
delta01	delta0/44.44	0.022502 m	constant
delta02	delta0-0.85	0.15 m	constant
delta03	delta0-0.975	0.025 m	constant
fin_tip_height1	0.012[m]	0.012 m	fin tip height, a1
fin_tip_height2	0.0063[m]	0.0063 m	fin tip height, a2
L	0.1[m]	0.1 m	fin length
mesh_size	0.0035[m]	0.0035 m	mesh size
Patm	1[atm]	1.0133E5 Pa	atmospheric pressure
premiter	2*3.1415*L/4	0.15708 m	fin top surface perimeter (complete)
rho	998.2071[kg/m^3]	998.21 kg/m³	density
th1	0.01[m]	0.01 m	fin thickness, b

FIGURE 9.106. Parameters used to create 3D webbed radial fin with hyperbolic profile geometry (global level).

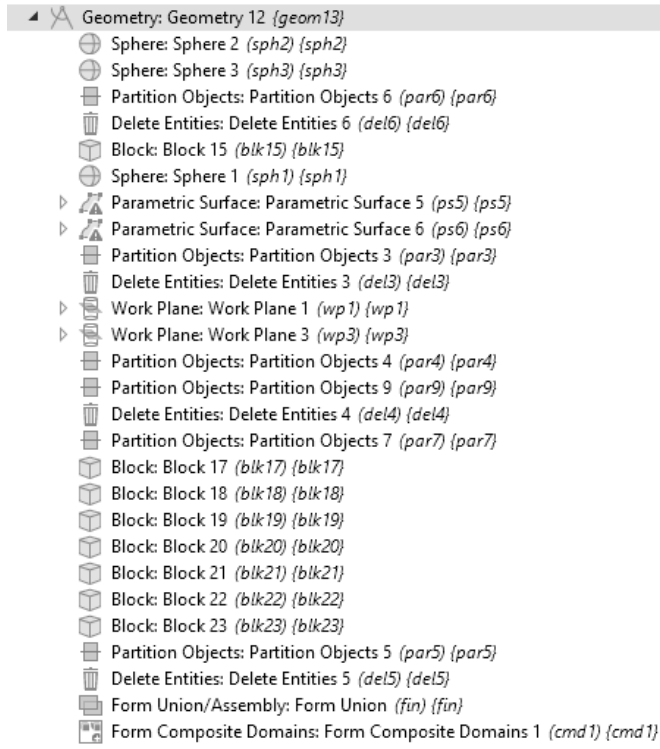


FIGURE 9.107. Geometry building components for 3D webbed radial fin with hyperbolic profile.

This geometry was generated internally in COMSOL Multiphysics. The geometry of this fin is sufficiently complex so that it would be more expedient to use a dedicated CAD tool for its creation. However, despite the extra effort, there are advantages to creating the model internally in COMSOL Multiphysics. It is easier to control the geometry from within the FEM tool, for example, if a parametric study is to be carried out. Additionally, internal geometry generation means one does not need to purchase additional modules to interface with the CAD tools.

For this example, the 3D geometry is created using blocks, spheres, and parametric surfaces. Figure 9.107 shows a complete list of the geometry components used. The most complex element in this geometry is the modeling of the webbed openings (Figure 9.108). Steps to build this geometry are described as follows:

(1) create two concentric spheres with radii of *0.85L* (*sp2*) and *0.5L* (*sp3*). These are to define the fin exterior and interior curve diameters (Figure 9.110b);

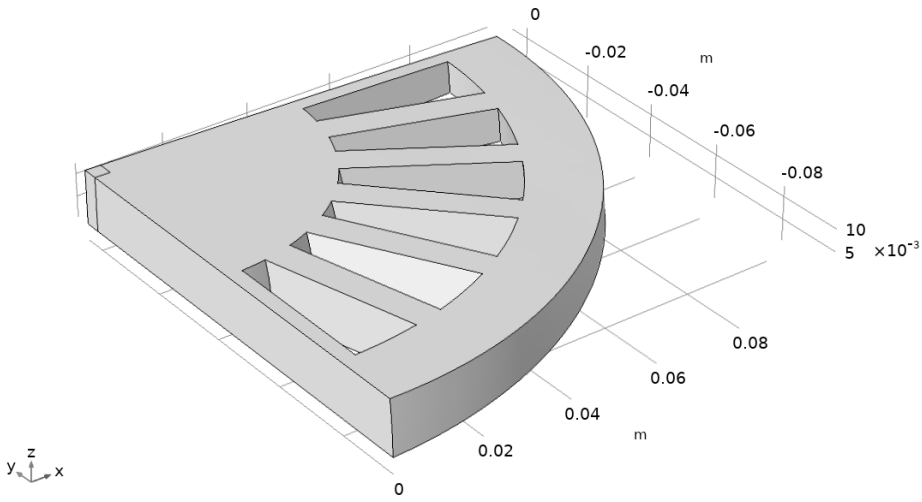


FIGURE 9.108. 3D webbed radial fin with hyperbolic profile.

(2) create a block (*blk15*) to represent the elongated rectangular wall placed at the center of the concentric spheres (Figure 9.110b);
(3) create a sphere (*sph1*) for the extended surface;
(4) define parametric surfaces to represent the upper and lower concave surfaces and use them as partitioning work planes (*ps5* and *ps6* work planes in Figure 9.109, Figure 9.111, and Figure 9.112). Figure 9.110 shows the spheres and blocks before partitioning. The upper and lower solid entities remaining after partition are deleted (Figure 9.113);
(5) define *z-x* and *y-z* work planes (*wp1 and wp3*) in order to take advantage of the symmetry (Figure 9.114). These work planes are used in order to divide the solid into four parts—the right quarter that is closer to the viewer is kept (Figure 9.115); and
(6) define blocks (*blk17* to *blk23*) passing through the centerline to create objects corresponding to the webs and use them as partitioning objects for the remaining geometry (Figure 9.116 and Figure 9.117).

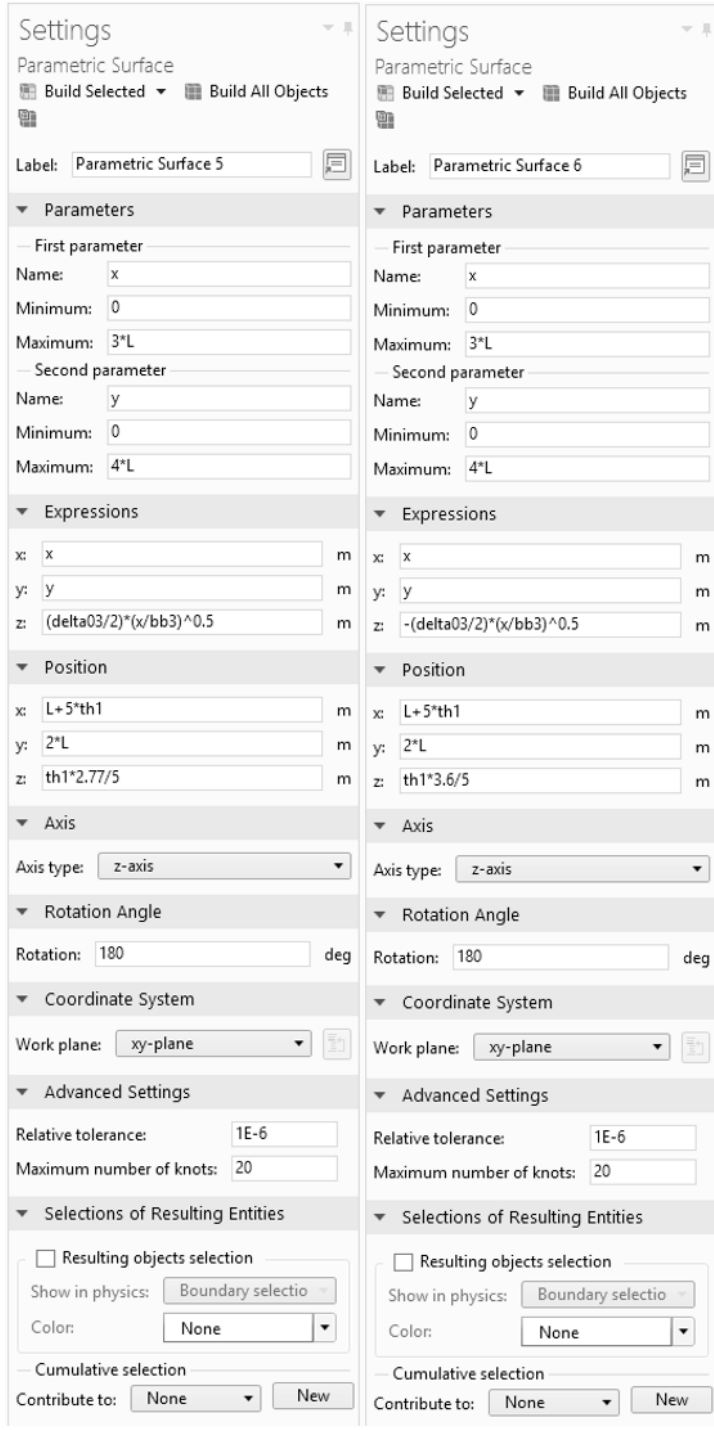


FIGURE 9.109. Defining parametric surfaces for 3D webbed radial fin with hyperbolic profile (*x-y* work plane, along *z*-coordinate): (a) *ps5* work plane (left), (b) *ps6* work plane (right).

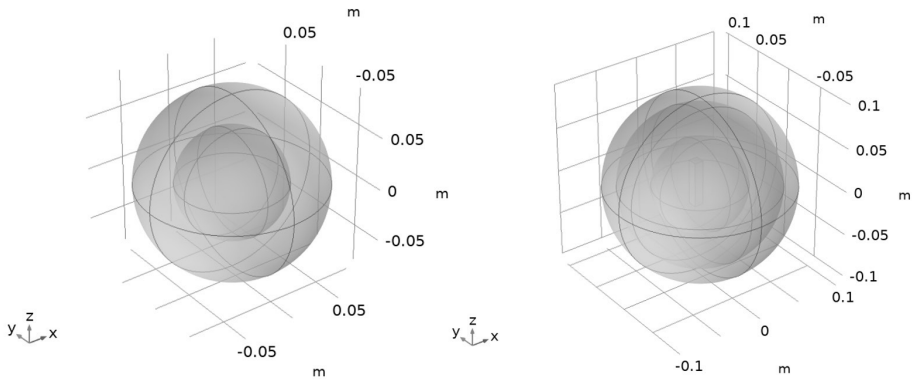


FIGURE 9.110. Creating geometries for 3D webbed radial fin with hyperbolic profile: (a) Deleting the interior sphere (*sp3*) (left), (b) Adding the block and sphere (*blk15* and *sp1*) (right).

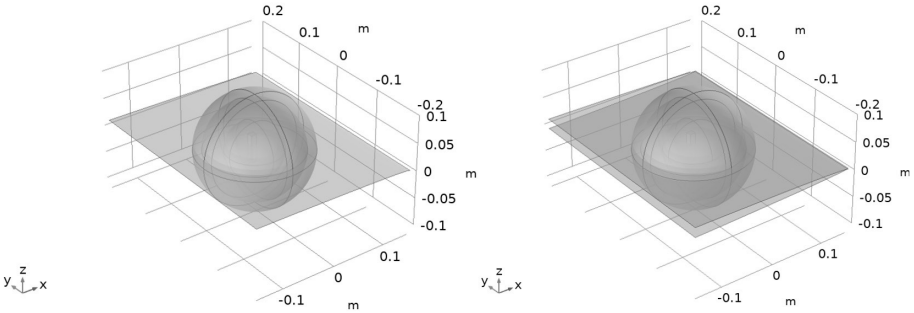


FIGURE 9.111. Creating parametric surfaces for 3D webbed radial fin with hyperbolic profile (*x-y* work plane, along *z*-coordinate), transparent view: (a) *ps5* work plane (left), (b) *ps6* work plane (right).

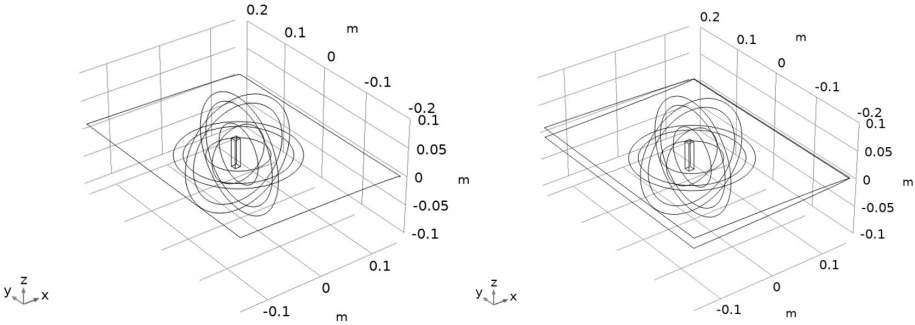


FIGURE 9.112. Creating parametric surfaces for 3D webbed radial fin with hyperbolic profile (*x-y* work plane, along *z*-coordinate), wireframe view: (a) *ps5* work plane (left), (b) *ps6* work plane (right).

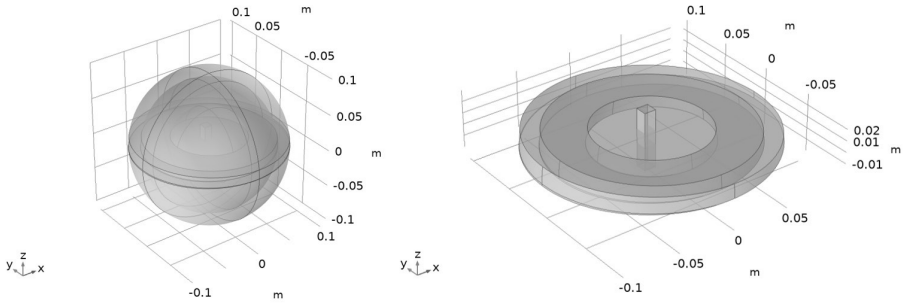


FIGURE 9.113. Processing 3D webbed radial fin with hyperbolic profile: (a) Partitioning using *ps5* and *ps6* work planes (left), (b) Deleting entities (right).

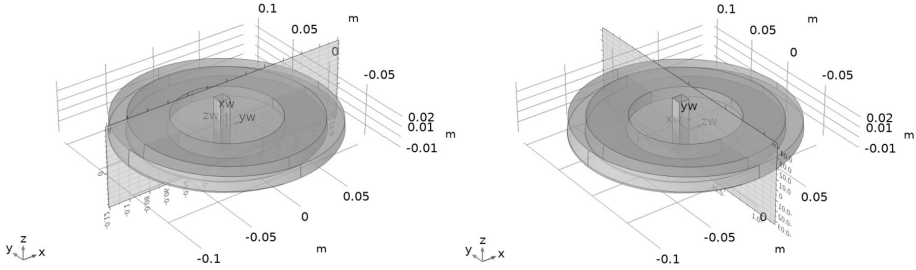


FIGURE 9.114. Partitioning 3D webbed radial fin with hyperbolic profile with defined work planes: (a) *wp1*, *z-x* work plane, $y = th_1/2$, (b) *wp3*, *y-z* work plane, $x = 0$.

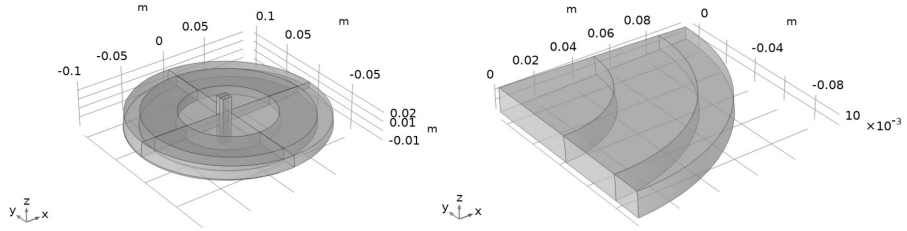


FIGURE 9.115. Creating partitions for 3D webbed radial fin with hyperbolic profile: (a) Identifying the right section as the wanted quarter, (b) Remaining quarter after deleting the purple-highlighted regions.

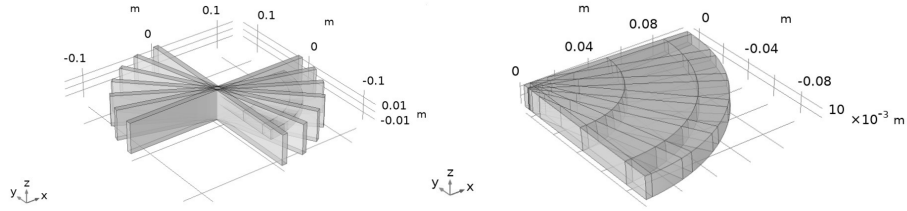


FIGURE 9.116. Processing 3D webbed radial fin with hyperbolic profile: (a) Building central webbing blocks (*blk17* to *blk23*) (left), (b) Creating partitions using the created blocks.

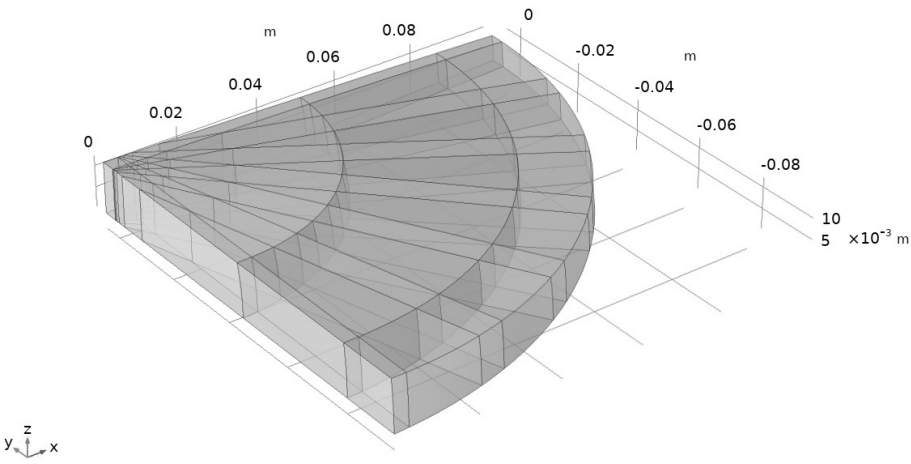


FIGURE 9.117. 3D webbed radial fin with hyperbolic profile, purple-highlighted entities are to be deleted.

Completed geometry is displayed in Figure 9.108 and the mesh is shown in Figure 9.118. The horizontal part (fin) is meshed with free tetrahedral elements while the vertical part (wall) is not meshed in this example. The fin top surface area and perimeter are measured in the model (Figure 9.119 and Figure 9.120) and the results can be compared with the calculated values from the parameter table (Figure 9.106).

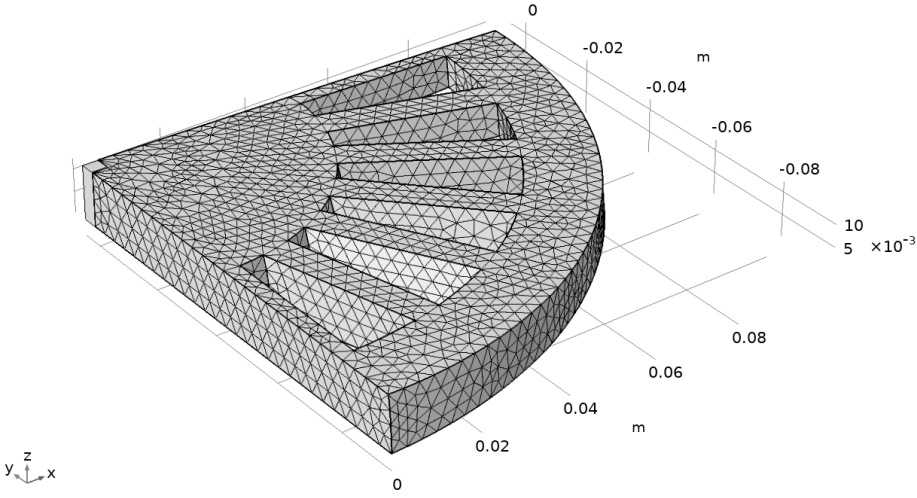


FIGURE 9.118. 3D meshed webbed radial fin with hyperbolic profile.

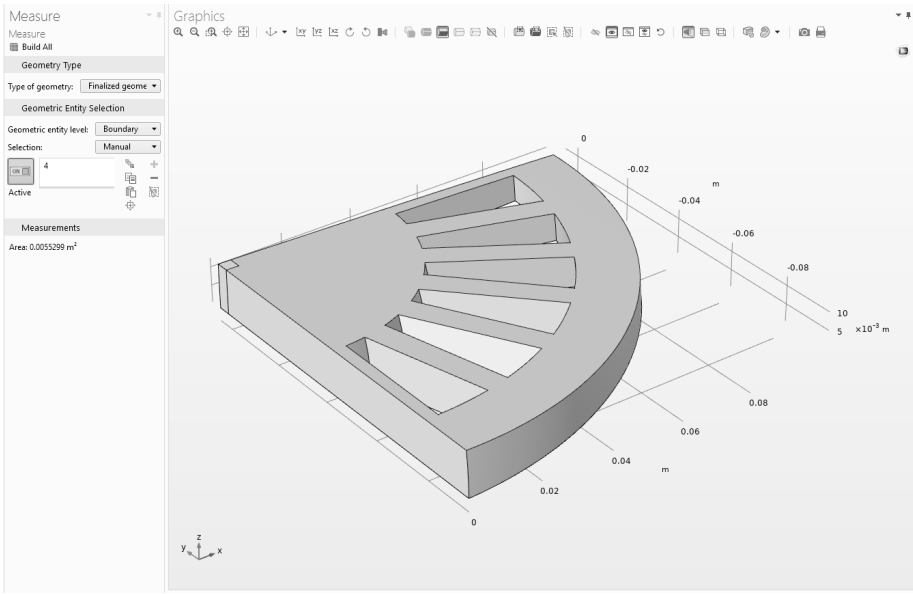


FIGURE 9.119. Measuring webbed radial fin top surface area for 3D webbed radial fin with hyperbolic profile.

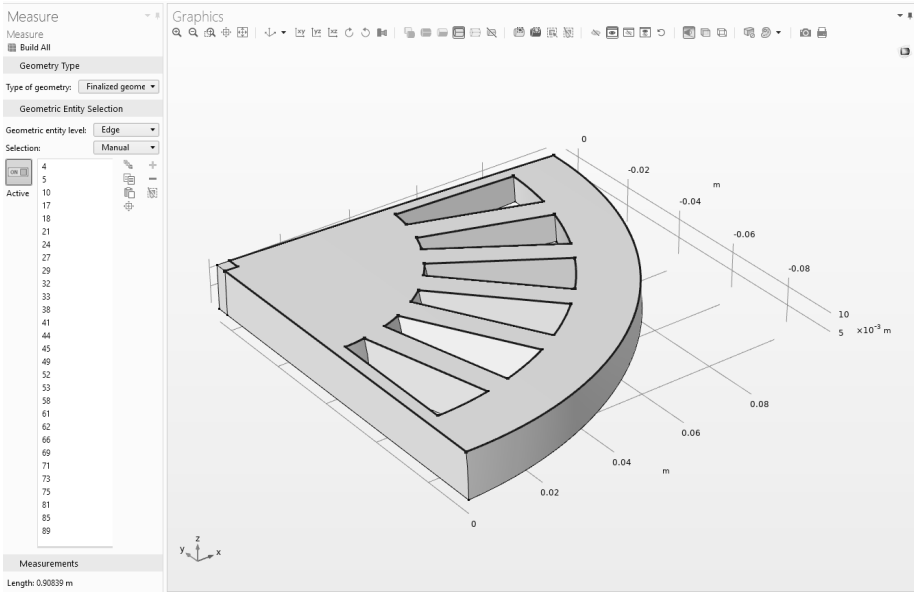


FIGURE 9.120. Measuring webbed radial fin top surface perimeter for 3D webbed radial fin with hyperbolic profile.

9.11 ROTINI FIN—A FIN WITH A TWIST

Rotini pastas are short and are corkscrew shaped. It is an Italian term meaning *small wheels*. It is not only a shape that is geometrically interesting, with its twists and turns, but it also works well as a pasta, with its large surface area taking up all that sauce. If an observant reader ever made a rotini pasta, they would soon learn that these pasta shapes cool faster than other types such as spaghetti (the long stranded thin ones) [181]. The Author believes that this fast cooling must be due to the very good heat dissipation properties, which can be explained by large surface area. The 3D rotini fin employed in this study is shown in Figure 9.121.

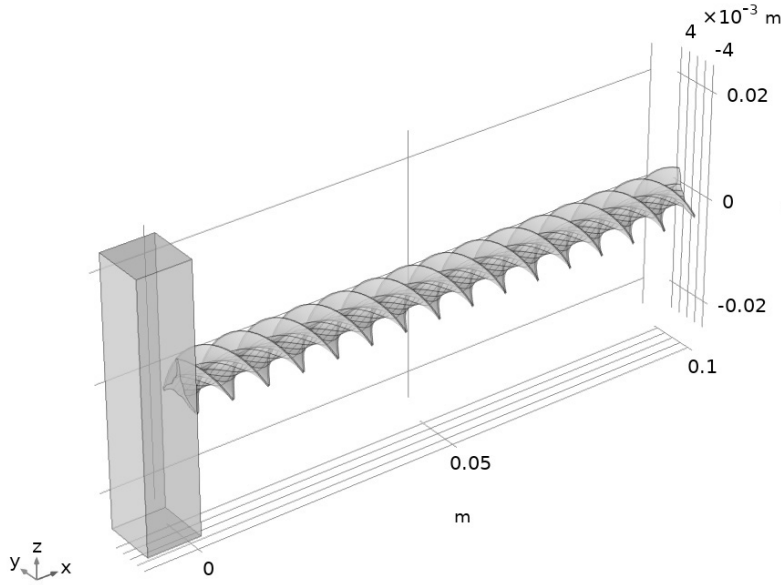


FIGURE 9.121. 3D rotini fin geometry.

To create this complex 3D geometry, a dedicated third-party CAD tool (Solid Edge) is employed. The cross section of this fin is presented in Figure 9.122. The geometrical parameters for this geometry are shown in Figure 9.123. The 3D geometry is obtained by combining components created internally in COMSOL Multiphysics with those imported from the CAD tool. Figure 9.126a shows the geometry components used to create this part. After importing the rotini fin geometry into the FEM tool (*imp1*), a block (*blk1*) is created to represent the wall. It is placed so as to be in contact with one end of the fin. Figure 9.124 shows

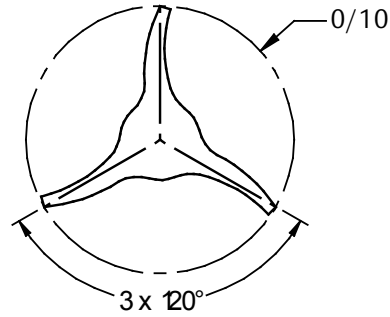


FIGURE 9.122. Dimensions of rotini fin cross section (created in Solid Edge).

Settings

Parameters

Label: Parameters 1

Parameters

Name	Expression	Value	Description
L	0.1[m]	0.1 m	fin wall length
th1	0.01[m]	0.01 m	wall width
th2	th1	0.01 m	wall depth
height	L/2	0.05 m	wall height
mesh_size	0.04[m]	0.04 m	mesh size

FIGURE 9.123. Parameters used to create 3D rotini fin (global level).

the import settings used. There are different importable file types such as ∗*.stl* format, mesh, 3D CAD file, or COMSOL Multiphysics files. In this exercise, a file in ∗*.stp* format was imported; it belongs to the 3D CAD file grouping.

While all other fin geometries presented in the earlier examples were truly symmetric about at least one plane, the rotini fin is not symmetric about any plane. However, one may hypothesize that, for the purposes of heat transfer analysis, the key factors that affect it are the volume of the part and its surface area. Thus, an astute analyst may test the idea of dividing the rotini fin by half with an *x*-*z* plane and checking whether results comparable to the full model can be obtained. If this proves to be the case and if one needs to do multiple solutions with varying parameters, one can then shorten the solution time by working with only half of the geometry. To explore the idea of working with half of the rotini fin geometry, the geometry is divided in half by the work plane in Figure 9.125. The left domain (when facing the wall), the highlighted entities in Figure 9.126b, are deleted.

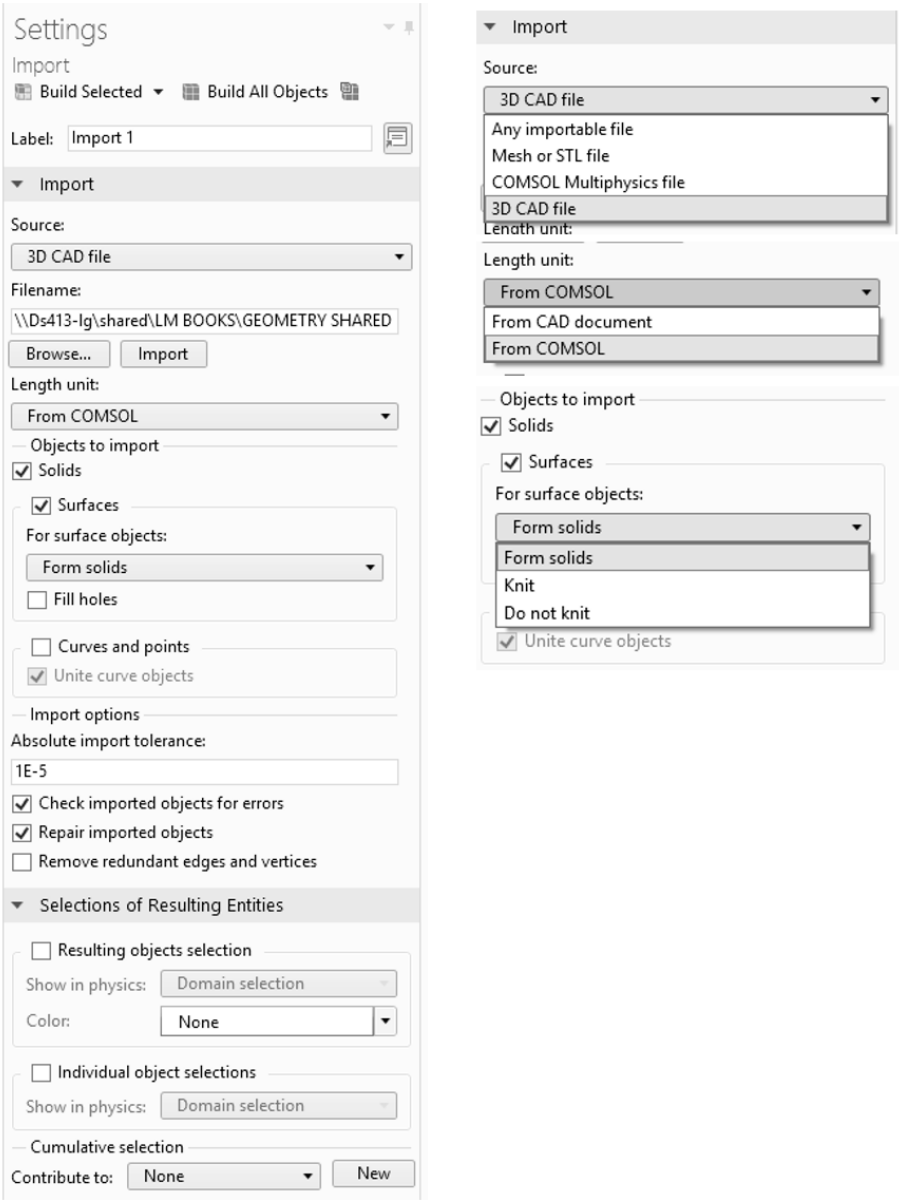


FIGURE 9.124. Geometry import settings.

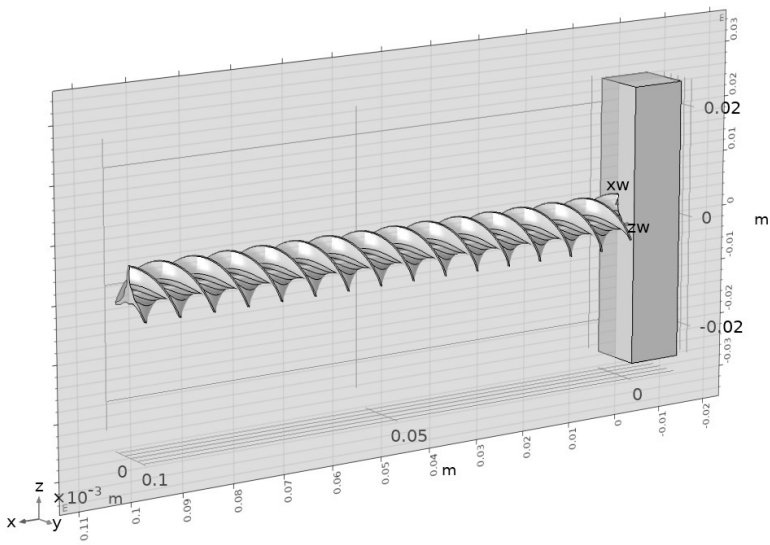


FIGURE 9.125. Creating work plane for 3D rotini fin (*wp3*, *z-x* work plane, *y = 0*).

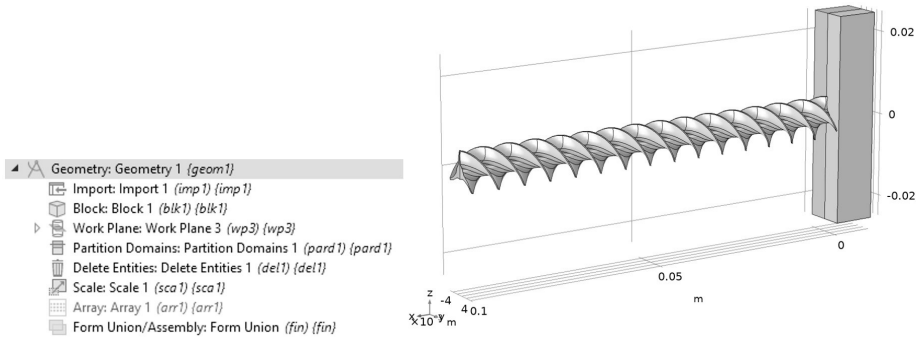


FIGURE 9.126. (a) Geometry building components for 3D rotini fin (left), (b) Purple-highlighted entities to be deleted (right), (*wp3*, *z-x* work plane, *y = 0*).

Figure 9.127 presents the mesh for the finalized half-fin geometry. Tetrahedral mesh elements are employed to mesh the fin and wall; however, each is selected as a separate domain so that the mesh size can be individually controlled. Figure 9.128 and Figure 9.129 show the volume and surface area for a half-size rotini fin, while Figure 9.130 and Figure 9.131 show the volume and surface area for a full-size rotini fin. The half-fin surface area calculation excludes the flat surface at the partitioning plane, and both area calculations exclude the flat surface of the fin in contact with the wall. The ratio of the volumes and areas is nearly exactly equal to 2, supporting the hypothesis that half of the geometry is likely to produce heat transfer model results equivalent to the full geometry (Table 9.1).

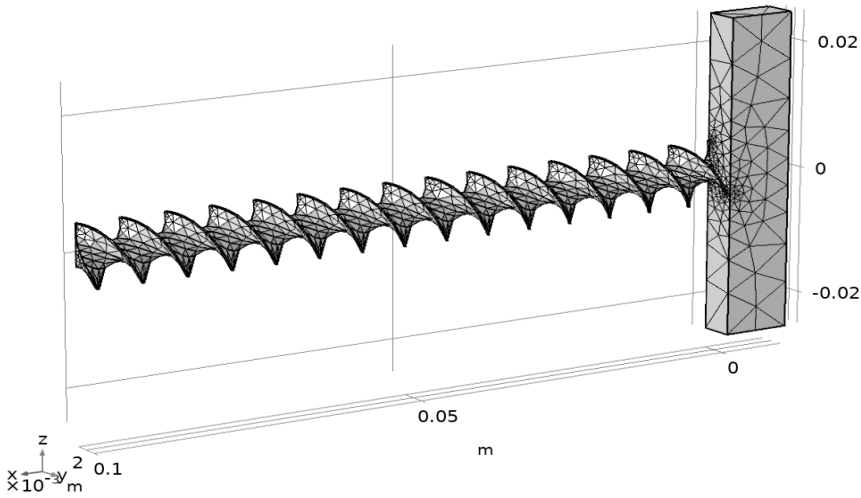


FIGURE 9.127. Creating mesh for 3D rotini fin.

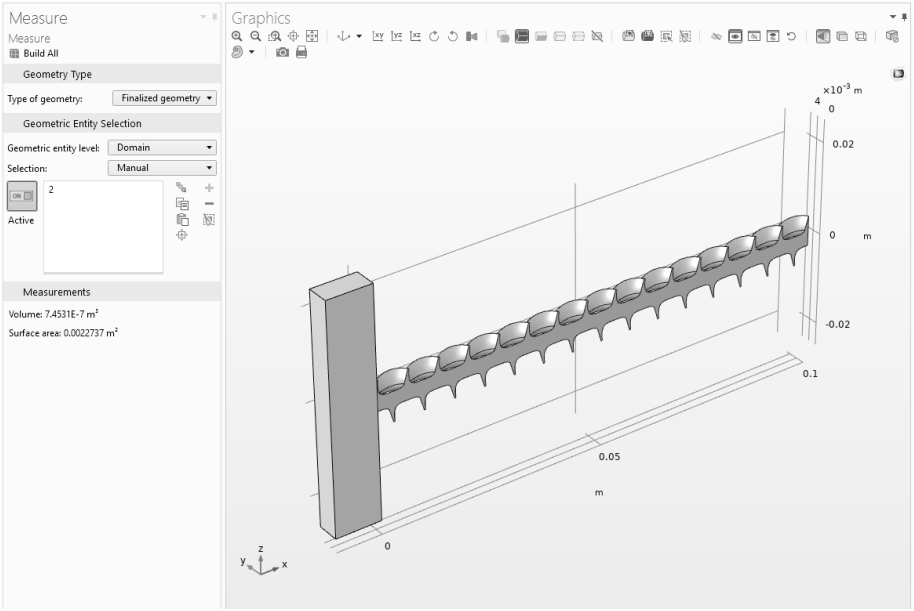


FIGURE 9.128. Measuring volume for 3D half-rotini fin.

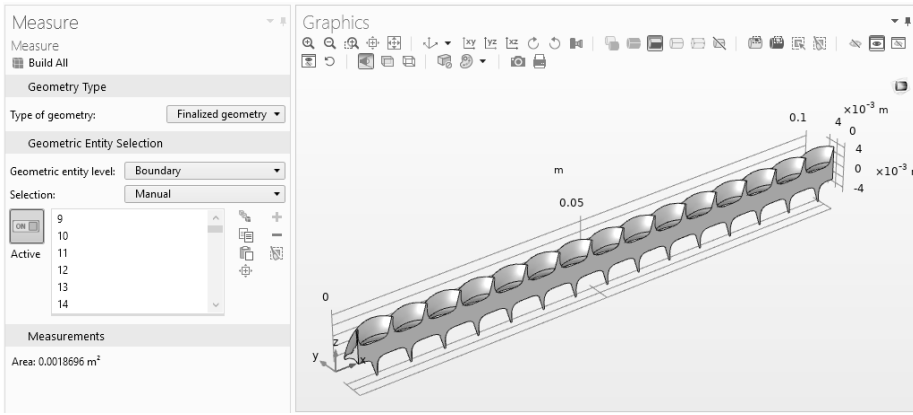


FIGURE 9.129. Measuring convective area for 3D half-rotini fin.

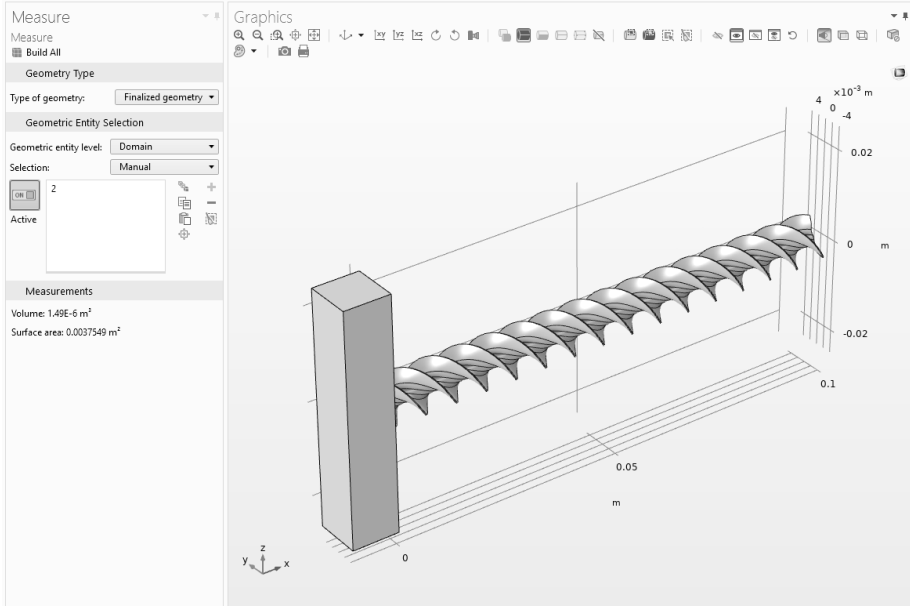


FIGURE 9.130. Measuring volume for 3D rotini fin.

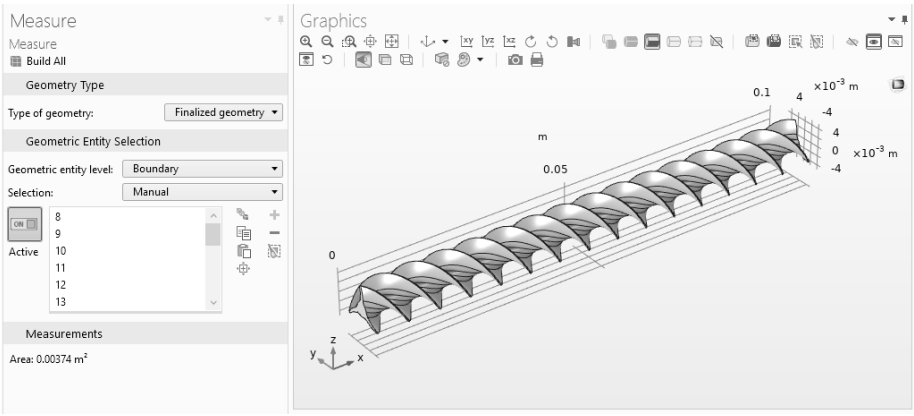


FIGURE 9.131. Measuring convective area for 3D rotini fin.

TABLE 9.1 Convective area and volume for rotini fin.

Parameter	Full Model	Half Model	Ratio (Full/Half)
Volume	1.49E–6	7.4531E–7	1.9992
Area	3.74E–3	1.8696E–3	2.0004
Area/Volume	2.85E+3	3.38E+3	0.8432

9.12 COMPARISON BETWEEN THE FINS

A comparison is made between the fins presented in this work. The comparison looks at the surface-to-volume ratio for each fin geometry (Figure 9.132). To make the comparisons meaningful, an enclosing box for the cross section of 0.01×0.01 m² and the fin length of 0.1 m are assumed. The presented areas and volumes are for the full geometry (not the geometry remaining after partitioning by symmetry plane). The areas and volumes considered are just for the fin, not the wall (i.e., the convective surface area of the fin is being measured). This is the area of the fin which is exposed to the exterior ambient. In this comparison, the rotini fin comes out as the clear winner, with its surface-to-volume ratio of 2,510 1/m, being 2.8 times greater than the fin with the next highest value (cylinder with finned channel).

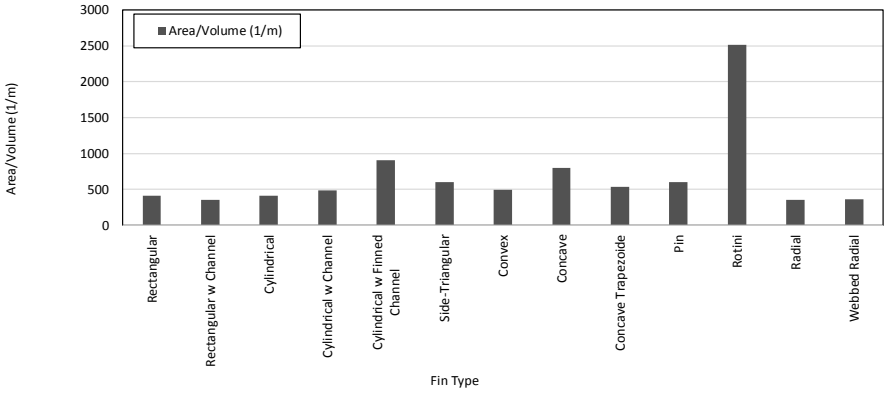


FIGURE 9.132. Comparison between fin area and volume ratios.

CHAPTER 10

GEOMETRY MODELS AND APPLICATIONS

A new feature of COMSOL Multiphysics introduced several years ago is the ability to create applications that provide a simplified user interface to the software modeling capabilities. In this way users with a variety of skill levels may use the geometry configuration settings you have embedded within the application without the need to know the full complexity of the geometry construction. For example, such applications can serve as educational tools that can demonstrate the impact of geometry variations to users in academic and scientific communities. This publication has a companion disc that includes COMSOL Multiphysics model files of the examples presented in Chapter 9 ("Extended Surfaces") (Figure 10.1). Each file includes a geometry model and the related application.

Although meshing the geometry is not the main focus of this work, a mesh is included with geometry models and applications as bonus material and also as a stepping-stone to help the reader gain a better understanding of how the geometry affects other elements of the FEM model. The reader may attempt to work through some of the examples and vary the geometry and mesh parameters as desired to observe their effect. This is particularly useful to identify the mass changes with the subcomponent size. The reader may also use the geometry models and recommended mesh densities as the starting point for the subsequent steps of adding physics and defining the study steps. Applications were set up to follow a consistent pattern for their interface. Most application variables are editable. On occasion when changing the input variables, entities that are to be deleted need to be redefined within the program or reactivated. This is a common practice, especially when

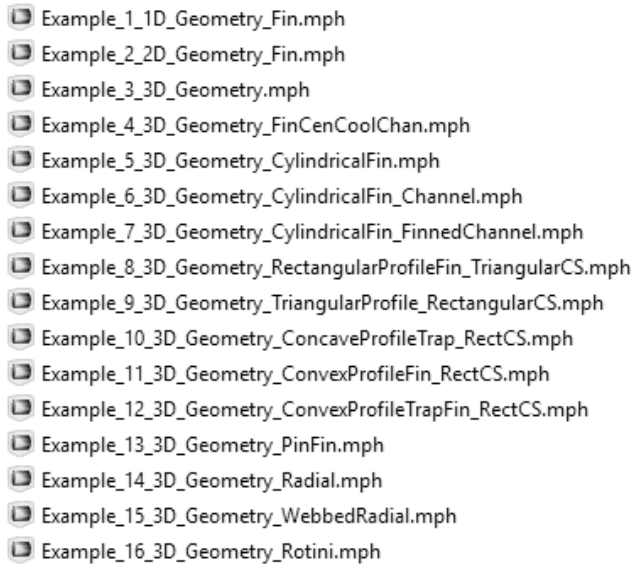


FIGURE 10.1. COMSOL Multiphysics Model files for the example geometries included on the companion disc.

work planes and partitions are defined. Ensure the geometry feature subcomponents are properly set when running the applications with your custom-defined variables or parameters. Check your work before settling for the results.

10.1 OVERVIEW OF THE EXAMPLES PROVIDED

All the example application dialog boxes consist of three sections. Looking at the application interface, from left to right, the sections are as follows: (1) input parameters—geometry size variable(s) and mesh size parameter(s); (2) geometry window with the *Update* button in the top-left corner; and (3) mesh window with the *Update* button in the top-right corner. Each window also includes a separate set of small buttons (top ribbon) used for display control, such as zoom in/out, zoom select, fit, snapshot, and print.

10.2 CASE STUDY 1—GEOMETRY MODEL FOR 1D FIN

This example includes the 1D fin model in the FEM tool and the associated application (Figure 10.2). Note how the meshing feature is configured for this example. The custom meshing size option is used so

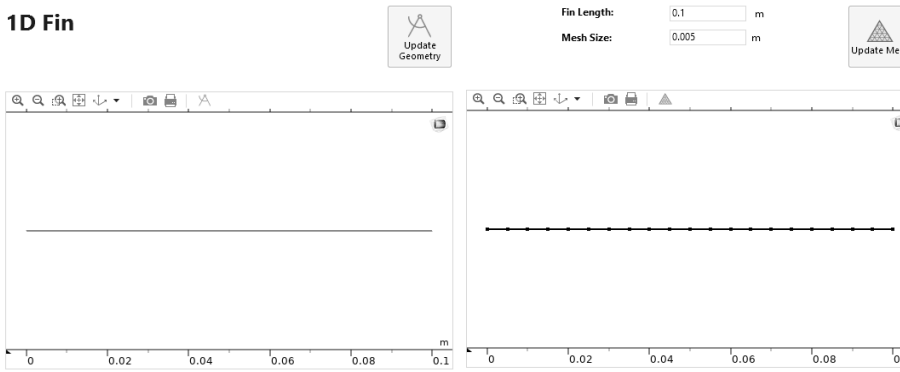


FIGURE 10.2. Creating 1D fin: (a) Geometry (left), (b) Mesh (right).

that the user can change the mesh size from within the application. In general, it is possible to choose a physics-suggested mesh and select the mesh size defaults (e.g., normal, fine, and coarse) to obtain a mesh grid that is defined based on the physics such as boundary conditions and geometrical configurations. This example corresponds to the 1D scenario presented in Section 9.1.1 ("One-Dimensional Fin Geometry").

10.3 CASE STUDY 2—GEOMETRY MODEL FOR 2D FIN

This example includes the 2D fin model in the FEM tool and the associated application (Figure 10.3). Similar to the 1D scenario, the custom mesh size setting capability is included. Different mesh element types are used for different regions of the part: the vertical base uses triangular elements while the horizontal fin uses rectangular elements. It is possible to add physics to the model and use the provided application to build upon by adding new features and graph windows. The user may choose to have different thicknesses for the fin left and right ends to observe how the nonuniform thickness throughout the fin length affects the mesh grid variation. If the auxiliary (partitioning) block is rotated (–2.4 degrees), different end thicknesses are obtained, and the resulting geometry would not be equivalent to the one resulting from partition by the horizontal symmetry axis in the original example. You can consider this case as a variation of the 2D scenario presented in Section 9.1.2 ("Two-Dimensional Fin Geometry").

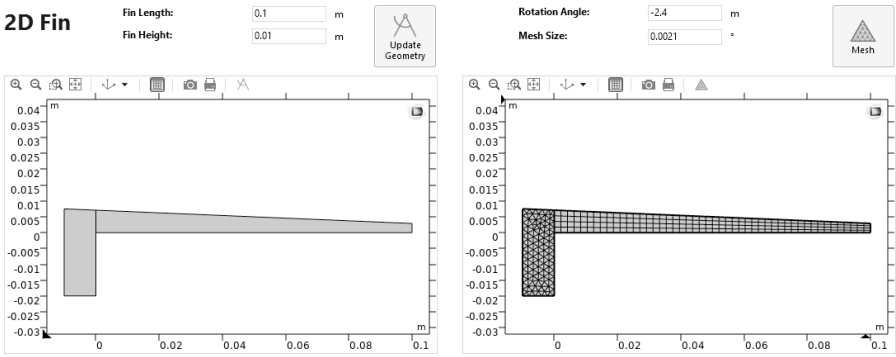


FIGURE 10.3. Creating 2D quadrilateral fin with constant thickness throughout the length: (a) Geometry (left), (b) Mesh (right).

10.4 CASE STUDY 3—GEOMETRY MODEL FOR 3D FIN

This example contains the 3D fin model in the FEM tool and the associated application (Figure 10.4). As in the 2D fin example, different element types are used: tetrahedral for the vertical block and hexahedral for the horizontal fin. A separate maximum element size setting is provided for each region. The user may wish to experiment with different mesh distributions. The approach will be discussed in the next example where a central channel is added to the geometry. The symmetry plane shown in Figure 9.14 was used to divide the model (fin and wall) into two equal volumes, with one half deleted. The user may add physics to the model and further develop the provided application.

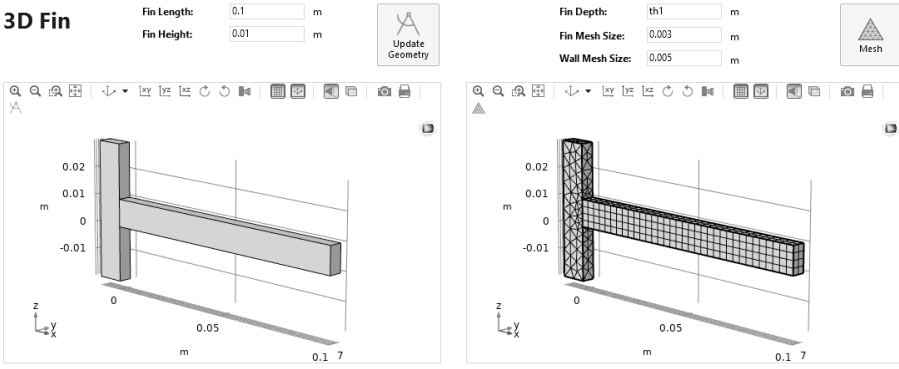


FIGURE 10.4. Creating 3D side-rectangular fin with rectangular cross section: (a) Geometry (left), (b) Mesh (right).

10.5 CASE STUDY 4—3D RECTANGULAR CROSS SECTION FIN WITH CENTRAL CHANNEL

This example contains the 3D fin model in the FEM tool and the associated application (Figure 10.5). Note that unlike in the previous fin without the central channel, the same element type (tetrahedral) is used throughout this geometry. A symmetry plane similar to that presented in Figure 9.21 was used to divide the geometry (fin and wall) into two equal volumes at the centerlines, with one half discarded. The user may add physics to the model and develop the application to include other features. In the vicinity of the boundaries separating the central channel from the rest of the solid part (fin), boundary elements are defined to ensure a finer mesh for the area surrounding the channel. These boundary elements would be needed if the channel were filled with fluid.

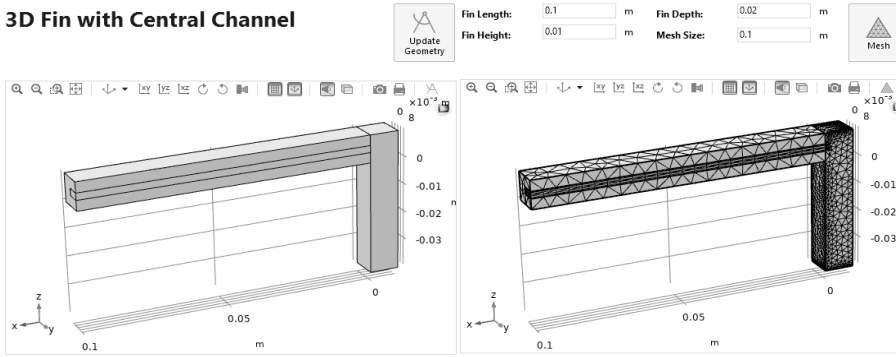


FIGURE 10.5. Creating 3D side-rectangular fin with central channel (rectangular cross sections): (a) Geometry (left), (b) Mesh (right).

10.6 CASE STUDY 5—3D CYLINDRICAL FIN

This example demonstrates geometry creation and meshing for the 3D cylindrical fin (circular cross section) by means of the FEM CAD tool as well as their applications (Figure 10.6) with and without central channels. The user may adjust the fin dimensions and mesh size as desired. Tetrahedral elements are used for all domains. The element grid size can be controlled in general; the size of the elements in the horizontal part (fin) is set to be proportional to those in the vertical part (wall).

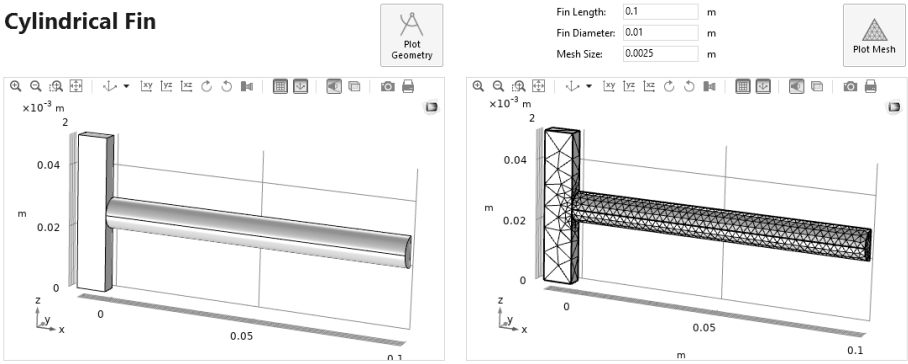


FIGURE 10.6. Creating 3D cylindrical fin: (a) Geometry (left), (b) Mesh (right).

The application for the cylindrical fin with a rectangular central channel is presented in (Figure 10.7). A block with a rectangular cross section was added to the original cylinder in order to represent the central channel.

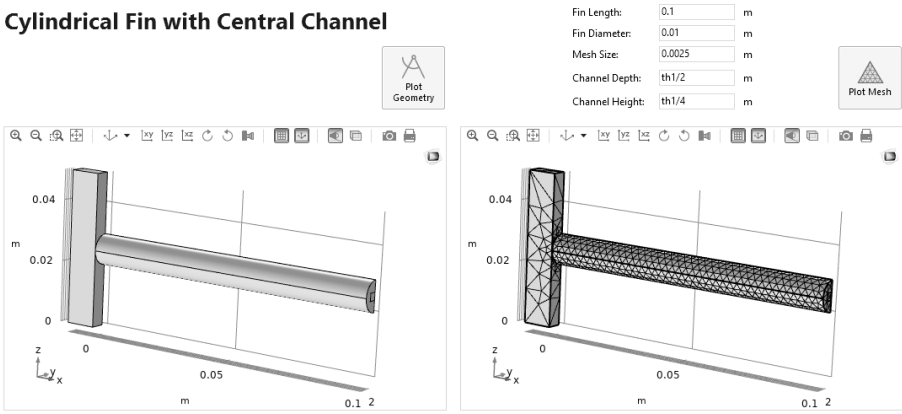


FIGURE 10.7. Creating 3D cylindrical fin with rectangular central channel: (a) Geometry (left), (b) Mesh (right).

The application for the cylindrical fin with finned central channel is presented in (Figure 10.8). This geometry is partially created using the CAD import tool. For the extended surface with a finned central channel, it is possible to set up the boundary element grid size in COMSOL Multiphysics *Model Builder*. This is recommended if the channel is filled with a moving or stationary fluid. A half of the geometry is shown after being divided by the vertical symmetry plane (Figure 9.42). Note that in this scenario, the only parameter controllable via the application interface is the mesh size.

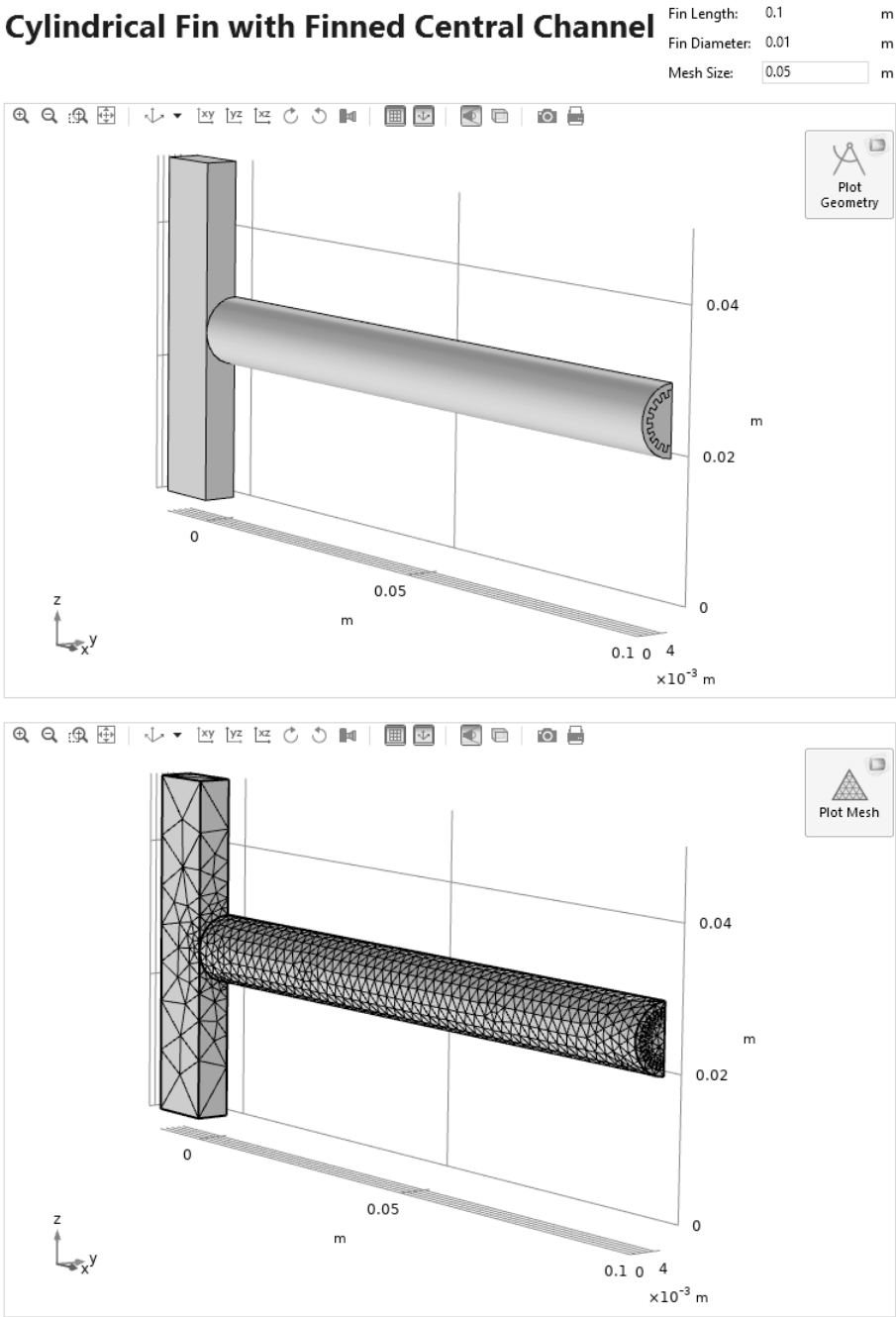


FIGURE 10.8. Creating 3D cylindrical fin with finned central channel: (a) Geometry (top), (b) Mesh (bottom).

10.7 CASE STUDY 8—3D RECTANGULAR FIN WITH TRIANGULAR CROSS SECTION

This example presents the FEM model of the 3D rectangular fin with a triangular cross section and the associated application (Figure 10.9). The vertical base is meshed with the tetrahedral elements while the horizontal fin meshing is performed with the hexahedral elements. No symmetry planes are applicable in this example. To create the uniformly spaced mesh in the horizontal direction along the fin, the user can use the swept mesh feature. This is done by first meshing one triangular end face of the fin and then sweeping this mesh along the length of the fin to the opposite triangular face. The size of the elements in the sweep direction is set to be proportional to the element size parameter.

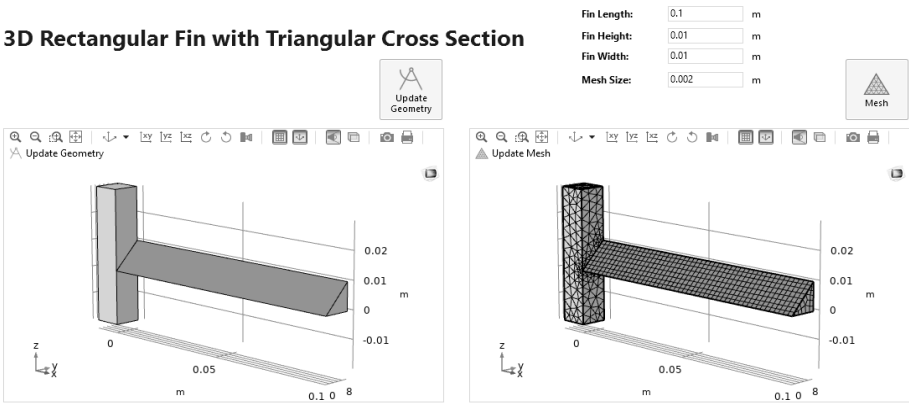


FIGURE 10.9. Creating 3D rectangular fin with triangular cross section: (a) Geometry (left), (b) Mesh (right).

10.8 CASE STUDY 9—3D FIN WITH RECTANGULAR CROSS SECTION AND TRIANGULAR SIDE PROFILE

This example presents the FEM model of the 3D rectangular cross section fin with a triangular side-profile and the associated application (Figure 10.10). The vertical base here is meshed with the tetrahedral elements while the horizontal fin is meshed with the hexahedral elements. The vertical *x*-*z* symmetry plane is applicable in this example. The user can change the fin dimensions and the mesh size as desired.

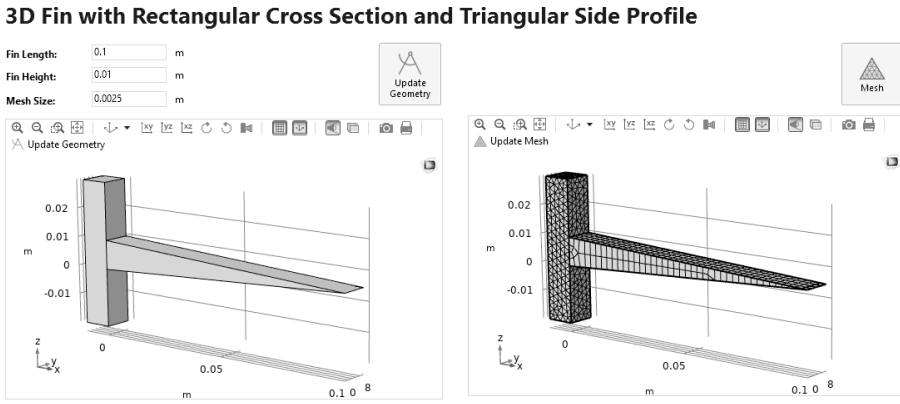


FIGURE 10.10. Creating 3D fin with rectangular cross section and triangular side profile: (a) Geometry (left), (b) Mesh (right).

10.9 CASE STUDY 10—3D FIN WITH RECTANGULAR CROSS SECTION AND CONCAVE SIDE PROFILE

This example includes the FEM model and the application (Figure 10.11) for the 3D fin with a rectangular cross section and concave side profile. The tetrahedral elements are used both for the vertical base and for the horizontal fin. A swept mesh feature was used to generate uniform horizontal spacing in the y-coordinate for the fin elements. The vertical x-z symmetry plane is applicable in this example. The user can vary fin dimensions and mesh size.

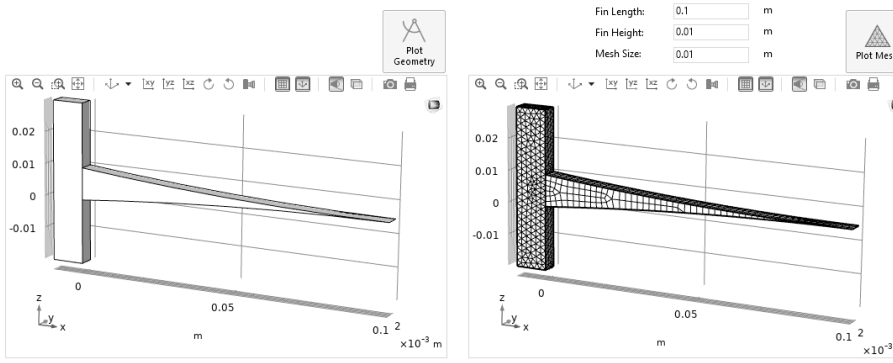


FIGURE 10.11. Creating 3D fin with rectangular cross section and concave side profile: (a) Geometry (left), (b) Mesh (right).

10.10 CASE STUDY 11—3D FIN WITH RECTANGULAR CROSS SECTION AND CONVEX SIDE PROFILE

This example includes the FEM model and the application (Figure 10.12) for the 3D fin with a rectangular cross section and convex side profile. Tetrahedral elements are used both for the vertical base and for the horizontal fin. The vertical *x*-*z* symmetry plane is applicable in this example; fin dimensions and mesh size can be adjusted.

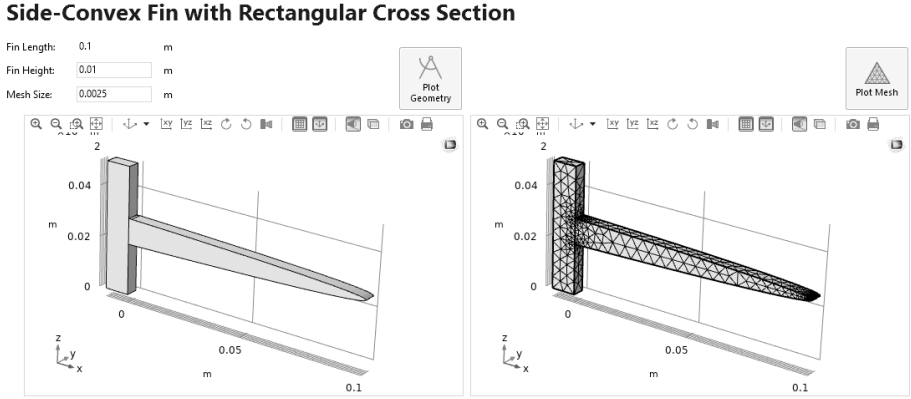


FIGURE 10.12. Creating 3D fin with rectangular cross section and convex side profile: (a) Geometry (left), (b) Mesh (right).

10.11 CASE STUDY 12—3D FIN WITH RECTANGULAR CROSS SECTION AND TRAPEZOIDAL-CONCAVE SIDE PROFILE

This example includes the FEM model and the application (Figure 10.13) for the 3D fin with a rectangular cross section and trapezoidal-concave side profile. Tetrahedral elements are used for the vertical base and hexahedral elements for the horizontal fin. A swept mesh feature was utilized to build uniform horizontal spacing for the fin elements. The vertical *x*-*z* symmetry plane is applicable in this example; fin dimensions and mesh size can be adjusted.

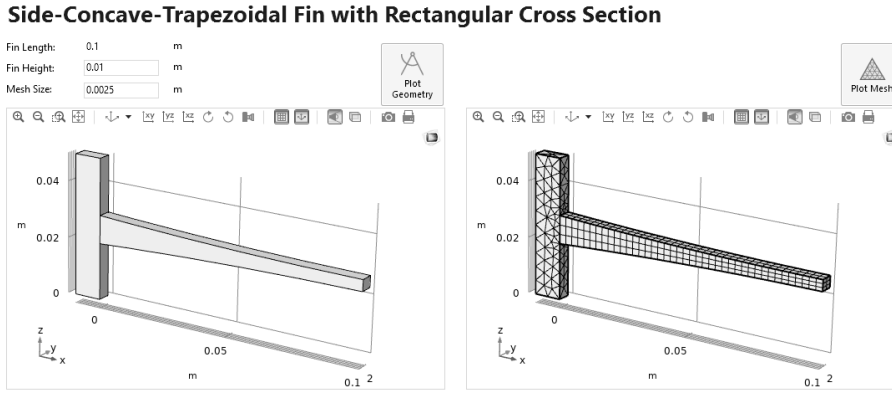


FIGURE 10.13. Creating 3D fin with rectangular cross section and trapezoidal-concave side profile: (a) Geometry (left), (b) Mesh (right).

10.12 CASE STUDY 13—3D PIN FIN WITH CIRCULAR CROSS SECTION

This example includes the FEM model and the application (Figure 10.14) for the 3D pin fin. Tetrahedral elements are used for the vertical base and hexahedral elements for the horizontal fin. A mapped mesh feature is employed to produce uniform horizontal spacing for the fin elements. This was then swept through the volume. The vertical *x*-*z* symmetry plane is applicable in this example; fin dimensions and mesh size can be adjusted.

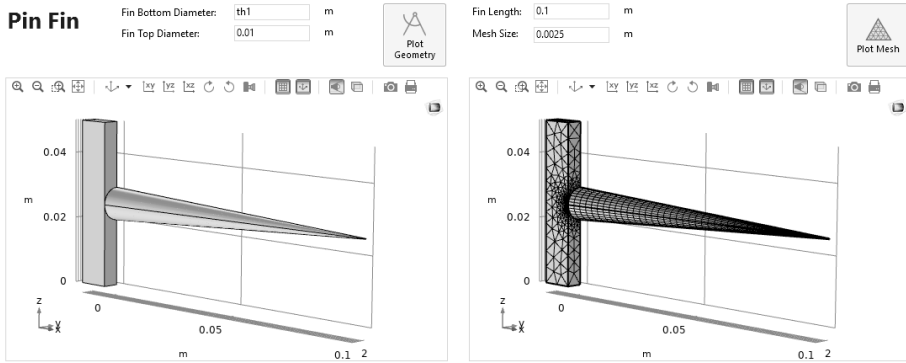


FIGURE 10.14. Creating 3D pin fin: (a) Geometry (left), (b) Mesh (right).

10.13 CASE STUDY 14—3D RADIAL FIN WITH HYPERBOLIC PROFILE

This example includes the FEM model and the application for the 3D radial fin with hyperbolic profile (Figure 10.15). Tetrahedral (vertical solid) and hexahedral (horizontal solid) elements are employed. The vertical *y-z* and *z-x* work planes form the symmetry planes. The user can vary fin dimensions and mesh size. The swept mesh feature was used to create uniform horizontal spacing for the fin elements.

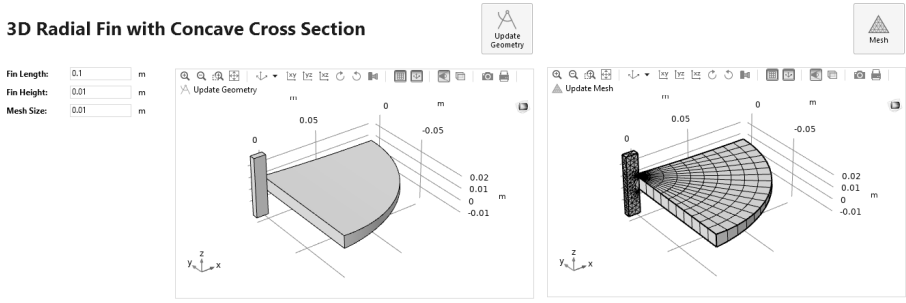


FIGURE 10.15. Creating 3D radial fin with hyperbolic profile: (a) Geometry (left), (b) Mesh (right).

10.14 CASE STUDY 15—3D WEBBED RADIAL FIN WITH HYPERBOLIC PROFILE

This example includes the FEM model and the application (Figure 10.16) for the 3D webbed radial fin with hyperbolic profile. The vertical *z-x* and *y-z* work planes form the symmetry planes. Tetrahedral mesh is used for this part—the vertical column (wall) is not included in the mesh. Mesh size is the only variable that can vary for this geometry.

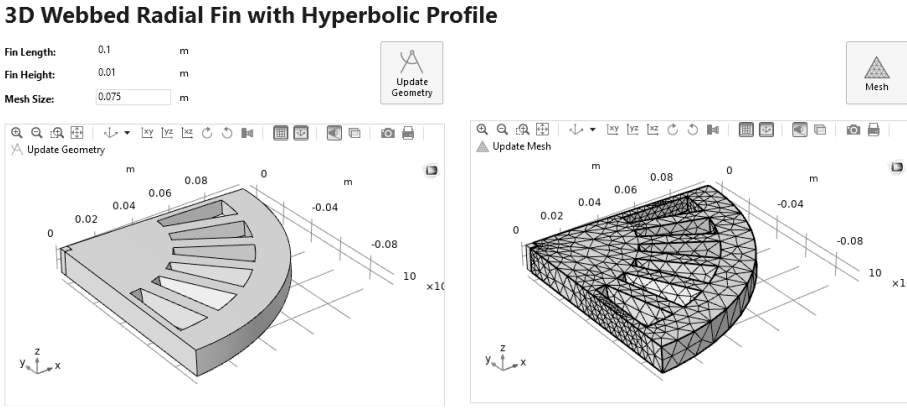


FIGURE 10.16. Creating 3D webbed radial fin with hyperbolic profile: (a) Geometry (left), (b) Mesh (right).

10.15 CASE STUDY 16—3D ROTINI FIN

This example includes the FEM model and the application (Figure 10.17) for the 3D rotini fin. Tetrahedral elements are used to mesh the entire fin geometry. The user can adjust mesh size and wall dimensions only (as the geometry of the fin was imported).

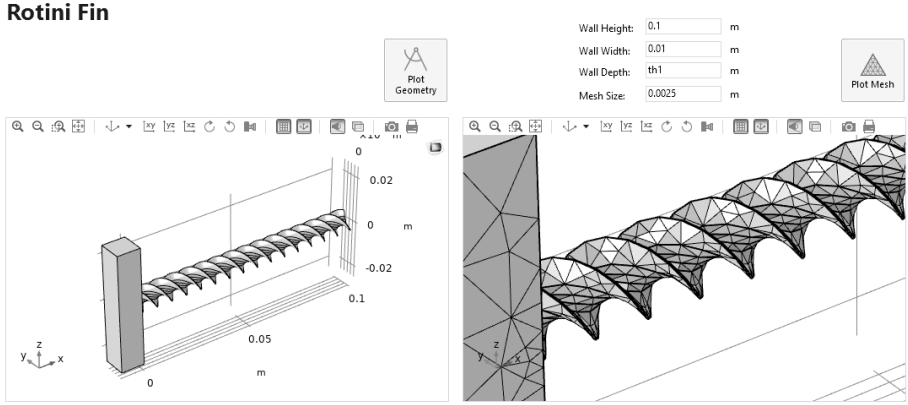


FIGURE 10.17. Creating 3D rotini fin: (a) Geometry (left), (b) Mesh (right).

CHAPTER 11

GOOD PRACTICES

The term *best practices* is a well-known expression in a variety of engineering disciplines in which the concept of product life-cycle, expressed by the acronym CDIO (Conceive, Design, Implement, and Operate), is used. When working on a model, analysis, or process of any kind, a variety of techniques may be employed and improved upon. Hence, the Author does not necessarily agree with the expression *best practices*; instead there are *good practices*, those which are more likely to lead to a useful outcome. Nevertheless, the challenge to improve processes and designs is always there to achieve better performance and to eliminate waste (i.e., *Muda*, a Japanese word for Futility, Uselessness, and Wastefulness) by focusing on critical-to-quality characteristics [182].

There is no preferred approach to design geometries; the process usually fits in three categories: (1) approach, (2) order, and (3) interface. The approach tells the story of the origin of the assembly or part, where and how it is created. The order informs the successive steps that have been taken for the geometry to be generated—if it is ordered (each step is the milestone for the next steps) or unordered (steps are independent of one another). The interface describes how the imported assembly relates to the host environment or one another. The host environment in these scenarios is usually FEM specialized tools while the originator can be either a CAD or FEM tool, or even a combination of both.

As the technology progresses and the hardware capabilities grow, the approach, order, and interface improve either by iterative updates of existing packages or by occasional introduction of completely new software. With this constant change, the concept of standardization

becomes even more important, since the cost associated with converting the geometries generated in the prior revisions of the specialized tools (FEM or CAD) may become prohibitive. Projects are delayed when the geometries created with an older version CAD tool cannot be easily translated to the one compatible with the new CAD tool (or vice versa)—the only acceptable version for a newly developed FEM tool. Note that the term *FEM* is used often in this work; however, it may be extended to other types of models where physics of any kind are investigated (e.g., Computational Fluid Dynamics—CFD). Although the community of the fields' specialists suggest workarounds—and the vendors attempt to introduce compatible products—the challenges remain both in terms of the human labor required and project delivery timeline constraints.

When creating designs, the element of creativity in some situations is as important as adhering to the known and tested techniques in others. If designers, engineers, doctors, and decorators were to follow old-fashioned knowledge and construction techniques, humans would be still living in the Flintstones version of cave homes. Even contemporary cave homes offer modern amenities, despite the primeval surroundings [183,184,185]. The following are just a few extraordinary examples showing what is possible when human imagination, planning, and impeccable execution join in a harmonious union (Table 11.1) [186,187,188,189]:

TABLE 11.1. Examples of innovative home architecture.

House Name	City	Country	Year
Tabātabāei House	Kashan	Iran	1880
Dune House	Atlantic Beach	United States	1964
Montaña Mágica Lodge	Huilo	Chile	1970
Freedom (Azadi) Tower	Tehran	Iran	1971
Wat Samphran Temple	Khlong Mai	Thailand	1985
Indira Gandhi Planetarium	Patna	India	1993
The Longaberger Headquarters	Newark	United States	1997
Nautilus House	Mexico City	Mexico	2006
Piano and Violin Building	Huainan	China	2007
Climate House®	Bremerhaven	Germany	2009
Sutyagin House	Arkhangelsk	Russia	2015

Thinking outside the construction square is the reason for exceptional creations at any time, in any region, and within any social strata. Independent thinking in an unfettered environment are prerequisites for this process. The kinds of resources and experiences available to

the creative worker in addition to their imagination have been critical to bringing to life radical innovations. It is the interconnectivity among diverse fields of study, such as art, engineering, and fashion, that has brought about revolutionary products—just think of the iPad in 2010.

Carefully review the geometrical data before and after importing the geometry into the analysis tool. This is particularly needed if the geometry imported into the analysis tool is scaled or the dimensions are not consistent with those of the analysis tool—for example, meters versus centimeters. One good approach for confirming correctness of the imported geometry dimensions is to carefully examine them with the measurement tool, which is normally available within any commercial analysis package (such as FEM or CFD). You can select regions of the geometry with known dimensions (e.g., length and surface area) where comparisons with known correct dimensions can be made.

You may be surprised to find out that there have been many cases, where the dimensions either were a multiple of the correct values or the units were not consistently used between the two tools. If you miss catching such errors, you usually find discrepancies when performing the analysis; however, that is not always the case, so do not count on it. Therefore, make sure that geometry inspection is a step with a set checkpoint. Since an analysis (model) is as good as its inputs and geometry makes the most important ingredient to an analysis (model), there is no geometry that is "Simple Enough" for this step to be neglected.

Variables are recommended if the geometry is created in the FEM tool or it is connected by a LiveLink, where the variables can be defined in FEM and linked to the geometry creation in the CAD tool. If the geometry is imported, no variable is normally used to define it in FEM. If you decide to use this method, select a variable name that is meaningful and that will help you to correctly recognize this variable. Working on the geometry is one of the preparatory tasks which you would appreciate later if done methodically, saving future wasted effort and headaches. In majority of cases, you can save the defined variables to a text or table in the form of an input file (e.g., a simple text file). Note that when importing variables (or parameters) using this method, the newly generated variables (or parameters) are appended at the end of the original variable (or parameter) table, meaning that they will be duplicated if they were already defined. In this case, the newly added parameters' colors are shown as red, which is a helpful tool for making the necessary revisions (e.g., deleting or changing the name of one of the parameters).

Another goal to keep in mind is the ability to recognize when to stop seeking perfection—at some point the law of diminishing returns comes into force and any benefit derived from further geometry refinement is

swamped by deficiencies elsewhere in the design, analysis, and manufacturing process. For example, imagine you have a geometry with a 5-mm fillet radii that you have managed to solve after a few hours of model debugging. Is it now worth it to change the fillet size to 6 mm to see if the smoother curvature will lead to a faster convergence and thus shorter solution time? Most likely, such a small change will not bring any appreciable benefit, and you should focus your energy on addressing other, more critical, issues. Instead, you may wish to take advantage of the defeaturing, repair, visual presentations, Boolean operations, or stitch features during or after importing a geometry. Some imperfections occasionally result in new findings. For example, if the geometry is complex in regions and because of that a small portion of it is causing solution errors in the CAD tool, perhaps you can delete that portion (if approved after careful investigation). This reduces the complexity of the geometry and also of the solution, resulting in saving time and computational resources.

The last note is that the designers should always try to think ahead in their creation process by asking themselves: What is happening next and one more step after that? What kind of accessibility features do I need to include in my design? Will I need to reinforce the structure? Do I need to incorporate redundancy systems to enhance safety? Do I need to check the historical data for lessons learned and compatibility of my chosen design with the environment? What are the steps to be taken to ensure a socially responsible, environmentally friendly, ethically compliant, and personally fulfilling project? What is happening after that? How is the creation being used? What would be the possible outcomes if the product were not employed as intended? Thus, the designers should apply a wide-angled outlook to their visionary projects.

This mode of thinking ahead may be familiar to those who had to defend their doctoral or master's thesis. For a thesis defense session, one has to be prepared to defend their model, including choice of physics, any simplifying assumptions, sensitivity analysis, input data accuracy, validation methods, and results presentation quality. So, while you are working away on your model and thinking "I'll just do this step here quickly and not worry about it for now," a few months later, when you are presenting your work and have forgotten about this little shortcut you took, it may prove to undermine your work. So, every time you make any response choice (positive, negative, or a maybe), remember that you may be called upon to defend it later [1,143].

CHAPTER 12

LEAN SIX SIGMA IMPLEMENTATION

A product or project lifecycle starts from the idea conception and ends at the operation stage. The objective of improving a project is to improve the bottom line, taking into account the historical data, current status, and its trend. The concept of improvement is incomprehensible if the base conditions, by which the improvements are to be judged, are unknown. Therefore, the two concepts of knowing the base conditions and the current coordinates should be kept in mind. If there is a noticeable deviation from the base conditions, the path is to be corrected—in air navigation science, this is also known as *drift correction*. The initial state of the project, design, or product are the starting point and the future of the product at time t is its destination.

The path that connects those two points, given the requirements and acceptable imperfections, is its trend. This path should be protected as if your life depended on it—no compromises should be made. It is *nonnegotiable*, you get it, you do not need permission, you only give *notice*! The effort you invested in achieving a certain objective is yours only, the investment you made to accomplish what you started is yours only, and therefore, you have all the rights to be able to evaluate your current status by measuring how far you have come from the initial point and where you are on the chosen path. This is why developing the methodology by which you can make this assessment as accurately and faithfully as possible is extremely important.

This is where the concept of Lean Six Sigma is useful. The related body of knowledge provides a set of tools and techniques to measure the process progress, quantify the deviations, and predict their effects on the process trend. Quality, time, and cost are the intertwined trio that are determining factors in deciding if you are happy where you are

with the process and product. There is room for improvement in most cases—and your guilt-free "discontent" is quite acceptable. Otherwise, you would not have been trained in the field of process improvement. Empirical statistical techniques are employed in order for both collected qualitative and quantitative data to be analyzed. The data analysis identifies the critical-to-quality characteristics and variables that are the input to your process model. This method also allows you to identify trivial variables and characteristics, the ones that have no place in your decision-making process.

Imagine going to a restaurant. There are generally a number of items on any checklists that should get a tick mark for the dining experience to be a good one, in which the customer can comfortably leave a positive comment for the business. It is not that the customer is always right as some may say (from a scientific point of view, it may be among the most *inane* comments one may proffer). However, it is an obvious breach of confidence if the host demonstrates poor attitude, food is clumsily presented (both in terms of appearance and temperature), and the atmosphere is less than relaxing. Any of these factors will make the experience a less enjoyable one [190].

This is an example of qualitative critical variables, where the variables determining your acceptable level of standard are determined qualitatively. This can be done by different means such as use of a Likert scale, where five-, seven-, or ten-point scales are used, including a number of responses from the "best" (e.g., strongly agree and extremely likely) to the "worst" (e.g., strongly disagree and extremely unlikely) scenarios. Note that the quoted terms *best* and *worst* are relative to the objective of the question and are not absolute—they represent extreme attitudes. For example, the best scenario for item service can be the host's highly professional attitude while the worst scenario for the cost can be the food's high price. The two *high*s therefore do not fall in the same category.

Losing or gaining weight is an example of a quantitative study based on critical variables. Assume that your current weight does not meet your standards; for example, you would like to trim the love handles to fit into your newly purchased ball gown. Your current weight is the baseline and your desired weight is the one based on your ideal Body Mass Index (BMI), which is a function of your height. You may choose to place yourself a little to the left of this recommended value (become lighter than the recommended weight), since you prefer a slenderer look. You decide that you need to accomplish this weight management within a month. Let us assume that you do not consider using surgery or extreme treatment techniques to achieve this ideal weight. You are then to follow a diet (exercise and food) to ensure that your BMI trend follows the

interpolated BMI you have projected from the initial start date to the end of the timeline. The weekends might be the times to check for process milestones—when you check your actual BMI against your planned values. If for whatever reason this trend has deviated from the set path, and the variation is significant, you will know that something went awry in this journey and you should make the correction as soon as possible to be able to fit into that ball gown within a month. After achieving this ideal weight (let us imagine in twenty days), you should ensure that you maintain your figure by continuing or adjusting your diet.

The same approach can be brought to engineering applications. The engineers or scientists decide upon a target value and strive to achieve it by: (1) recognizing that the improvement is necessary and feasible; (2) defining the areas in which improvements can and should be made; (3) measuring the maximum rate of return in each identified area; (4) analyzing how the changes influence the overall bottom line; (5) improving the methodology to introduce revisions by making educated decisions; (6) controlling the output by monitoring the process—adhering to the recognized best practices; (7) standardizing the processes and establishing new best practices; and (8) integrating the methodology throughout the process(es) or operation(s) by allocating appropriate resources—expertise, time, and funding. The product of this effort, in addition to the engineers and scientists obtaining a colored belt, is an improved relationship between cost, quality, and time achieved by removing unnecessary steps (wastage or redundancy)—the visible or hidden steps that add no value to the experience.

The main potential sources of waste are Transportation, Inventory, Material, Waiting, Over-production, Over-processing, Defects, and Skills (TIMWOODS). Being responsible citizens, engineers and scientists strive to reduce waste to the extent possible in order to respect (1) nature, (2) people or the surrounding world including the resources they indirectly interact with, and (3) the immediate environment. Value entitlement defines this interaction, which is merely a business transaction of some sort either in the form of services, products, experiences, and the responsibilities that individuals carry with respect to others, such as the entitlement to live in a peaceful, respectful environment, the *nonnegotiables.*

Lean Six Sigma defines quality as the state of the realization of the full value of entitlement in all aspects of business relationships. Entitlement is the right value of expectations, which takes the form of utility (form, fit, and function), access (volume, time, and location), and worth (economical, emotional, and intellectual). Entitlement is what you are entitled given available resources. It is the rightful level of expectations of every aspect of a business relationship. Realizing full value of

entitlement in all aspects of business relationships is vital. Entitlement capability versus actual capability defines the potential capability. Entitlement covers all aspects of life, from products to services to experiences. Think of the Cessna 172 aircraft you just rented. Imagine a case where you have planned for a cross-country trip to your grandmother's house that is 100 nautical miles away. Sometime soon after performing the takeoff routines you notice that the transponder does not operate as it should. You can neither ask for the flight following from your closest control tower nor travel over 25 nautical miles without a properly functioning transponder. Unless you did not think highly of the flying club, trusting that you were in their safe hands, were not taken advantage of, and being guarded by laws and consumer rights, you had no reason to believe that the flying club forgot to inform you of the inconvenient truth that the aircraft maintenance engineer had taken the transponder out to replace it, for there was a snag against it. This may have cost you your last conversation with your grandmother. In this case, your entitlement has been risked by the flying club and its associates and there are consequences associated with it for violating all three elements of utility (not functional), access (inconvenient time), and worth (emotional).

One way to implement Lean Six Sigma is to design smart experiments. *Smart* here is a double entendre, meaning both ingeniously designed tests as well as standing for Specific, Measurable, Attainable, Relevant, and Timely (SMART). For example, to design an experiment related to LED manufacturing, the engineers and scientists need to define what they are looking for, from conducting the experiments to presenting the result. They may come up with a variety of objectives such as studying the LED thermal-stress operation, heat-material characterization and interaction, and manufacturing aspects of the process.

Recall the example in which the total temperature change in a heat transfer problem was 100 °C and your numerical calculations deviated from the expected value by 1 °C. You may say "only" one percent error! "Wow . . . fantastic!" However, you will soon find that there are scenarios in which even this *remarkable* number is not good enough. For example, 99 percent correct performance equals twenty lost pieces of mail per hour and two bad landings per day by major airlines. On the other hand, there is no need to select a more complex geometry or a finer mesh over a coarse one if your objective is to provide techniques in which the part can be simplified using symmetrical features, or where sensitivity analyses suggest a simpler geometry would suffice.

To perform root cause analysis to identify error sources, the first step is to find these sources by different means such as brainstorming. The ideas generated by brainstorming, when you are questioning the process and methodology, can be organized by means of 5S methodology (Sort, Set in order, Shine, Sustain, and Standardize). The following is quoted

from the Author's work *Using COMSOL® in Heat Transfer Modeling from Slab to Radial Fin*—to show the importance of this technique [1].

> *Recall that first solo (your first solo flight) party; the evening you made fresh home-made cookies—with love and respect, carefully packed party supplies such as tea-cups, plates, cutleries, napkins, sugar bowl, and electric kettle in a blue cooler bag and drove to your evening [ground school] class. You made the freshly-brewed tea and let everybody help themselves with the goodies. When the party was over, you were left with cleaning up the party chaos. You started with: (1) sorting the items into their proper places—the extra cups were transferred to the blue bag, (2) setting items in order by inserting smaller cups in larger ones, (3) shining the glass bowl after washing it, (4) standardizing the steps by taking mental note of the observed pros and cons of organizing such a party, and (5) sustaining your [faint] smile after a long day.*

Step 2 helps you to organize the thoughts. Sorting steps can be organized more systematically by assigning the items to 5M baskets (i.e., Material, Machine, Method, Measurement, Man, and Money). Standardization, step 4, is a vital part in design, and the challenge of any activity is to ensure this item is fulfilled so that the improvements can be sustained. For this reason, design standards have been developed that include systems of measurement and acceptable tolerances. Implementing good practices is a systematic approach that should be applied when planning, executing, and reporting design-related tasks to comply with certification requirements.

After recognizing the parameters that may affect the process or product outcome, a number of tests may be conducted in which the critical ones among them may be selected for further review. Sensitivity analysis characterizes the rate at which the dependent variable changes as a function of the significant independent critical variables identified previously. Select the most important contributing items, the ones that make the most impact—the few critical to quality variables, and eliminate the rest—the trivial many. When designing experiments, create a table encompassing the critical variables and decide on the tests and the number of repeats. The rows of the table are associated with the experiments and the columns with critical process parameters. You may decide to run experiments for complete sets of variables along with their combinations. For example, for two and three sets of process parameters you can set up three and seven sets of experiments. The effect of each critical variable on the dependent variable can then be analyzed using regression analysis—a mathematical relationship that identifies the goodness of fit to data by statistical tools. Report key performance indices as the last step.

CHAPTER 13

CONCLUSION

All living beings express themselves by some conscious or subconscious form of geometrical patterns, the form of their daily creative art works such as the food they cook, the exercise they choose, the outfit they wear, the colors they select, and their interactions with nature. It is the truest form of expression in its forms, the science of shapes, positions, boundaries, patterns, and colors—what you see is what you get. There are no tricks or illusions associated with it. We find more meaning by the introduction of colors and patterns, helping to recall objects by their curves, colors, and other features. On a more practical level, mathematical interaction of human beings with their environment is a learned process, commencing in childhood and continuing through adulthood.

Human attraction to architectural shapes, such as the harmonious curves of the designs by Spanish architect Antoni Gaudí, is a testament to the universal human capability to interface, imagine, implement, and inspire—highlighting the importance of marrying the four concepts in everyday activities in order to create, innovate, and harmonize peacefully not only the beings but also the doings. The element of force has no place when setting up boundary conditions, and neither does psychological pressure. Repeated in a continuous fashion, with ill-intention, it results in all sorts of psychological drama that is not called for, with the adverse effects serving no purposes. Young and adult beings are to turn away from such exposure in the "schoolyards" and instead find solace in getting to know their beautiful surroundings, finding meaning in every moving, living, and perceivable object they interrelate with, seeking, and implementing knowledge in its ethical forms.

If one's interaction with the surrounding world were facilitated by means of building blocks such as Lego or Meccano, making architectural features such as houses, cities, firefighting machines, aircraft, and spaceships, one would know that the choice of the blocks as well as their colors are the primary and secondary reasons for selecting them among others. One would say to their play buddy, Robby, the doll: "I need a two-by-four brick to use for the window mantle, a four-by-twelve brick to make the bench on the porch, or a two-by-three brick sloped 45 degrees to use over the window to provide some shade." If one has taken care of the pieces by collecting them when not in use and setting them aside in a safe location away from their younger siblings, they may have ended up with a few of the bricks still decorating the chest of drawers, building a house in the middle of a large land with surrounding trees and flowers; the steps that connect the first level to the second one; and the shuttered windows that protect the house against any storms. One may also have placed people, a girl and a boy, simulating walking people, the owners, with their car parked somewhere on the driveway.

This is all made by means of geometrical features; the color is really the secondary issue when making such constructions, as one would rather use the four-by-twelve blue brick than the two-by-three yellow brick sloped 45 degrees to create the bench, even though one may find the latter a more cheerful color. Any activity involving interactions with the surrounding world, on a small or large scale, forms the foundation for future creativity and innovation as the child's intellectual capability increases.

Applications of geometry are seen in fields from decorative arts to performance-enhancing equipment to the ones that are directly related to the human respect for family roots and harvest such as property ownership and land survey. Owning a property is one of the ways for humans to express themselves, both in ancient and in modern times. Having large land property has long been a sign of status, wealth, and power. Therefore, the means to measure and thus define precisely this source of prosperity has always been the motivation to innovate the related technology. This application field remains the truest form of geometry, the name being rooted in the *Earth* and *measurement*.

Expressing oneself in different forms of art such as inlay is an example of employing geometrical shapes to decorate places. Here individual elements are placed into the depressions of a substrate (base object). The inserted materials may be wood, metal, precious stones, or ivory.

Not only does geometry affect us visually, but it also can be said to affect sounds, from birds to opera singers to musical instruments. Geometrical shapes are, of course, directly visible in the case of musical

instruments, but may also be there invisibly in the case of vocal cords of opera singers, for example. The expansions and contractions of the body muscles of a living being—directing more or less air through the openings—changing the coordinates of vertices, curves, and areas—constricting and dilating them, are what leads to the creation of aural beauty. Thus, a champion whistler uses the aerodynamics of the vocal cords by raising the tongue to constrict the air flow through the palate. The vortices generated by the air pattern inside the mouth, in combination with certain lips puckering up and adjusting the location of the tongue and jaw positions, can create a geometrical shape that directs the air in just the right way to produce that prize-winning whistle.

Introducing sound baffles to the process such as whistling through the teeth results in creating more complex sounds. The combination of the said techniques results in generating harmonious tunes that are pleasing to the ears. The generated sounds are therefore controlled for the range (how low and high they may go), pitch (frequency in which the sound is generated, in addition to embellishments in order to interpret musical tunes) [191]. Vocal operatic techniques have been developed in order for singers who do not use electronic amplification to take advantage of the natural ability to project the voice in order to produce the biggest effect on the audience—capturing the essence and the emotions of the story behind the aria and making it audible to the extent possible [192,193,194].

Geometry also plays a key role in the human's fascination with precious stones. Their natural appearance is much enhanced by applying different cuts (e.g., cabochon and faceted) and shapes (e.g., emerald, round, and square). Although one may assume that the larger the precious stone is, the more expensive it becomes, that is not necessarily the case. For example, although a round diamond may be smaller than a princess cut in terms of carats, its processing time is longer, given that the initial size may be the same for both cases, with more material wastage occurring in the latter scenario. The shape of the stone also affects the way it is perceived; the too deep or shallow ones are lifeless, since the light entering the piece either leaves the opposite or side facets of the incident surface; therefore, the light does not bounce back in different directions inside the piece, and is not reflected into the viewer's eyes. With a stone of the right depth, the incident light hits any of the side or opposite facets but then bounces back inside the stone and reflects to other facets—since the incidence angle exceeds the critical angle value—and eventually reaches the viewer's eyes. This is known as total internal reflection (a special case for Snell's law of refraction), which happens when light is traveling from a medium with a higher refractive index to one with a lower index. Diamonds have a very high

refractive index (2.42) compared to air (1). Due to this physical property, the incident light creates a greater number of internal reflections, resulting in its true brilliancy.

The ultimate goal is to enhance competency; therefore, human beings recognize and fulfill their true entitlement. There is a difference between potential and actual competence—which reflects the entitlement capability. This difference determines product quality. Contrary to the common perception that the world is getting more complex as time passes due to technological enhancements, people are the driving force—the ones who use their faculties to make the world a better place. Let us provide some examples of ethical leadership in the use of science and technology to address entitlement by improving the quality of life and through the responsible use of natural resources. Technology and capabilities in design fields are being continuously heightened; responsible design choices are invaluable. The benefits of making such constructions spread to both designers, individuals, and the environment.

Looking around us, there are numerous examples in which this brilliance of the human mind can be recognized—from the creation of vertical gardens as platforms for planting, working, and shading environments, providing residences to exotic plants and birds in order to improve quality of life to the development of LEED (Leadership in Energy and Environmental Design) projects to improve efficiency, health, and costs to achieve a sustainable environment. Designing, constructing, or retrofitting intelligent, efficient, eco-friendly, and sustainable buildings, parks, museums, and indoor rain forests to educate and give solace to the public in combination with waste management, water treatment, and use of renewable energy (e.g., planetary tides, waves, and wind) is a responsible use of natural resources [195,196,197,198, 199,200,201,202,203,204,205].

Examples of human ingenuity are everywhere. In *Incredibles* 2, the sequel to *The Incredibles* by Disney PIXAR Animation Studios, Elastigirl is Mr. Incredible's wife who steps up her game, a former jet pilot who is made of a material that can stretch to 1-mm thin and up to 100-ft tall in a variety of shapes—forming a parachute to a rubber shield and resuming its original state [206,207]. Such animated worlds are possible thanks to the latest software and hardware developments; and these days, we can not only enjoy them on the screens in front of us, but we can also be immersed in these creations by means of Virtual Reality (VR) technologies. Put on a VR headset and, as you are pushing the pedals of your stationary bike at home, you see yourself rolling along the winding roads of Provence, going through the twists and turns as the trees and the smiling spectators are passing by. You can now play golf even when there is a snowstorm outside and receive helpful

feedback after each stroke about how far the ball went and the ball's trajectory. You can now take to the virtual skies in a flight simulator exploring the breathtaking scenery of the Swiss Alps or go to virtual space and battle aliens as a commander of a spaceship. These are all possible through the power of simulation—the computer-generated models using shapes to recreate reality.

As the concluding remarks of this work are written, International Space Station (ISS) Expedition 59–60/Союз (meaning *union*) MS-12 is being launched, which is sending two American astronauts and a Russian cosmonaut to the ISS. They have been preparing for years for this event. As they got fitted into their spacesuits and spacecraft, interviewed, and walked to the vehicles at Star City in Moscow Oblast in Russia—the home to Valentina Vladimirovna Tereshkova—a Russian politician, engineer, textile assembly worker, pilot, and cosmonaut, the first woman in space and the only one who has been on a solo space mission when she flew the Vostok 6 in 1963—and Yuri Alekseyevich Gagarin—a Russian pilot and cosmonaut, the first human to fly in space in 1961, flying Vostok 1, both the Author's childhood heroine and hero—to be delivered to their spacecraft. They reached this point by displaying a high level of professionalism, showing all of us how much responsibility and care we must all strive for to reach these higher goals—one cannot get there if one takes it easy or "relaxes" on the job.

Engineering ethics principles require professional engineers who are working as designers to protect the public and to follow the guidelines that arose from human moral codes. Being part of this collective agreement, one promises to be faithful and true to the plans made, designs created, and words spoken. Think about this every time you use your power of creation. Geometrical patterns are used in everything around us, from places of worship to the clothing we wear, the food we eat, the spacesuit that protects us when oxygen is not available, the space we call home, and the emotional nudge our intellect faces when making decisions, the internal GPS.

The concept of *form*, *fit*, and *function* is applicable in almost everything where the practical aspect of design and creation are paramount. Being a practical designer does not necessarily mean being a pedantic bore but to be able to see the purpose of the creation (the way it is supposed to function) and design its shape (form) so that its subcomponents work in harmony with each other (fit). Forget about design "balance," as balance is an overstated concept serving no purpose. The resolution is harmony created through attention to the details, every nut and bolt—without them screaming "See me! I am alive." Engineering is a practical field with all the mightiness and power it offers to the world. With power comes responsibility and accountability—so use it well.

Making ethical decisions is part of living in this world, with the responsibility for executing it being deeply felt even more when *peer pressure* encourages the opposite. This is where Lean Six Sigma comes into play, the methodology to improve efficiency and effectiveness of activities while reducing wastage. As part of being human, if we may call ourselves as such, behaving more accountably, responsibly, and respectfully toward our surroundings—including natural resources and life in general—is the true creation purpose. Note the difference between *accountable* and *responsible.* The former is individual-based while the latter is collective-based. Being accountable makes the individual answerable to actions. Responsibility also relates to any consequences due to the actions before or after the action has taken place. Given the world's complexities, adhering to Lean Six Sigma rules is a lifestyle to be thought about and promoted.

APPENDIX A

MATLAB CODE FOR CREATING GOLDEN SPIRAL GEOMETRY

Table A.1 presents the parameters used to create the *golden spiral* approximated by a Fibonacci spiral (Figure 4.1). Table A.2 presents the geometry sequence used to create this design in COMSOL Multiphysics. The left portion of Table A.2 (five columns) lists the settings to create the eight rectangles, while the five right columns list the settings to create the eight arcs, made using circles. If reproducing this geometry, define the parameters listed and then, under the geometry node, enter the rectangles in sequence followed by the circles.

TABLE A.1. Parameters used to create a *golden spiral* approximation with Fibonacci spiral.

Parameter	Value
S1	21
S2	13
S3	8
S4	5
S5	3
S6	2
S7	1
S8	1

TABLE A.2. Geometry sequence for a *golden spiral* approximation with Fibonacci spiral in COMSOL Multiphysics.

Rectangle Label	Width	Height	Corner X	Corner Y	Circle Label	Radius	Center X	Center Y	Rotation (degree)
r1	S1	S1	0	0	c1	S1	S1	0	90
r2	S2	S2	S1	S1−S2	c2	S2	S1	S3	0
r3	S3	S3	S1+S4	0	c3	S3	S1+S4	S3	−90
r4	S4	S4	S1	0	c4	S4	S1+S4	S4	−180
r5	S5	S5	S1	S4	c5	S5	S1+S5	S4	90
r6	S6	S6	S1+S5	S4+S5−S6	c6	S6	S1+S5	S4+S8	0
r7	S7	S7	S1+S5+S6−S7	S4	c7	S7	S1+S5+S7	S4+S7	−90
r8	S8	S8	S1+S5	S4	c8	S8	S1+S5+S7	S4+S7	−180

BIBLIOGRAPHY

(1) Layla S. Mayboudi, *Heat Transfer Modelling Using COMSOL®: Slab to Radial Fin (Multiphysics Modeling Series)*, pp. 250, Mercury Learning and Information, 2018.

(2) Stefan R. Jongerius, David Lentink, "Structural Analysis of a Dragonfly Wing, Experimental Mechanics," doi: 10.1007/s11340-010-9411x, 2010.

(3) David Lentink, Stefan R. Jongerius, and Nancy L. Bradshaw, "The Scalable Design of Flapping Micro-Air Vehicles Inspired by Insect Flight, Flying Insects and Robots," pp. 185–205, Springer, Berlin Heidelberg, 2010; http://www.delfly.nl/publications/Jongerius%20and%20Lentink%20 2010%20Dragonfly.pdf

(4) https://en.oxforddictionaries.com/definition/geometry

(5) https://www.nasa.gov/mission_pages/apollo/missions/apollo8.html

(6) https://www.nasa.gov/audience/forstudents/k-4/stories/first-person-on-moon.html

(7) Jules Verne, Henri de Montaut (Illustrator), Louis Mercier (Translator), *From the Earth to the Moon*, pp. 338, Illustrated 1874 Edition: 100th Anniversary Collection, SeaWolf Press, 2019.

(8) https://sacredsites.com/sacred_places/sacred_geometry.html

(9) Parke Godwin (Author), Johann Wolfgang von Goethe (Creator), Johann Wolfgang Goethe, *The Autobiography of Goethe: Truth and Poetry, from My Own Life*, pp. 407, Palala Press, 2015.

(10) Markus Hattstein (Editor), Peter Delius (Editor), *Islam Art and Architecture*, pp. 624, h.f.ullmann publishing, 2011.

(11) https://digitalcollections.tcd.ie/home/index.php?DRIS_ ID=MS58_003v&#folder_id=14&pidtopage=MS58_130r&entry_ point=1

(12) https://commons.wikimedia.org/wiki/File:KellsFol034rChiRhoMonogram.jpg

(13) Glen W. Bowersock, *Mosaics as History: The Near East from Late Antiquity to Islam (Revealing Antiquity)*, pp. 160, First Edition, Belknap Press, 2006.

(14) https://commons.wikimedia.org/wiki/File:Roof_hafez_tomb.jpg

(15) Hamidreza Kazempour, *The Evolution of Muqarnas in Iran*, pp. 332, Supreme Century, 2016.

(16) CA TF Editores, *Alhambra and the Generalife: Official Guide*, pp. 350, T.F. Editores, S.L.C., 2011.

(17) https://commons.wikimedia.org/wiki/File:Isfahan_Lotfollah_mosque_ceiling_symmetric.jpg

(18) John T. Manning, "Age-advertisement and the Evolution of the Peacock's Train," Journal of Evolutionary Biology, vol. 2, no. 5, pp. 379–384, 2002; http://onlinelibrary.wiley.com/doi/10.1046/j.1420-9101.1989.2050379.x/pdf.

(19) https://en.oxforddictionaries.com/definition/geometry

(20) https://www.britannica.com/science/geometry

(21) Ted Chiang, *Stories of Your Life and Others*, pp. 336, First Edition, Tor Books, 2002.

(22) Brian M. Fagan, *Return to Babylon: Travelers, Archaeologists, and Monuments in Mesopotamia*, pp. 386, Revised Edition, University Press of Colorado, 2007.

(23) Mathieu Ossendrijver, *Babylonian Mathematical Astronomy: Procedure Texts (Sources and Studies in the History of Mathematics and Physical Sciences)*, pp. 618, Springer, 2012.

(24) Ramin Takloo-Bighash, *A Pythagorean Introduction to Number Theory: Right Triangles, Sums of Squares, and Arithmetic*, pp. 279, First Edition, Springer, 2018.

(25) Arnold Buffum Chace, Raymond Clare Archibald, Henry Parker Manning, *The Rhind Mathematical Papyrus, British Museum 10057 and 10058, V1: Free Translation and Commentary, and Bibliography of Egyptian Mathematics*, pp. 220, Literary Licensing, LLC, 2012.

(26) Marshall Clagett, *Ancient Egyptian Science: A Source Book. Volume Two: Calendars, Clocks, and Astronomy*, pp. 752, American Philosophical Society, 2004.

(27) Patricia F. O'Grady, *Thales of Miletus: The Beginnings of Western Science and Philosophy*, pp. 336, First Edition, Routledge, 2018.

(28) Clifford Herschel Moore, Orphism, *Pythagoreanism and The Mysteries*, pp. 42, Kessinger Publishing, LLC, 2010.

(29) John Dillon, *The Roots of Platonism: The Origins and Chief Features of a Philosophical Tradition*, pp. 116, Cambridge University Press, 2019.

(30) William Morison, *The Lyceum, Internet Encyclopedia of Philosophy*, 2016; https://www.iep.utm.edu/lyceum/.

(31) https://classicalwisdom.com/philosophy/cult-of-pythagoras/

(32) Antiphon (Author), Andocides (Author), K. J. Maidment (Translator), *Minor Attic Orators, Volume I: Antiphon, Andocides (Loeb Classical Library No. 308)*, pp. 608, Later Printing Edition, Harvard University Press, 1941.

(33) W. R. Knorr, *The Evolution of the Euclidean Elements: A Study of the Theory of Incommensurable Magnitudes and Its Significance for Early Greek Geometry*, pp. 396, Reprint of the Original Edition 1975, First Edition, Synthese Historical Library Softcover, 1980.

(34) Margaret E. Baron, *The Origins of the Infinitesimal Calculus*, pp. 304, Dover Publishing, 1987.

(35) Archimedes (Author), Sir Thomas Heath (Translator), *The Works of Archimedes*, pp. 576, Dover Publications, 2002.

(36) E. S. Kennedy, "An Islamic Computer for Planetary Latitudes," Journal of the American Oriental Society, vol. 71, no. 1, pp. 13–21, 1951.

(37) Barry Arthur Cipra, *Digits of Pi, What is Happening in the Mathematical Science*, pp. 28–39; http://www.ams.org/publicoutreach/math-history/hap-6-pi.pdf.

(38) Jonathan Borwein, Peter Borwein, "The Arithmetic-Geometric Mean and Fast Computation of Elementary Functions," Society for Industrial and Applied Mathematics (SIAM) Review, vol. 26, no. 3, pp. 351–365, 1984; www.cecm.sfu.ca/personal/pborwein/PAPERS/P26.pdf

(39) https://pi2e.ch/blog/2019/05/10/google-takes-over/

(40) https://www.nature.com/articles/s41567-019-0444-5

(41) https://www.britannica.com/topic/Sulba-Sutra

(42) Carl B. Boyer (Author), Uta C. Merzbach (contributor), *A History of Mathematics*, pp. 688, Third Edition, Jossey-Bass, 2011.

(43) by Euclid (Author), Dana Densmore (Editor), T.L. Heath (Translator), *Euclid's Elements*, pp. 529, Later Printing Edition, Later Printing Edition, Green Lion Press, 2002.

(44) Walter Eugene Clark (Translator), *The Aryabhatiya of Aryabhata: An Ancient Indian Work on Mathematics and Astronomy*, pp. 126, Kessinger Publishing, LLC, 2010

(45) Brahmagupta, *Algebra: With Arithmetic and Mensuration*, pp. 486, Ulan Press, 2011.

(46) Roshdi Rashed (Editor), *Al-Khwarizmi: The Beginnings of Algebra (History of Science and Philosophy in Classical Islam)*, pp. 392, Bilingual Edition, Saqi Books, 2010.

(47) Roshdi Rashed, *Athanase Papadopoulos, Menelaus' Spherics (Scientia Graeco-Arabica)*, pp. 874, First Edition, De Gruyter, 2017.

(48) François Charette, Ḥabash al-Ḥāsib: Abū Jaʿfar Aḥmad ibn ʿAbd Allāh al-Marwazī, In Thomas Hockey; et al. (eds.), "The Biographical Encyclopedia of Astronomers," pp. 455–7, Springer, 2007; https://islamsci.mcgill.ca/RASI/BEA/Habash_al-Hasib_BEA.pdf.

(49) Marco Panza, "The Role of Algebraic Inferences in Naʾīm Ibn Mūsā's Collection of Geometrical Propositions, Arabic Sciences and Philosophy," vol. 18, no. 2, pp. 165–191, 2008.

(50) https://www.britannica.com/place/Harran

(51) Behnaz Hashemipour, Būzjānī: Abū al-Wafāʾ Muḥammad ibn Muḥammad ibn Yaḥyā al-Būzjānī, In Thomas Hockey; et al. (eds.), "The Biographical Encyclopedia of Astronomers," pp. 188–189, Springer, 2007; https://islamsci.mcgill.ca/RASI/BEA/Buzjani_BEA.pdf.

(52) Farzaneh Ghaffari, Mohsen Naseri, Razieh Jafari Hajati, Arman Zargaran, "Rhazes, A Pioneer in Contribution to Trials in Medical Practice," AMHA - Acta Medico-Historica Adriatica, vol. 15, no. 2, pp. 261–270, 2017; https://www.researchgate.net/publication/322989401_Rhazes_a_pioneer_in_contribution_to_trials_in_medical_practice

(53) Abdelghani Tbakhi, Samir S. Amr, "Ibn Al-Haytham: Father of Modern Optics," Ann Saudi Medicine, vol. 27, no. 6, PMC6074172, pp. 464–467, 2007; https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6074172/.

(54) Emilia Calvo, "Ibn Muʿādh: Abū ʿAbd Allāh Muḥammad ibn Muʿādh al-Jayyānī," In Thomas Hockey; et al. (eds.), The Biographical Encyclopedia of Astronomers, pp. 562–3, Springer, 2017; https://islamsci.mcgill.ca/RASI/BEA/Ibn_Muadh_BEA.htm.

(55) Behzad Hashemipour, "Khayyām: Ghiyāth al-Dīn Abū al-Fatḥ ʿUmar ibn Ibrāhīm al-Khayyāmī al-Nīshāpūrī," In Thomas Hockey; et al. (eds.), The Biographical Encyclopedia of Astronomers, pp. 627–628, springer, 2007; https://islamsci.mcgill.ca/RASI/BEA/Khayyam_BEA.htm.

(56) Emilia Calvo, "Khāzin: Abū Jaʿfar Muḥammad ibn al-Ḥusayn al-Khāzin al-Khurāsānī," In Thomas Hockey, et al. (ed.), The Biographical Encyclopedia of Astronomers, pp. 628–629, Springer, 2007; https://islamsci.mcgill.ca/RASI/BEA/Khazin_BEA.htm.

(57) Ptolemy (Author), Gerald J. Toomer (Translator), Owen Gingerich (Foreword), *Ptolemy's Almagest*, pp. 712, Revised Edition, Princeton University Press, 1998; https://isidore.co/calibre/get/pdf/Ptolemy%26%2339%3Bs%20Almagest%20-%20Ptolemy%2C%20Claudius%20%26amp%3B%20Toomer%2C%20G.%20J__5114.pdf

(58) Gerald J. Toomer, "Ptolemy (Claudius Ptolemæus)," In Gillispie, Charles (ed.), "Dictionary of Scientific Biography," Scribner and American Council of Learned Societies, vol. 11, pp. 186–206, 1970; http://www.u.arizona.edu/~aversa/scholastic/Dictionary%20of%20Scientific%20Biography/Ptolemy%20(Toomer).pdf.

(59) Sayeh Meisami, *Nasir al-Din Tusi: A Philosopher for All Seasons*, pp. 128, Islamic Texts Society, 2019.

(60) https://www.britannica.com/biography/Menelaus-of-Alexandria

(61) Joe Rosen, Lisa Quinn Gothard, *Encyclopedia of Physical Science Volume 1 and 2*, Infobase Publishing, pp. 416, Facts on File, 2009.

(62) Alex Gillespie, "Descartes' Demon: A Dialogical Analysis of Meditations on First Philosophy," Theory and Psychology, vol. 16, pp. 761–781, 2006; https://web.archive.org/web/20100618050955/http://stir.academia.edu/documents/0011/0112/Gillespie_Descartes_demon_a_dialogical_analysis_of_meditations_on_first_philosophy.pdf.

(63) Gerd Faltings, "The Proof of Fermat's Last Theorem by R. Taylor and A. Wiles," Notices of the American Mathematical Society, vol. 42, no. 7, pp. 743–746, 1995; https://www.ams.org/notices/199507/faltings.pdf.

(64) Tamer M. Rudavsky, "Gersonides: Levi ben Gerson," In Thomas Hockey; et al. (eds.), The Biographical Encyclopedia of Astronomers, pp. 415–417, Springer, 2007; https://islamsci.mcgill.ca/RASI/BEA/Gersonides_BEA.htm.

(65) Richard Crangle, "A Quite Rare Entertainment: An Optical Show in Paris in 1656," New Magic Lantern Journal, pp. 76–78, vol. 9, no. 5, 2003; http://www.magiclantern.org.uk/new-magic-lantern-journal/pdfs/4008787a.pdf

(66) Niccolò Guicciardini, "John Wallis as Editor of Newton's Mathematical Work," Notes and Records of the Royal Society of London, vol. 66, no. 1, pp. 3–17, 2017.

(67) Clark Kimberling, "Conics Associated with a Cevian Nest," Forum Geometricorum, vol. 1, pp. 141–150, 2001; http://forumgeom.fau.edu/FG2001volume1/FG200121.pdf.

(68) https://www.britannica.com/biography/Giovanni-Ceva

(69) https://www.britannica.com/biography/Tommaso-Ceva

(70) Arnold Emch, Edited and Translated by G. B. Halsted, "Review of Giralamo Saccheri's Euclides Vindicatus," Bulletin of American Mathematical Society, vol. 28, no. 3, pp. 131–132; 1922; http://www.ams.org/journals/bull/1922-28-03/S0002-9904-1922-03514-8/S0002-9904-1922-03514-8.pdf.

(71) https://www.britannica.com/biography/Nikolay-Ivanovich-Lobachevsky

(72) Stephen M. Stigler, *The History of Statistics: The Measurement of Uncertainty before 1900*, pp. 432, Reprint Edition, Belknap Press: An Imprint of Harvard University Press, 1990.

(73) https://www.britannica.com/biography/Farkas-Bolyai

(74) N. Bourbaki (Author), P. Spain (Translator), *Functions of a Real Variable*, pp. 350, Springer, 2004.

(75) Tamás Dénes, "Real Face of János Bolyai," Notices of the American Mathematical Society, vol. 58, no. 1, pp. 41–51, 2011; http://www.ams.org/notices/201101/rtx110100041p.pdf.

(76) "The Singular Value Decomposition in Symmetric (Lowdin) Orthogonalization and Data Compression;" http://www.wou.edu/~beavers/Talks/Willamette1106.pdf.

(77) John Stillwell, *Sources of Hyperbolic Geometry (History of Mathematics, V. 10)*, pp. 153, American Mathematical Society, 1996.

(78) https://www.britannica.com/biography/Gregorio-Ricci-Curbastro

(79) Pietro Nastasia, Rossana Tazzioli, "Toward a Scientific and Personal Biography of Tullio Levi-Civita (1873–1941)," Historia Mathematica, vol. 32, no. 2, pp. 203–236, 2005; https://www.sciencedirect.com/science/article/pii/S0315086004000229.

(80) Jeremy Gray, *A History of Abstract Algebra: from Algebraic Equations to Modern Algebra*, pp. 440, First Edition, Springer, 2018.

(81) https://www.britannica.com/science/Riemannian-geometry

(82) https://www.britannica.com/biography/Felix-Klein

(83) https://www.britannica.com/biography/Bernhard-Riemann

(84) https://www.britannica.com/biography/Ferdinand-von-Lindemann

(85) Henri Poincare, *The Value of Science: Essential Writings of Henri Poincare*, pp. 608, First Edition, Modern Library, 2001.

(86) Cao Huai-Dong, Zhu Xi-Ping, "Complete Proof of Poincare and Geometrization Conjectures- Application of the Hamilton Perlman Theory of the Ricci Flow," ASIAN Journal of Mathematics, vol. 10, no. 2, pp. 165–492, 2006; http://www.ims.cuhk.edu.hk/~ajm/vol10/10_2.pdf

(87) Wilbur R. Knorr, "Archimedes and the Spirals: The Heuristic Background," Historia Mathematica, vol. 5, no. 1, pp. 43–75, 1978.

(88) Duncan M. Y. Sommerville, *Analytical Geometry of Three Dimensions*, Cambridge University Press, 1959.

(89) https://www.britannica.com/technology/groma

(90) Brian Sturman, Alan Wright, "The History of Tellurometer," pp. 11, 2008; www.fig.net/resources/proceedings/fig_proceedings/fig2008/papers/hs01/hs01_03_sturman_wright_2833.pdf.

(91) A. Johnson, A. Pimpinelli, "Pegs and Ropes: Geometry at Stonehenge, Research Laboratory for Archaeology and the History of Art," University of Oxford, Oxford, pp. 1–10, 2015. https://houseoftruth.education/files/pdf/Pegs_and_Ropes_Geometry_at_Stonehenge.pdf

(92) https://en.wikipedia.org/wiki/Abel_Foullon

(93) https://www.britannica.com/biography/Leonard-Digges

(94) Jon Bedford, *Traversing the Past: The Total Station Theodolite in Archaeological Landscape Survey*, pp. 58, Historic England, 2016.

(95) Liesbeth Cornelia de Wreede, L.C., Utrecht University Repository, *Willebrord Snellius (1580–1626): A Humanist Reshaping the mathematical Sciences*, Utrecht University, 1974; https://dspace.library.uu.nl/handle/1874/22992.

(96) https://www.britannica.com/biography/Edmund-Gunter

(97) https://en.wikipedia.org/wiki/Jonathan_Sisson

(98) https://www.britannica.com/biography/Jesse-Ramsden

(99) Jacques Cassini, *Histoire De L'académie Royale Des Sciences, Année ... Avec Les Mémoires De Mathématique et De Physique, Pour La Même Année: Tirez Des Registres De Cette Académie*, pp. 660, French Edition, Nabu Press, 2010.

(100) Cesar-Francois Cassini De Thury, *Description Geometrique De La France (1783)*, pp. 212, Kessinger Publishing, LLC, 2010.

(101) https://en.wikipedia.org/wiki/William_Roy

(102) Ordnance Survey, Tim Owen, *Map Makers to Britain Since 1791*, pp. 196, Ordnance Survey, 1992.

(103) William Lambton, "An account of the Trigonometrical Operations in Crossing the Peninsula of India, and Connecting Fort St. George with Mangalore," Asiatic Researches or Transactions of the Society Instituted in Bengal for Enquiring into the History and Antiquities, pp. 290–384, 1811; https://archive.org/stream/asiaticresearche10asia#page/n307/mode/2up.

(104) https://www.britannica.com/biography/George-Everest

(105) http://www.wnonline.co.za/article.php?id=85

(106) G.R. Bozzoli, *Forging Ahead*, pp. 120, Witwatersrand University Press, 1997.

(107) Barry Kavanagh, Tom Mastin, *Surveying: Principles and Applications*, pp. 592, Ninth Edition, Pearson, 2013.

(108) https://en.wikipedia.org/wiki/Rangefinder

(109) https://www.intersense.com/

(110) United States of America Department of Defense, *Global Positioning System Standard Positioning Service Performance Standard*, pp. 160, Fourth Edition, 2008.

(111) Eric Gakstatter, *RTK Networks – What, Why, Where*, Presented September 22, 2009, USSLS/CGSIC Meeting 2009; https://www.gps.gov/cgsic/meetings/2009/gakstatter1.pdf.

(112) NOAA Coastal Services Center, National Oceanic and Atmospheric Administration (NOAA) Coastal Services Center, *Lidar 101: An Introduction to Lidar Technology, Data, and Applications*, pp. 72, Revised, Charleston, SC, 2012; https://coast.noaa.gov/data/digitalcoast/pdf/lidar-101.pdf.

(113) The International Civil Aviation Organization (ICAO), *ICAO's Circular 328 AN/190, Unmanned Aircraft Systems (UAS)*, pp. 54, 2011; https://www.icao.int/Meetings/UAS/Documents/Circular%20328_en.pdf.

(114) http://markgolding.co.uk/a-history-of-geometry

(115) John Wilkins, *An Essay Towards a Real Character, and a Philosophical Language*, pp. 660, Printed for SA: Gillibrand, and for John Martin Printer to the ROYAL SOCIETY, 1668.

(116) Mike Goldsmith, *Good Practice Guide No. 118, A Beginner's Guide to Measurement*, pp. 17, National Physical Laboratory, 2017; https://www.npl.co.uk/special-pages/guides/gpg118_begguide2measure.

(117) https://www.britannica.com/science/furlong

(118) Simon Stevin, Edited by Ernst Crone, E. J. Dijksterhuis, R. J. Forbes, M. G. j. Minnaert, a. Pannekoek, *The Principal Works of Simon Stevin*, pp. 625, 1955; http://www.dwc.knaw.nl/pub/bronnen/Simon_Stevin-%5BI%5D_The_Principal_Works_of_Simon_Stevin,_Mechanics.pdf.

(119) https://www.britannica.com/biography/Bartholomeo-Pitiscus

(120) Leslie Stephen, Henry Davis, *Dictionary of National Biography, Edited by Leslie Stephen, vol. 1–21 (By Leslie Stephen and Sidney Lee), vol. 22–26 (By Sidney Lee), vol. 27–63 (with Supplementary volumes), Vol. XLIX*, pp. 518, Historical Print Editions, British Library, 2011.

(121) https://www.britannica.com/biography/Gabriel-Mouton

(122) Jon Meacham, *Thomas Jefferson: The Art of Power*, pp. 800, First Edition, Random House, 2012.

(123) A Chronological History of the Modern Metric System (to 2008), pp. 142, 2008; http://www.metricationmatters.com/docs/MetricationTimeline.pdf.

(124) http://hyperphysics.phy-astr.gsu.edu/hbase/Particles/atomsiz.html

(125) https://www.physicsforums.com/threads/water-oh-bond-length.300007/

(126) https://en.wikipedia.org/wiki/Carbon_dioxide

(127) https://en.wikipedia.org/wiki/Abdomen#/media/File:Scheme_ant_worker_anatomy-en.svg

(128) Roy E. Ritzmanna, Roger D. Quinnb, Martin S. Fischerc, *Convergent Evolution and Locomotion through Complex Terrain by Insects, Vertebrates and Robots*, Arthropod Structure and Development, vol. 33, pp. 361–379, 2004.

(129) https://www.britannica.com/biography/Antoni-Gaudi

(130) https://oldmachinepress.com/2012/10/05/chrysler-a57-multibank-tank-engine/

(131) https://www.britannica.com/technology/Sherman-tank

(132) Shell Global, *Avgas Facts and Future, The Future of General Aviation Fuels*; https://www.shell.com/business-customers/aviation/aeroshell/knowledge-centre/technical-talk/techart12-30071515.html.

(133) https://www.quora.com/How-much-fuel-does-a-Boeing-747-or-777-consume-per-hour-while-in-flight-and-what-is-the-approximate-cost-of-that-fuel-for-the-airline

(134) https://3dprint.com/222268/bmw-3d-printed-roof-bracket/

(135) https://www.bmw.com/en/innovation/3d-print.html

(136) https://cessna.txtav.com/en/turboprop/denali

(137) https://www.geaviation.com/bga/engines/ge-catalyst

(138) https://www.ge.com/additive/blog/new-manufacturing-milestone-30000-additive-fuel-nozzles

(139) https://money.cnn.com/2017/04/07/technology/adidas-3d-printed-shoe/index.html

(140) http://www.hoet-optiek.be/en/

(141) George Orwell, *Animal Farm*, pp. 128, International Edition, Penguin Classic, 2013.

(142) https://www.britannica.com/biography/Paul-Dirac

(143) Layla S. Mayboudi, *Flight Science Mathematics Techniques Sensibility*, pp. 490, Mercury Learning and Information, 2019.

(144) https://en.wikipedia.org/wiki/Flat_Earth#Belief_in_flat_Earth

(145) http://www.in2013dollars.com/uk/inflation/1870?amount=500

(146) https://www.britannica.com/biography/Eratosthenes

(147) http://www.spiritualnorth.com/blog/flat-earth-phd-hoax-timeline-and-summary

(148) https://www.britannica.com/biography/Alexis-Claude-Clairaut

(149) Richard H. Rapp, Lecture Notes, Papers, Publications and Reports (School of Earth Sciences), *Geometric Geodesy Part I*, Ohio State University Department of Geodetic Science and Surveying, 1991; https://kb.osu.edu/handle/1811/24333.

(150) Persian, also known as Farsi, belongs to the Indo-European language family. It is a language spoken in countries that were Persianate (e.g., Iran, Afghanistan, Tajikistan), which were once part of greater Iran, by over 110 million people. The name *Iran* (also known as Persia) is pronounced as ɪˈrɑːn, and not "I Ran". The name Iran was adopted in 1935, and means noble people; the country was known as Persia until then. It is bordered by the Caspian Sea on the north and by the Persian Gulf on the south, with a population over 79,100,000 (as of 2015). The country was a monarchy until 1979 when it was renamed the Islamic Republic after a popular uprising. Iran's language is Farsi (Persian) and the letters are written in the Persian alphabet that is rooted in Phoenician alphabet, the oldest verified alphabet. It is an abjad style of writing, meaning that each letter is a consonant and the reader is to add the vowels. This 22-letter Phoenician alphabet was expanded to 32 letters into the current Farsi alphabet. Although some Farsi letters are similar in writing to Arabic letters, the languages are different, as Arabic is a Semitic language versus Persian (Farsi), which is an Indo-European language. Persian has had a lexical influence on other languages such as Armenian, Turkic, and Indo-Aryan languages, for it has represented a prestigious cultural language in regions such as Western, Central, and Southern Asia. Although it has influenced Arabic language to some extent, new words were borrowed from Arabic by the Persians (Iranians) after the Arab Conquest of Iran. Due to its ancient historical significance, the Persians (Iranians)

were able to resist the Arabic language dominance in reading and writing. Farsi (Persian) was strengthened by fields such as literature, art, poetry, music, science, and mathematics through the works of Persian (Iranian) scholars (e.g., *Shahnameh* by Ferdowsi, *Rubaiyat* by Omar Khayyam, *Haft Peykar*—the inspiration for the *Turandot* opera by Puccini—by Nizami Ganjavi, *Gulistan* by Sadi—the decorator of the United Nations building door head) in addition to the polymaths and physicians (e.g., Avicenna), mathematicians and astronomers (e.g., Khwarizmi), and other polymaths and scientists who were the Renaissance people of their time. https://www.britannica.com/topic/Persian.

(151) https://www.spatial.com/products/3d-acis-modeling

(152) Dag Bergsjö, *Product Lifecycle Management – Architectural and Organisational Perspectives*, pp. 81, Chalmers University of Technology, 2009.

(153) https://www.solidworks.com/

(154) https://www.mathworks.com/products/matlab.html

(155) https://www.mathworks.com/products/simulink.html

(156) Alain Deutsch, "Static Verification of Dynamic Properties," PolySpace Technologies, pp. 1–8, 2003; https://web.archive.org/web/20120313084616/http://nesl.ee.ucla.edu/courses/ee202a/2005f/papers/Static_Verification.pdf.

(157) https://www.mathworks.com/products/simevents.html

(158) https://solidedge.siemens.com/en/

(159) https://www.mathworks.com/products.html

(160) https://www.mathworks.com/help/pde/ug/three-ways-to-create-2-d-geometry.html

(161) https://www.mathworks.com/help/pde/ug/pde.pdemodel.geometryfrommesh.html?searchHighlight=%22create%20geometry%22&s_tid=doc_srchtitle

(162) https://www.autodesk.com/products/inventor/overview

(163) https://www.autodesk.com/products/revit/overview

(164) https://www.ptc.com/en/products/cad/creo

(165) https://www.3ds.com/products-services/catia/products/

(166) http://www.cadam.com/

(167) https://www.3ds.com/products-services/catia/

(168) https://www.plm.automation.siemens.com/global/en/products/nx/

(169) https://www.engineering.com/PLMERP/ArticleID/7438/Inside-Daimler-Mercedes-Switch-from-Dassault-Systemes-to-Siemens-PLM-and-NX.aspx

(170) https://www.forbes.com/sites/antonyleather/2018/06/05/intels-new-28-core-monster-5ghz-desktop-processor-is-most-powerful-ever/#7f7db49f4ba3

(171) https://en.wikipedia.org/wiki/List_of_finite_element_software_packages

(172) https://www.comsol.com/products

(173) https://www.comsol.com/community/

(174) K. N. Shukla, "Heat Pipe for Aerospace Applications—An Overview, Journal of Electronics Cooling and Thermal Control," vol. 5, pp. 1–14, 2015; http://www.scirp.org/journal/jectc. http://dx.doi.org/10.4236/jectc2015.51001.

(175) https://www.britannica.com/topic/Millau-Bridge

(176) https://www.britannica.com/technology/microelectromechanical-system

(177) Ghenadii Korotcenkov, "Chemical Sensors: Comprehensive Sensor Technologies," Chemical Sensors Applications (Sensors Technology Series), vol 6, pp. 300, UK Edition, Momentum Press, 2011.

(178) Tony Hadland (Author), Hans-Erhard Lessing (Author), Nicholas Clayton (Contributor), Gary W. Sanderson (Contributor), *Bicycle Design: An Illustrated History (The MIT Press)*, pp. 584, 2016.

(179) https://www.britannica.com/biography/Hugo-Junkers

(180) Steven G. Noyce, Richard R. Vanfleet, Harold G. Craighead, Robert C. Davis, "High Surface-Area Carbon Microcantilevers," Nanoscale Advances, vol. 1, no. 3, pp. 1148–1154, 2019; https://pubs.rsc.org/en/content/articlehtml/2019/na/c8na00101d.

(181) https://pastafits.org/pasta-shapes/

(182) Lean Six Sigma approach focuses on enhancing bottomline by improving performance and eliminating waste (i.e., *Muda*; a Japanese word for *Futility*, *Uselessness*, and *Wastefulness*), focusing on critical to quality characteristics. The training for Lean Six Sigma is provided through the belt-based training system. The belt personnel are designated as white, yellow, green, and black belts, similar to judo.

(183) https://www.bing.com/images/search?q=cave+homes&qpvt=cave+homes&FORM=IGRE

(184) https://www.imdb.com/title/tt0053502/

(185) https://www.toptenrealestatedeals.com/homes/weekly-ten-best-home-deals/2012/08-01-2012/8/

(186) https://io9.gizmodo.com/fridays-best-deals-kindle-paperwhite-cole-haan-spong-1833150924

(187) https://en.irantrawell.com/the-tabatabaei-house-▏-kashan/

(188) https://www.klimahaus-bremerhaven.de/en/about-us/the-klimahaus/construction-and-draft.html

(189) https://www.klimahaus-bremerhaven.de/en/about-us/the-klimahaus/architecture.html

(190) http://www.dawett.ca/dining-spot-6-qualities-make-great-restaurant/

(191) "How We Whistle, Explore Science," National Geographic, December 2015

(192) Think of a soprano performing an aria; for example, "Del Primo Pianto—of the First Tears" from Turandot opera written in three acts by Giacomo Puccini in 1924, a well-known Italian composer after Verdi; https://www.theopera101.com/operas/turandot/.

(193) https://www.britannica.com/biography/Giacomo-Puccini

(194) https://www.britannica.com/art/oratorio

(195) https://www.gardensbythebay.com.sg/

(196) https://new.usgbc.org/leed

(197) https://www.usgbc.org/articles/rebuilding-and-resiliency-leed-greensburg-kansas

(198) https://www.greatgardensofcornwall.co.uk/eden-project/

(199) https://www.theatlantic.com/magazine/archive/2016/01/the-height-of-efficiency/419124/

(200) https://tech.cornell.edu/campus

(201) Mika Gröndahl, Guilbert Gates, "The Secrets of a Passive House," The New York Times, September 25, 2010.

(202) http://www.hi-energy.org.uk/renewables/wave-energy.htm

(203) https://www.nbcwashington.com/news/national-international/Five-Innovative-Green-Projects-from-Around-the-World-375893561.html

(204) http://www.middleeastgreenbuildings.com/11864/top-10-green-building-projects/

(205) https://www.thenational.ae/world/iran-looks-to-solar-alternative-for-energy-1.456219

(206) Disney PIXAR Animation Studios is an American computer animation film studio that is located in Emeryville, California and is a subsidiary of The Walt Disney Company. https://www.pixar.com/.

(207) The Incredibles is a 2004 American computer-animated superhero film written and directed by Brad Bird, produced by Disney Pixar Animation Studios, and released by Walt Disney Pictures. The film follows a family of superheroes who are forced to hide their super powers and live a quiet suburban life.

(208) https://www.imdb.com/title/tt0317705/?ref_=nv_sr_2. https://movies.disney.com/the-incredibles.

INDEX