CHAPTER 1

Introduction

After reading this chapter, you will be able to:

- define Computer-Aided-Engineering (CAE);
- discuss different types of numerical methods;
- explain Finite Element Analysis (FEA);
- identify the importance and benefits of finite element analysis; and
- discuss ANSYS software modules/capabilities.

1.1 OVERVIEW

This chapter provides an introduction to computer-aided-engineering (CAE) analysis using the finite element method. Computer-Aided-Design (CAD) analysis examples as well as salient features of the finite element method are presented to illustrate the design analysis process. An introduction to ANSYS software and its capabilities and major features is provided. ANSYS software graphical user interface is also briefly described.

1.2 COMPUTER-AIDED ENGINEERING

Computers are commonly used in engineering for simulating optimal prototype designs that best meet established performance criteria. Several mathematical and computer science-based tools are frequently used in the construction, evaluation, and testing of design objectives and specifications. It is anticipated that in the very near future, computer modeling methods and tools will merely require a formal description of the desired behavior or structure to enable automatic simulation for product evaluation.

CAE is a technology that uses computers to analyze CAD geometry, which allows the designer to simulate and study how the product will function and behave so that the design can be refined and optimized. CAE software can help perform some of the steps in the design process, especially steps related to geometric modeling, analysis and synthesis. CAE tools/systems are available for a wide range of analyses. These include dynamics analysis, FEA, general purpose, and others. The dynamics analysis includes the kinematics of bodies and deals with motion and force concepts. Several CAE analysis packages such as ANSYS, ABAQUS, COMSOL, and others can calculate the resultant stresses of a design assembly with complex geometry by specifying the loads and using fundamental equations of statics and numerical methods. These packages can be used for design analysis of heat transfer, fluid flow, electromagnetics, piezoelectric, and multi-physics problems. Then, the structural deformation/stresses can be displayed using simulation models. Figure 1.1 shows the overall procedure for CAE and outlines the analysis results used to redesign the part. Design analysis techniques using various types of numerical methods are briefly described in the next section.

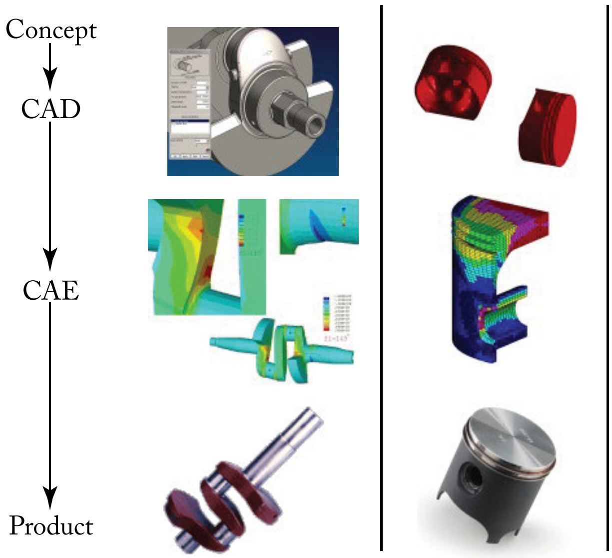


Figure 1.1: CAE approach to product design analysis and synthesis (source: `http://www.lozik.h1.ru/`).

1.3 TYPES OF NUMERICAL METHODS

Many practical engineering problems involve complex geometry, multiple materials, complex boundary and initial conditions. In order to find a solution to complex engineering problems, we need to resort to numerical methods as it is very difficult to find a closed-form/analytical solution. In general, any physical problem can be idealized as an engineering problem, as shown in Fig. 1.2. Engineering problems are mathematical models of physical situations. Many mathematical models for engineering problems are differential equations, which are derived using fundamental laws of physics (Newton's laws, Fourier law, etc.) and principles (conservation of mass, momentum and energy), with a set of boundary and/or initial conditions. For simple engineering problems, we can find an exact solution in an analytical form where the solution (response) is given in terms of several variables and parameters, as taught in "Mechanics of Materials" or "Elementary Fluid Mechanics" courses. However, for complex problems, numerical methods are required.

These include the Finite Difference Method (FDM), where derivatives are replaced by difference equations and the Finite Element Method (FEM) and the Boundary Element method (BEM) which use integral formulations to create a system of algebraic equations. In FEM, the whole domain/geometry is discretized whereas in the BEM, the boundary of the domain/geometry is discretized. The finite element design analysis techniques are discussed in the next section.

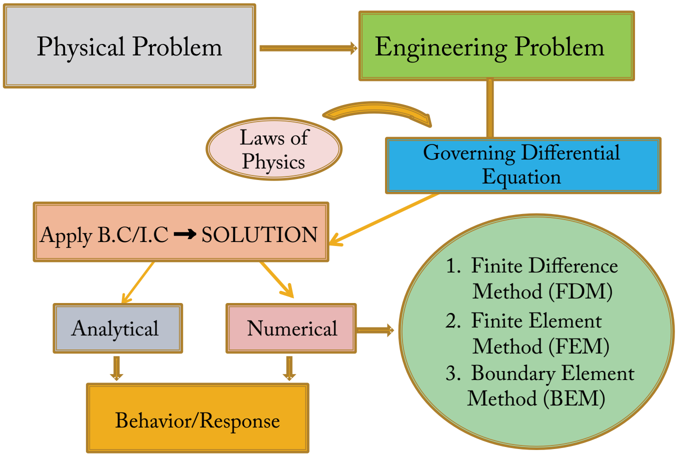


Figure 1.2: An overview of solutions to a physical problem.

1.4 FINITE ELEMENT ANALYSIS

Finite element analysis (FEA) is a general numerical method capable of solving a variety of engineering problems and the underlying theory is more than 100 years old, dating back to early 1900s. Major companies including Boeing, Ford, General Electric, Intel, IBM, Apple, and others employ general purpose computer software to efficiently complete daily engineering analysis tasks.

Finite element analysis is a numerical method that predicts the behavior (response) of a product subjected to loads (inputs). FEA is very popular and can find applications in the design analysis of mechanical, aerospace, biomedical, civil, and electrical systems. In FEA, the geometry of the designed part is broken down into smaller elements that are interconnected at nodes. In discretizing the part, the entire product geometry is filled with elements without any overlaps, and analyzed for functional performance. The part (example, an aircraft wing) is divided into a finite element mesh with smaller elements connected at nodes, as shown in Fig. 1.3. After applying loads and boundary conditions to the part, the resulting finite element equations are solved. FEA can be used for many types of analysis including stress, heat transfer, fluid flow, buckling, vibrations and multi-physics involving more than one discipline. The factor of safety against failure can be predicted from the stress analysis. Fig. 1.4 provides several design simulation examples to illustrate the applicability of FEA for analyzing a variety of engineering problems.

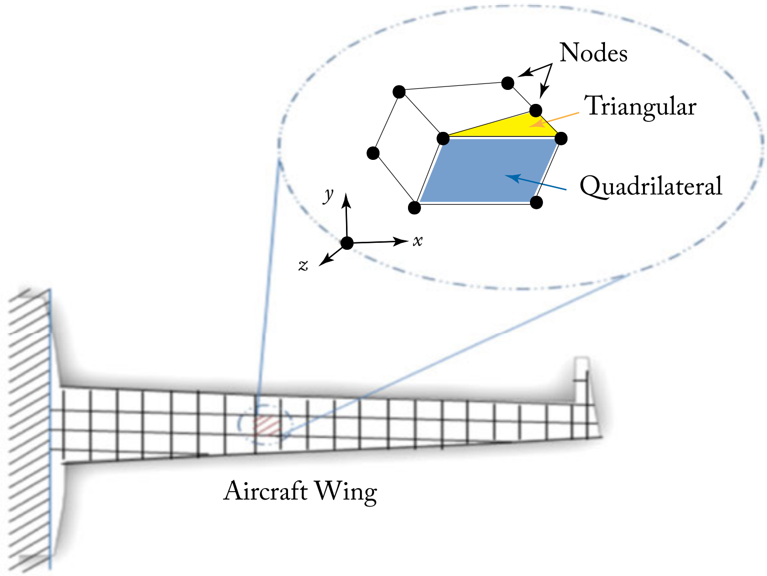


Figure 1.3: An aircraft wing is discretized with finite elements defined by shape/nodes.

1.4.1 THE NEED FOR FEA

FEA is a numerical method of solving systems of equations after discretizing the geometry with finite elements and finding a solution after applying boundary conditions to the assembled system. After formulating the finite element analysis problem, the resulting equations can be arranged for a linear system as

$$[A]\{x\} = \{b\},$$

where $[A]$ is the system matrix, $\{x\}$ is the response and $\{b\}$ is the input to the system. In stress analysis, the matrix $[A]$ becomes the stiffness matrix, vector $\{b\}$ becomes the force vector, and vector $\{x\}$ becomes the displacement vector. The method for solving response $\{x\}$ is discussed in detail in Chapter 2. In summary, the following are the main reasons for conducting design analysis using the finite element method:

- to reduce the amount of prototype testing;
- computer simulation allows multiple "what-if" scenarios to be tested quickly and effectively; and

- to simulate designs (surgical implants, Mars rover, oil rigs, etc.) that are not suitable for prototype testing or experimentation.

The bottom line:

- cost savings in the design cycle;
- time savings…reduce time to market!; and
- create more reliable and better-quality designs.

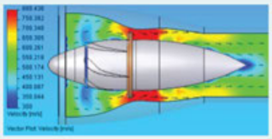
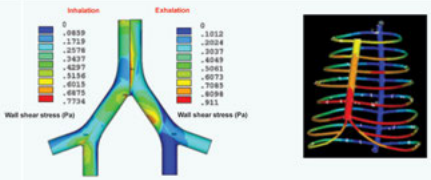
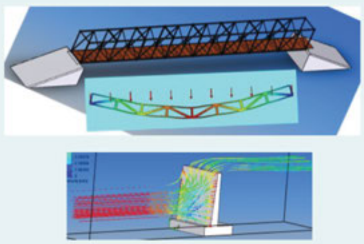
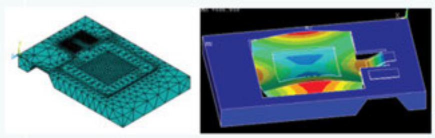
Discipline	Example
Aerospace Engineering	
Biomedical Engineering	
Civil Engineering	
Mechanical/Electrical Engineering	

Figure 1.4: Examples of FEA in many engineering disciplines.

1.4.2 BASIC STEPS IN FEA

The objective of FEA is to determine the unknowns (degrees of freedom) at the nodes and the resulting support reactions in any structure. There are three basic steps or phases involved in finite element analysis, as shown in Fig. 1.5.

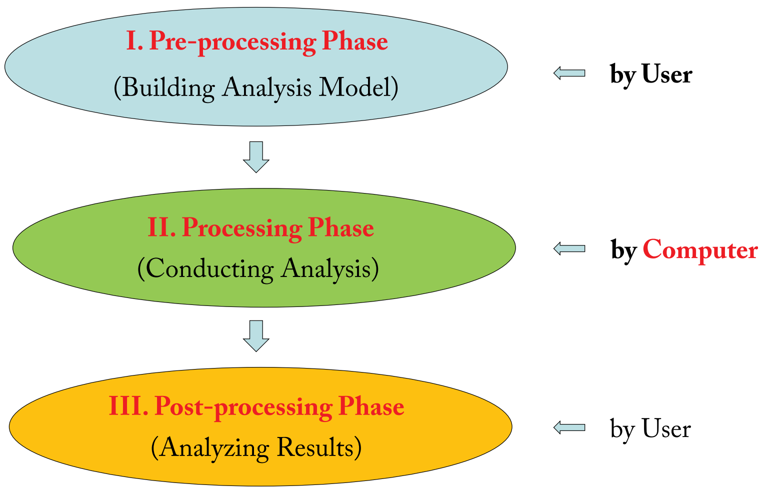


Figure 1.5: Basic phases of the FEM.

Pre-processing Phase: This phase involves building an analysis model by discretizing the domain geometry into specific finite elements defined by nodes and their connections, specifying the material properties, and applying the loads and boundary conditions.

Solution Phase: This phase involves developing a set of linear or nonlinear algebraic equations simultaneously to obtain nodal results (displacement values in stress analysis or nodal temperatures in heat transfer), for example.

Post-processing Phase: This phase involves viewing the results and obtaining results on other quantities or variables of interest such as stresses or heat fluxes derived from nodal variables.

1.5 INTRODUCTION TO ANSYS

ANSYS is a general-purpose finite element software that is tailored for analysis of multi-physics problems in solid mechanics, heat transfer, fluid mechanics, electromagnetics, and acoustics. It is a commercially available computer program which may be extensively used for studying the structural integrity and design of structural parts, and efficiency and design of thermal, fluid, magnetic, and electrical systems. It is used by mechanical, civil, aerospace, electrical, and chemical engineers in various industries. More specifically, ANSYS enables engineers to perform the following tasks:

- Build computer models or transfer CAD models of structures, products, components, or systems.
- Apply operating loads or design performance conditions.
- Study the physical responses, such as stress levels, temperature distributions, or the impact of electromagnetic fields.
- Optimize a design early in the development process to reduce production costs.
- Perform prototype testing in environments where it otherwise would be undesirable or impossible (for example biomedical applications).

The ANSYS software has a comprehensive graphical user interface that gives users easy, interactive access to program functions, commands, documentation, and reference material.

1.5.1 ANSYS MODULES

ANSYS software has the following modules.

ANSYS/Multiphysics. This is a most comprehensive product which includes all other ANSYS modules. Capabilities include structural, thermal, fluid, electromagnetic, and acoustic analyses. Coupling between structural, acoustic, electromagnetic, fluid, and thermal fields is also possible.

ANSYS/Structural. This module includes structural analysis capabilities for static and dynamic stress analysis of structures including the buckling and geometrical/material nonlinearities encountered in certain structures.

ANSYS/Mechanical. This module includes ANSYS/Structural with additional capabilities for linear/nonlinear and steady/transient heat transfer and acoustics analysis capabilities. Coupled thermal-structural and acoustic-structural analyses can also be performed.

ANSYS/LinearPlus. This module is restricted to linear static and dynamic analysis of structures.

ANSYS/Thermal. This module is restricted to steady/transient and linear/nonlinear heat transfer analysis of solids.

ANSYS/Emag. This module is designed for electromagnetics and electrostatics analysis of, electrical systems, motors, alternators, radars, etc.

ANSYS/Flotran. This Computational Fluid Dynamics Module (CFD) analyzes viscous, turbulent, incompressible and compressible fluid flow problems with and without heat transfer, including models for multi-species.

1.5.2 USING ANSYS

The simplest way to enter the ANSYS program is through the ANSYS Launcher, as shown in Fig. 1.6. The launcher has a menu containing push buttons that provide the choices you need to run the ANSYS program and other auxiliary programs.

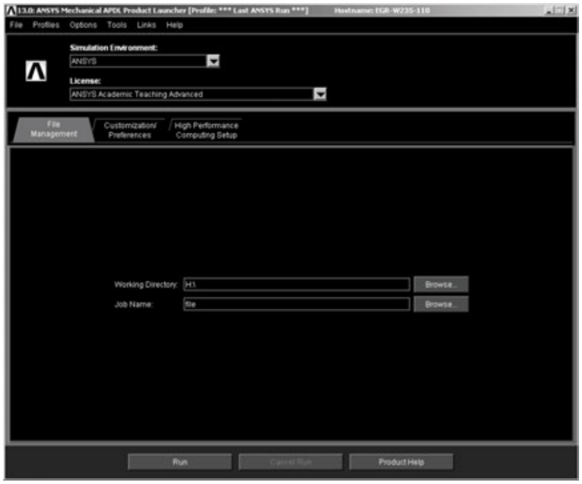


Figure 1.6: The ANSYS Launcher dialog box.

When using the Launcher to enter ANSYS, follow these basic steps:

- From the Start menu select **Programs** → **ANSYS 14.0** → **Mechanical APDL Product Launcher**.
- The ANSYS Launcher dialog box will appear, containing interactive entry options.

Working directory. This directory is the one in which the ANSYS run will be executed. If the directory displayed is not the one you want to work in pick the "**Browse…**" button to the right of the directory name and specify the desired directory. Note the directories you can use on engineering computers are on your home directory, **H:**, and USB flash drive **E:**.

Jobname. The jobname is the one that will be used as the prefix of the file name for all files generated by the ANSYS run. Type the desired jobname in this field of the dialog box (the extension for the file names will be .DB).

GUI configuration. This command brings up a dialog box that allows you to choose the desired menu layout and font size. Do not change the default settings, but simply press OK on this dialog box so that the proper settings file is created for the terminal you are using. This step is only required the first time you enter ANSYS.

A typical analysis in ANSYS involves three distinct steps.

1. *Preprocessing*. Using the **Preprocessor**, you provide data such as the geometry, materials, and element type.
    - Specify job name and title.
    - Set preferences.
    - Define element types and options.
    - Define real constants.
    - Define material properties.
    - Create model geometry.
    - Define mesh the entities.
2. *Solution*. Using the **Solution** processor, you define the type of analysis, set boundary conditions, apply loads, and initiate finite element solutions.
    - Apply Boundary Condition.
    - Specify solution controls (Static /Dynamic, Heat Transfer, etc.).
    - Specify transient characteristics—More to come later.
    - Solve.
3. *Postprocessing*. Using **General Postprocessor** (for static or steady state problems) or **Time-Hist Postprocessor** (for transient problems), you review the results of your analysis through graphical displays and tabular listings.

You enter a processor by selecting it from the **Preferences** submenu of ANSYS **Main** menu in the GUI. You can move from one processor to another by simply choosing the processor you want from the ANSYS main menu.

1.5.3 GRAPHICAL USER INTERFACE (GUI)

The simplest way to communicate with ANSYS is by using the ANSYS menu system, the GUI, as shown in Fig. 1.7. The GUI provides an interface between you and the ANSYS program. The program is internally driven by ANSYS commands. However, by using the GUI, you can perform an analysis with little or no knowledge of ANSYS commands. This process works because each GUI function ultimately produces one or more ANSYS commands that are automatically executed by the program.

Layout of the GUI. The ANSYS GUI consists of seven regions, as shown in Fig. 1.7. They include the following.

- **Utility Menu**—Contains utility functions that are available throughout the ANSYS session, such as file controls, selecting, graphics controls, and parameters. You also exit the ANSYS program through this menu.

- **Command Input Area** —Allows you to type in commands directly. All previously typed in commands can be viewed using the drop down arrow to the right of the command input area.

- **Main Menu**—Contains the primary ANSYS functions, organized by processors (preprocessor, solution, general postprocessor, design optimizer, etc.), as shown in Fig. 1.8.

- **Output Window**—Receives text output from the program. It is usually positioned behind the other windows, but you can bring it to the front when necessary.

- **Toolbar**—Contains push buttons that execute commonly used ANSYS commands and functions. You may add your own push buttons by defining abbreviations.

- **Graphics Window**—A window where graphics displays are drawn.

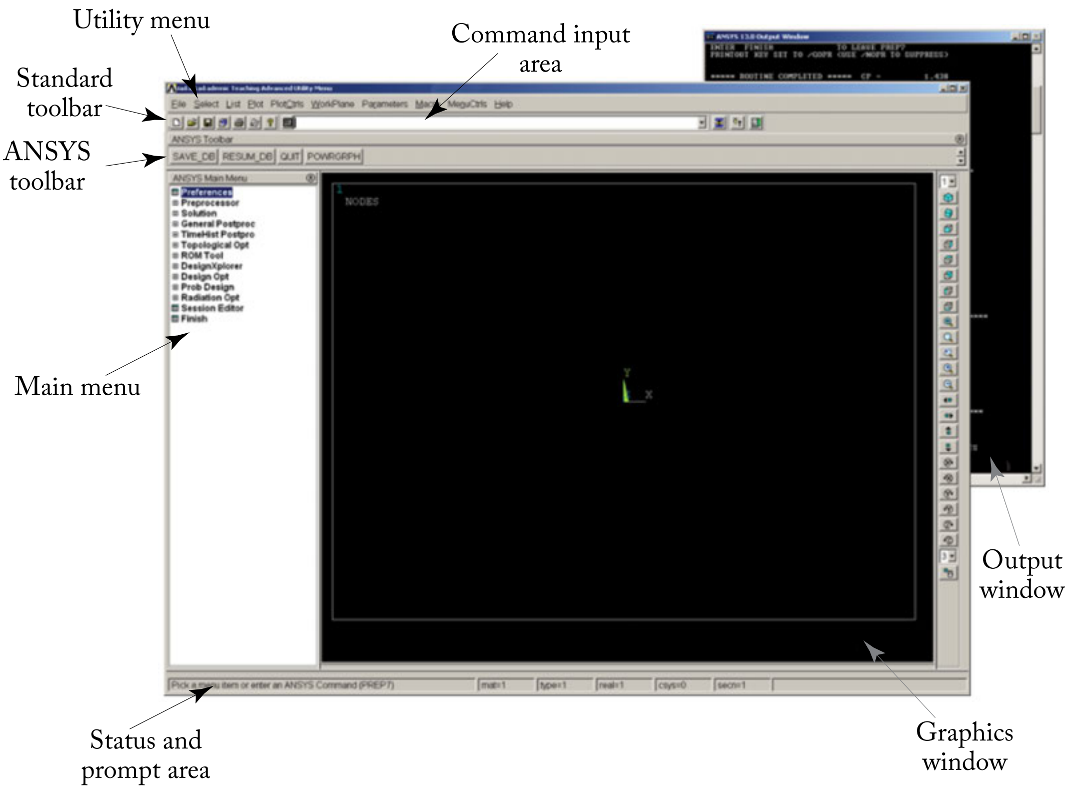


Figure 1.7: The ANSYS-GUI.

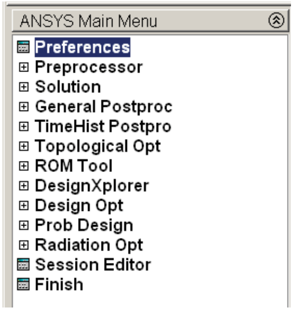


Figure 1.8: ANSYS main menu.

- **Status and Prompt Area** —Shows the status of the current analysis and prompts user for appropriate input.

1.5.4 ANSYS WORKBENCH

The recent versions of ANSYS software uses "Workbench" in addition to "Mechanical." ANSYS Workbench streamlines different components (pre- and post-processing tools and solvers) of ANSYS physics simulation. It also enables easier user-interfacing for multi-physics and coupled problems. Figure 1.9 shows the user interface of Workbench. The Workbench showcases ANSYS physics simulation pre- and post-processor, and solvers in one place. These are called "components." Users can drag a component on the right to use it, and connect components for multi-physics or coupled simulation.

As can be seen from Fig. 1.9, the components in ANSYS Workbench are categorized according to the problem to be solved. The components provide improved user interface for traditional ANSYS FEA tools such as Mechanical. In addition, models can be built using External CAD software (as well as ANSYS built-in software, the Design Modeler that can be found in "Geometry" component), and imported it to the component. Figure 1.10 shows user interface for "Static Structural" component, that uses ANSYS Mechanical as its backend to solve a static elasticity problem.

1.5.5 ANSYS WORKBENCH VS. ANSYS MECHANICAL

In the latest version of ANSYS software (beyond version 13), the Workbench option is introduced to conduct multi-physics and multi-software analysis. The examples presented in this book are based on "ANSYS Mechanical." The differences between the ANSYS Workbench and ANSYS Mechanical in terms of the process as well as GUI are presented in Figs. 1.11 and 1.12, respectively.

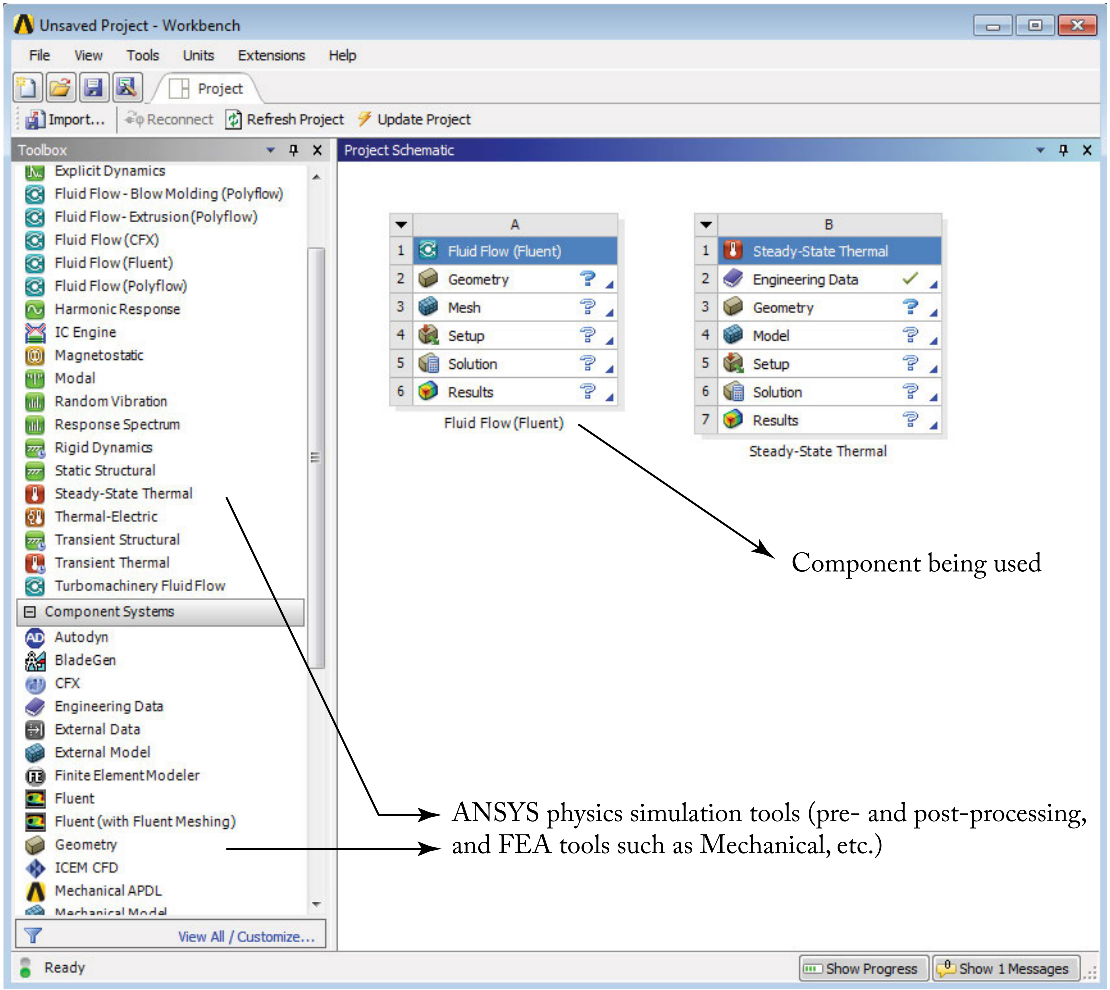


Figure 1.9: ANSYS Workbench user interface consists of components on the right that can be dragged to Workbench field from the left to be used in the solution. The components on the right consists of traditional ANSYS physics simulation solvers (such as mechanical, fluent) with improved user interface.

Note: In the ANSYS program, we can use any system of units for the problem as long as they are consistent (either SI or British system), as the program does not assume any system of units.

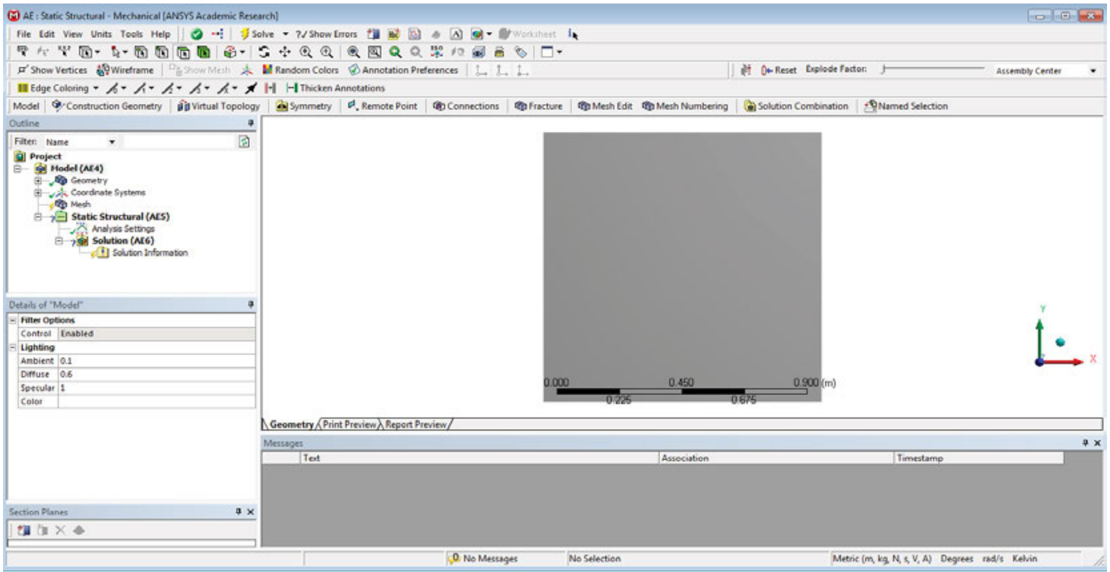


Figure 1.10: User interface for static structural analysis in Workbench.

1.6 EXERCISE PROBLEMS

1.1. What is CAE and why should we use it?

1.2. Identify five design analysis problems from everyday life.

1.3. What are the major steps/phases in FEA?

1.4. Select a product and determine the design analysis objectives. Discuss how the design analysis can be performed to improve product function.

1.5. Find two examples of designs and describe the design analysis steps.

1.6. Find design analysis examples related to civil engineering, mechanical engineering, and aerospace engineering.

1.7. What is ANSYS software? Discuss its capabilities for engineering design.

1.8. What are some of the major advantages in employing ANSYS software in a company for product design?

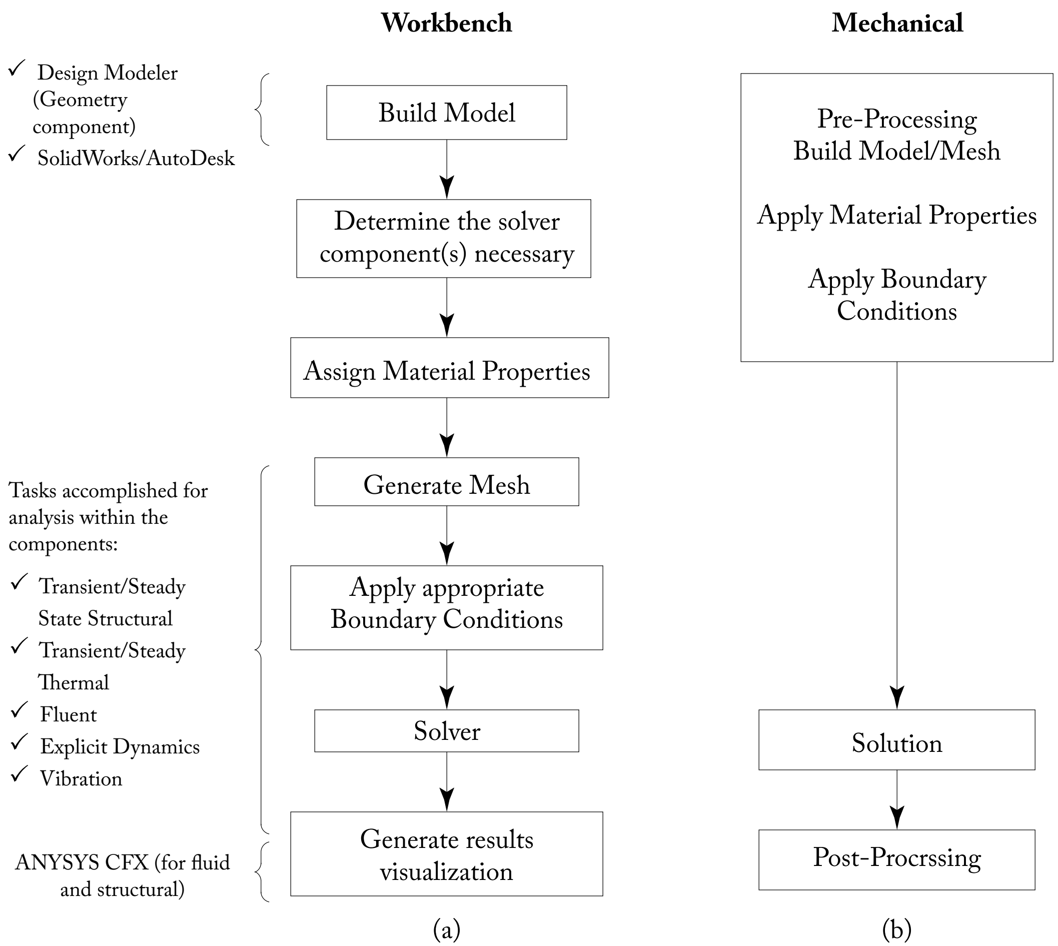


Figure 1.11: (a) Typical work flow of FEA using ANSYS Workbench. The notes show the components or external software (if necessary) to use for each step. Some components, such as Structural and Thermal, use ANSYS Mechanical as solver. (b) Typical work flow of FEA using ANSYS Mechanical. Each step is accomplished within ANSYS Mechanical.

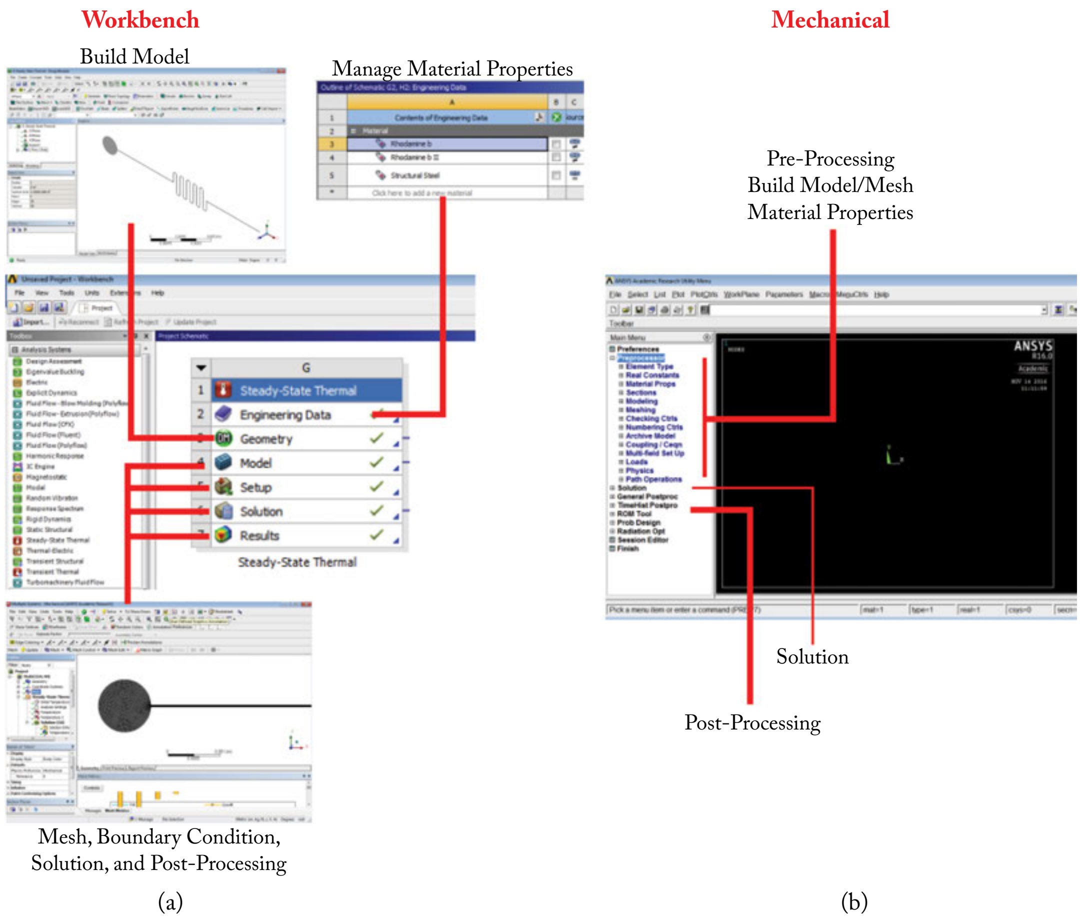


Figure 1.12: (a) Work flows in **ANSYS Workbench** are based on components that are modular. Typical components consist of model builder, material data manager, and user interface for meshing, solver, and post-processor. The modular design allows for easy data interchanges between component/solver. (b) Works in **ANSYS Mechanical** are confined within the user interface.

CHAPTER 2

Mathematical Preliminaries

After reading this chapter, you will be able to:

- explain matrices and their basic operations;
- find a solution to a matrix system of equations;
- find eigenvalues and eigenvectors of a matrix/system;
- evaluate an integral numerically; and
- use MATLAB to solve matrix algebra problems.

2.1 OVERVIEW

This chapter introduces matrix algebra, and discusses example procedures for solving matrix systems of equations and eigenvalue problems. An introduction to numerical integration is also presented as it is commonly used in finite element analysis. MATLAB instructions for carrying out matrix operations are also presented.

2.2 MATRIX ALGEBRA

FEA uses matrix algebra and vector calculus extensively, therefore an introduction to matrix algebra is presented in this chapter. Also, a matrix is a convenient way to represent engineering data. Here, we review a set of matrix operations and functions that apply to finite element analysis.

Row Vector: The row vector is a collection of elements (scalars) arranged in a row that has many columns (n). See for example, row vector $\{A\}$ as shown below:

$$\{A\} = \text{ elements that are arranged } (1xn) \text{ as,}$$
$$\{A\}_{1xn} = \{a_1,\ a_2,\ \ldots\ \ldots . a_n\}.$$

Column Vector: The column vector is a collection of elements (scalars) arranged in a column that has many rows (m).

$$\{B\} = \text{ elements that are arranged } (mx1) \text{ as}, \qquad \{B\}_{mx1} = \begin{Bmatrix} b_1 \\ b_2 \\ \dots \\ \dots \\ .. \\ b_m \end{Bmatrix}.$$

Matrix: A matrix consists of an array of mathematical objects arranged in rows and columns. The matrix size is defined by its rows and columns. Thus, a matrix of size $m \times n$ has m rows and n columns. Matrix $[\boldsymbol{A}]$ with 3 rows and 3 columns with elements is shown below.

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

2.2.1 MATRIX ADDITION AND SUBTRACTION

If matrix $\boldsymbol{A}$ and matrix $\boldsymbol{B}$ are $m \times n$ matrices, then the matrix addition or subtraction gives

$$\boldsymbol{C} = \boldsymbol{A} \pm \boldsymbol{B},$$

where C is also a $m \times n$ matrix (compatibility condition—numbers of columns of A and number of rows of B should be the same). This can be accomplished by adding or subtracting element by element as,

$$C_{ij} = A_{ij} \pm B_{ij} \text{ for } i = 1, 2, \dots m \text{ and } j = 1, 2, \dots n.$$

Example: For $m = 2$ and $n = 2$,

$$\underset{[A]}{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}} + \underset{[B]}{\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}} = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} \\ a_{21} + b_{21} & a_{22} + b_{22} \end{bmatrix}$$

$$\underset{[A]}{\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}} - \underset{[B]}{\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}} = \begin{bmatrix} a_{11} - b_{11} & a_{12} - b_{12} \\ a_{21} - b_{21} & a_{22} - b_{22} \end{bmatrix}$$

Note: multiplication of matrix $[A]$ and matrix $[B]$ is not the same as the multiplication of matrix $[B]$ and matrix $[A]$. That is,

$$[A][B] \neq [B][A].$$

2.2.2 SCALAR MULTIPLICATION

When a scalar quantity "α" is multiplied with a matrix $[A]$, the resulting matrix is the same size as $[A]$ but the elements are the product of elements α times the elements of $[A]$ as given below:

$$\alpha * [A]_{m \times n} = \alpha * a_{ij}.$$

Example:

$$5 \begin{bmatrix} 2 & 0 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 10 & 0 \\ 5 & 10 \end{bmatrix}.$$

2.2.3 TRANSPOSE OF A MATRIX

The transpose of a matrix $[\boldsymbol{A}]$ is obtained by interchanging its rows and columns and is denoted by

$$\boldsymbol{B} = \boldsymbol{A}^T \text{ or } B_{ij} = A_{ji}.$$

Example:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix}, \; A^T = \begin{bmatrix} a_{11} & a_{21} & a_{31} \\ a_{12} & a_{22} & a_{32} \end{bmatrix}.$$

Square Matrix: A matrix is defined as a square matrix if the number of rows is equal to the number of columns.

$$[A]_{n \times n} \rightarrow \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}_{2 \times 2} \rightarrow \begin{bmatrix} 3 & 1 & 1 \\ 2 & 2 & 3 \\ 4 & 1 & 4 \end{bmatrix}_{3 \times 3}.$$

Diagonal Matrix: A matrix $[A]$ is defined as a diagonal matrix if all the elements of a square matrix are equal to zero, except those on the principal diagonal.

$$[A] = \begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 \\ 0 & a_{22} & 0 & 0 & 0 \\ 0 & 0 & a_{33} & 0 & 0 \\ 0 & 0 & 0 & a_{44} & 0 \\ 0 & 0 & 0 & 0 & a_{nn} \end{bmatrix}_{n \times n}.$$

Unit or Identity Matrix: A unit matrix is a special case of a diagonal matrix where all the diagonal elements are equal to 1 and all off-diagonal elements are equal to zero.

Symmetric Matrix: A symmetric matrix is defined as a square matrix with property, for i not equal to j, as

$$a_{ij} = a_{ji}.$$

2.2.4 MULTIPLICATION OF VECTORS

Let "$\boldsymbol{a}$" and "$\boldsymbol{b}$" represent two vectors of size m (or $m \times 1$), then the *Dot* or *Inner Product* is defined as

$$c = \boldsymbol{a}^T \boldsymbol{b} = \boldsymbol{b}^T \boldsymbol{a} = \sum_{i=1}^{m} (a_i b_i)$$
$$= a_1 b_1 + a_2 b_2 + \cdots + a_n b_n.$$

The *Tensor Product* is defined as

$$\boldsymbol{D} = \boldsymbol{a}\boldsymbol{b}^T$$

or

$$D_{ij} = a_i b_j \text{ for } i, j = 1, 2 \ldots m.$$

Example: for $m = 3$,

$$D = \begin{bmatrix} a_1 b_1 & a_1 b_2 & a_1 b_3 \\ a_2 b_1 & a_2 b_2 & a_2 b_3 \\ a_3 b_1 & a_3 b_2 & a_3 b_3 \end{bmatrix}.$$

2.2.5 MULTIPLICATION OF MATRICES

If the size of matrix $[\boldsymbol{A}]$ is a $m \times l$ and the size of matrix $[\boldsymbol{B}]$ is a $l \times n$, then the matrix multiplication gives

$$\boldsymbol{C} = \boldsymbol{A}\boldsymbol{B},$$

where $[\boldsymbol{C}]$ is a $m \times n$ matrix. This can be accomplished by

$$C_{ij} = \sum_{k=1}^{m} a_{ik} b_{kj}$$

for $i = 1, 2 \ldots l$ and $j = 1, 2 \ldots n$.

Example: for $m = 2, l = 3, n = 2$, then

$$C_{i,j} = a_{i1} b_{1j} + a_{i2} b_{2j} + a_{i3} b_{3j}$$
$$C_{11} = a_{11} b_{11} + a_{12} b_{21} + a_{13} b_{31}$$
$$C_{12} = a_{11} b_{12} + a_{12} b_{22} + a_{13} b_{32}$$
$$C_{21} = a_{21} b_{11} + a_{22} b_{21} + a_{23} b_{31}$$
$$C_{21} = a_{21} b_{12} + a_{22} b_{22} + a_{23} b_{32}.$$

2.2.6 MULTIPLICATION OF A MATRIX WITH A VECTOR

If $[\boldsymbol{A}]$ is a $m \times n$ matrix and $\boldsymbol{x}$ is a $n \times 1$ vector, then

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{f},$$