

Ministry of Education of the Republic of Belarus

Educational Institution
“Francisk Skorina Gomel State University”

INTRODUCTION TO COMPUTER MODELING TECHNOLOGIES

Recommended as a study aid by the Educational
and Methodological Association for Education in the Fields
of Computer Science and Radio-Electronics for the specialties
6-05-0611-05 “Computer Engineering”
and 6-05-0713-02 “Electronic Systems and Technologies”

Gomel
Francisk Skorina State University
2026

UDC 004.94(076)
BBK 32.973.4я73
N64

Authors:

Yu. V. Nikityuk, A. A. Sereda,
V. A. Prokhorenko, D. L. Kovalenko, A. L. Samofalov

Reviewers:

Doctor of Engineering A. A. Boika,
PhD in in Physics and Mathematics N. V. Ryazantseva,
PhD in Philology I. A. Khorsun

Recommended for publication by the science and technology council
of Francisk Skorina Gomel State University

Introduction to computer modeling technologies : manual /
N64 Yu. V. Nikityuk [and others] ; Ministry of Education of the
Republic of Belarus, Francisk Skorina Gomel State University. –
Gomel : Francisk Skorina Gomel State University, 2026. – 240 p.
ISBN 978-985-32-0221-2

Computer modeling is an interdisciplinary field that requires a broad range of knowledge for successful mastery. Computer modeling technologies demand that a researcher be able to correctly define problems, rank input parameters when constructing a model, forecast research outcomes, conduct computer-based experiments, and analyze the results.

This educational and methodological manual contains theoretical material, examples of how to solve practical problems, and test questions for assessing knowledge. The manual presents five examples of problem-solving using the ANSYS analysis software, two examples using the ANSYS Workbench environment, and two examples using the ANSYS DesignXplorer module. Theoretical content accounts for slightly more than half of the manual's total volume.

The publication is addressed to students of specialties 6-05-0713-02 “Electronic systems and technologies”, 6-05-0611-05 “Computer engineering”.

UDC 004.94(076)
BBK 32.973.4я73

ISBN 978-985-32-0221-2

© Educational Institution
“Francisk Skorina
Gomel State University”, 2026

CONTENTS

INTRODUCTION	6
1 COMPUTER AND MATHEMATICAL MODELING	8
Topic 1.1 The role of modeling in scientific methods.....	8
Topic 1.2 Basic concepts and definitions	9
1.2.1 Definition of modeling and model.....	9
1.2.2 Fundamental properties of models.....	11
1.2.3 Objectives of modeling.....	13
1.2.4 Principles of modeling	14
Topic 1.3 Methodology of mathematical and computer modeling ..	15
Topic 1.4 Model classification.....	17
1.4.1 Material modeling.....	18
1.4.2 Ideal modeling	19
1.4.3 Classification of mathematical models.....	21
1.4.4 Classification of computer models	25
Topic 1.5 Stages of computer mathematical model development .	27
2 CALS/PLM, AND CAE SYSTEMS	32
Topic 2.1 CALS/PLM systems	32
2.1.1 Key terms and definitions.....	32
2.1.2 History of CALS/PLM systems.....	33
2.1.3 Structure of CALS/PLM systems	35
2.1.4 CALS/PLM systems: strategic objectives and	
core functions	38
2.1.5 Implementation of CALS/PLM systems	39
2.1.6 Advantages and Limitations of CALS/PLM	
Technologies	39
Topic 2.2 SAE systems	41
2.2.1 Capabilities of SAE systems.....	42
2.2.2 Market leaders in CAE systems.....	43
2.2.3 Main development trends in CAE systems.....	44
Topic 2.3 Computational methods for engineering analysis	
in CAE systems	44
2.3.1 Analytical methods	45
2.3.2 Numerical methods	46
2.3.3 Finite element method	49
3 ANSYS MECHANICAL APDL.....	53
Topic 3.1 Basic information about ANSYS MECHANICAL APDL	53
3.1.1 Software package organization.....	53

3.1.2 Software database	54
3.1.3 Graphical user interface	55
3.1.4 Graphical indication of objects	60
3.1.5 File management	61
Topic 3.2 Problem Analysis in ANSYS Mechanical APDL	63
3.2.1 Preliminary settings	63
3.2.2 Model construction	63
3.2.3 Setting loads and performing analysis	74
3.2.4 Postprocessing of results	78
Topic 3.3 Automated modeling in ANSYS MECHANICAL APDL	81
3.3.1 Complex commands	82
3.3.2 Batch File Structure	84
3.3.3 Programming elements	84
4 ANSYS WORKBENCH	92
Topic 4.1 Basic information about ANSYS WORKBENCH	92
Topic 4.2 ANSYS WORKBENCH Graphical Interface	93
Topic 4.3 Working with a project in ANSYS WORKBENCH	99
4.3.1 Overview of core modules and applications	99
4.3.2 Setting material properties	103
4.3.3 Geometry construction and import	108
4.3.4 Finite element model generation	114
4.3.5 Setting boundary conditions and loads	121
4.3.6 Calculation execution and results output	123
5 IMPLEMENTATION EXAMPLES FOR ANSYS ANALYSIS SOFTWARE	127
Problem 5.1 “Steady-State Thermal Analysis of Plate Heating in 3D Formulation”	127
5.1.1 Problem statement and assumptions	127
5.1.2 Implementation of the problem using the program’s graphical interface	127
Problem 5.2 “Transient Analysis of Ingot Solidification in 3D Formulation”	132
5.2.1 Problem statement and assumptions	132
5.2.2 Implementation of the problem using the program’s graphical interface	133
Problem 5.3 “Thermal-Stress Analysis of Thick-Walled Hollow Cylinder in 3D Formulation”	139
5.3.1 Problem statement and assumptions	139
5.3.2 Implementation of the problem using the program’s graphical interface	140

Problem 5.4 “Transient Thermal Analysis with Tabular Boundary Conditions in 3D Formulation”	146
5.4.1 Problem statement and assumptions	146
5.4.2 Implementation of the problem using the program’s graphical interface	147
Problem 5.5 “Steady-state thermal analysis of temperature field at pipe intersection”	152
5.5.1 Problem statement and assumptions	152
5.5.2 Implementation of the problem using the program’s graphical interface	153
6 IMPLEMENTATION EXAMPLES FOR ANSYS WORKBENCH ENVIRONMENT.....	162
Problem 6.1 Fundamentals of ANSYS Workbench: static strength analysis	162
6.1.1 Problem statement.....	162
6.1.2 Implementation of the problem using the Ansys Workbench graphical interface	162
Problem 6. 2 Calculation of thermoelastic stresses in ANSYS WORKBENCH.....	176
6.2.1 Problem statement.....	176
6.2.2 Implementation of the problem using Ansys Workbench graphical interface	177
7 IMPLEMENTATION EXAMPLES FOR ANSYS DESIGNXPLORER MODULE.....	191
Problem 7.1 “What-If” Parametric Study in ANSYS Workbench	191
7.1.1 Problem statement.....	191
7.1.2 Implementation of the problem using the Ansys Workbench graphical interface	192
Problem 7.2 Optimization in ANSYS DESIGNXPLORER	203
7.2.1 Introduction.....	203
7.2.2 Problem statement.....	204
7.2.3 Implementation of the problem using the Ansys Workbench graphical interface	205
8 TEST ASSIGNMENTS	220
Test 8.1 Introduction to computer modeling technologies	220
Test 8.2 ANSYS WORKBENCH software package.....	229
BIBLIOGRAPHY	235

INTRODUCTION

Among the basic competencies that must be developed in specialists graduating from the Faculty of Physics and Information Technology, the following can be highlighted: establishing physical patterns based on modern theoretical concepts, mathematical and computer methods; research work in areas using physical and mathematical methods of analysis and computer technologies; development of physical and mathematical methods for solving problems in engineering, economics and management; performing calculations in the process of designing standard designs and parts of electronic security systems, taking into account knowledge of the mechanics of materials, the theory of machines and mechanisms; analysis of the operation of various types of electrical and electronic components in the designs of devices and systems of specified parameters; creation of new materials and analysis of their properties. The formation of such competencies is inextricably linked with the modeling of various physical processes occurring in materials and products.

The development of mathematical physics and computational mathematics led to the creation of mathematical modeling – a new tool for studying nature. With the advent of computers, mathematical modeling began to intensively penetrate into all areas of scientific research and engineering practice. For this reason, computer modeling has now become a means of solving many complex practical problems.

Computer modeling is an interdisciplinary course that requires a broad range of knowledge for its successful mastering. Computer modeling technologies require the researcher to be able to correctly set tasks, rank input parameters when constructing a model, predict research results, conduct computer experiments and analyze their results.

The textbook examines the principles and methods of mathematical and computer modeling, provides basic concepts and definitions, and presents the main stages of constructing mathematical and computer models.

The concept of science-intensive software systems for computer modeling is presented, encompassing CAE technologies, which constitute the natural science core of the CAD-CAE-CAM technology triad in the CALS/PLM technology system.

The manual reviews the basic information about the ANSYS MECHANICAL APDL complex, presents the main stages of problem analysis in it using a graphical interface, and provides the necessary information for implementing automated modeling using the APDL programming language.

The manual provides basic information about the ANSYS WORKBENCH program, a description of its graphical interface, and a step-by-step algorithm for working with a project in ANSYS WORKBENCH.

Examples of implementing elasticity and heat transfer problems in the ANSYS engineering product, examples of implementing strength static analysis problems and calculating thermoelastic stresses in Ansys Workbench are given.

The manual includes a practical example of optimization problem implementation using Ansys DesignXplorer.

Using the Moodle software shell, test tasks have been developed for conducting current and final knowledge assessment on the course “Introduction to Computer Modeling Technologies”, which use tasks of various types (single choice, multiple choice, matching tasks) and different levels of complexity. Current knowledge assessment is carried out in the training mode and allows the test taker to objectively assess their knowledge.

This manual can be used for laboratory work and independent study, helping students develop practical skills in Ansys software.

1 COMPUTER AND MATHEMATICAL MODELING

Topic 1.1 The role of modeling in scientific methods

Modeling is one of the methods of studying the surrounding world. The concept of method means a set of techniques and operations of practical and theoretical mastering of reality. To master a method means to know how, in what sequence, to perform certain actions to solve various problems, and to be able to implement this knowledge in practice.

Methods of scientific inquiry are generally categorized into three groups: *universal*, *general scientific*, and *discipline-specific* methods (Figure 1.1).

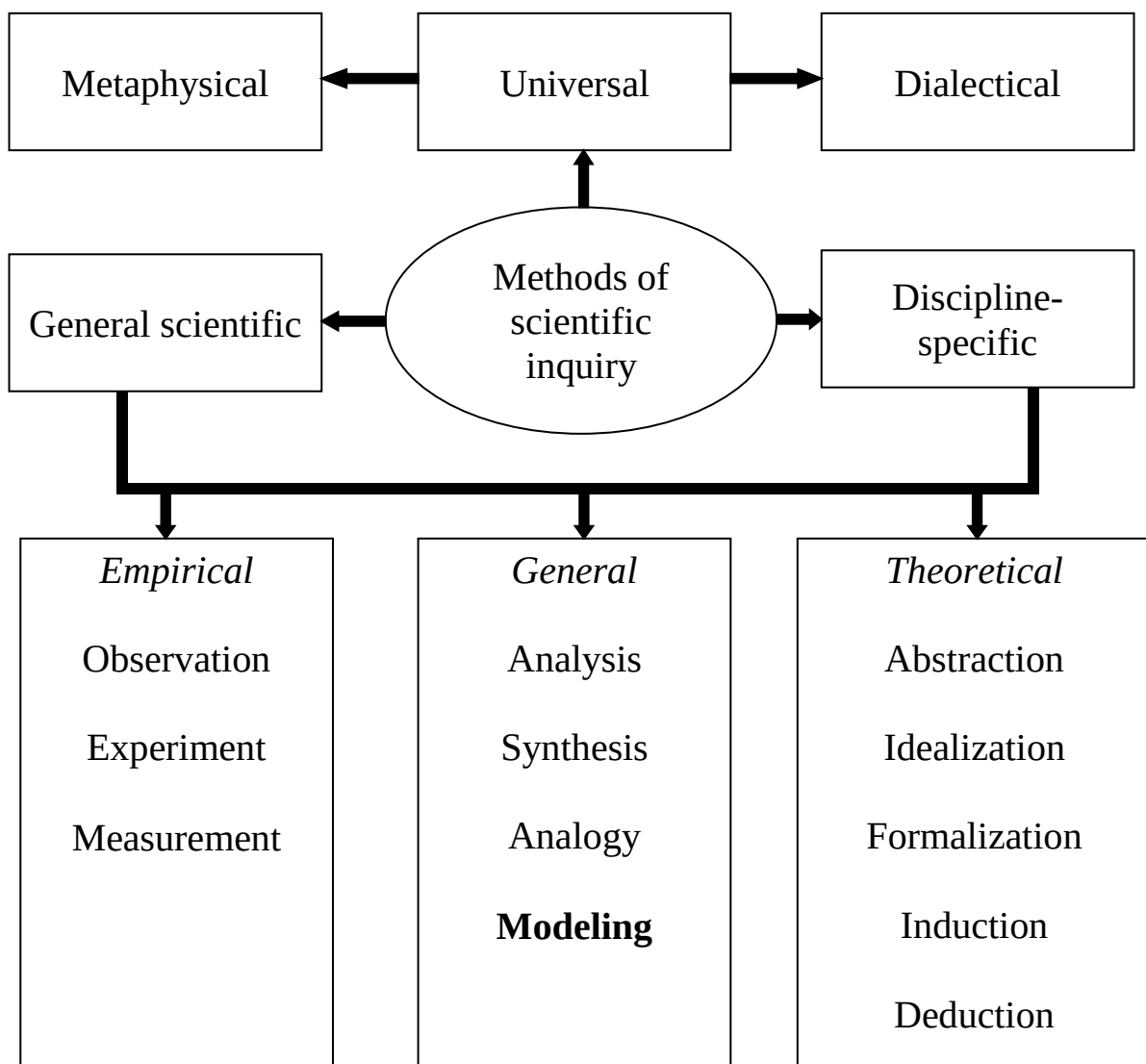


Figure 1.1 – Methods of scientific inquiry

In the history of knowledge, two principal methods are recognised: dialectical and metaphysical. These constitute broad philosophical methodologies. The metaphysical approach examines things and phenomena of the surrounding universe in isolation, disregarding their interconnections and perceiving them as if in a static, unalterable condition. The dialectical method, conversely, necessitates the examination of objects and phenomena in all their intricate interrelations, considering the actual processes of their transformation and evolution.

General scientific methods have a wide transdisciplinary applicability. The classification of these methods is closely related to the concept of levels of scientific knowledge. Scientific knowledge exists at two levels: empirical and theoretical. Certain scientific approaches are employed exclusively at the empirical level (observation, experimentation, measurement), whereas others are utilized solely at the theoretical level (idealization, formalization).

Modeling refers to methods that are used at both empirical and theoretical levels.

Self-Assessment Questions

- 1 What is the definition of the term “method”?
- 2 Into what categories are the methods of scientific inquiry classified?
- 3 What types of general methods are you familiar with?
- 4 Highlight the levels of scientific inquiry.

Topic 1.2 Basic concepts and definitions

1.2.1 Definition of modeling and model

The concepts of model and simulation are most common in the field of education, scientific research, design and engineering work, and in serial technical production. Most often, the term “model” is used to denote:

- a device that reproduces the structure or action of some other device (reduced, enlarged or full size);
- an analogue (drawing, graph, plan, diagram, description, etc.) of specific phenomenon, process or object.

The term “model” is polysemantic. Dictionaries contain up to eight different meanings of this term. The following definitions of a model are most common in scientific literature:

- a model as an analogue of a real object;
- a model as a sample of a future product.

Hypotheses and analogies play an important role in developing models. Hypotheses and analogies, which to a certain extent reflect the real world, must be visual or reduced to logical schemes convenient for research. It is precisely such schemes, which simplify reasoning and logical constructions, allow experiments to be carried out and lead to an understanding of natural phenomena, that are called *models*. In other words, a model is an object-substitute for the original object, ensuring the study of certain properties of the original that are of interest to the researcher.

So, *modeling* is a method of understanding the surrounding world, which is the process of replacing the object of study with some model of it and conducting research on the model in order to obtain the necessary information about the object.

In this case, a model can be defined as an image of a simulated object, convenient for conducting research and allowing for an adequate display of the properties and characteristics of the object of interest to the researcher.

In connection with the development of modern information technologies, the use of computer modeling technologies is expanding in solving problems of development, production and maintenance of products. Computer models are becoming a form of presenting the results of design and engineering activities. At the same time, the role of computer modeling as an alternative to physical testing is increasing, allowing for a significant reduction in testing costs during the creation of products. The key concept in the field of computer modeling is the *computer model*.

The following terms and corresponding definitions are used in regulatory documents governing the use of computer modeling.

A *model* is an entity that reproduces a phenomenon, object, or property of a real-world object.

The *object of modeling* is a phenomenon, object or property of an object in the real world.

A *modeling aspect* is a separate property or a set of properties of a modeling object that is the subject of research.

A *mathematical model* is a model in which information about the modeling object is presented in the form of mathematical symbols and expressions.

An *information model* is a model in which information about a modeling object is presented as a set of data elements and relationships between them.

Modeling is the study of the properties and/or behavior of a modeling object, performed using its models.

A *computer model* is a model created in a computer (computing) environment and representing a set of data and program code necessary for

working with the data. A computer model is based on a mathematical model implemented in the form of program code and data defining a specific modeling object. To use a computer model in the modeling process, it is necessary to use computer modeling software and computing technology.

Computer modeling software refers to programs that perform mathematical calculations and are intended for the preparation of initial data, processing of calculation results, as well as other auxiliary programs. Computer modeling software, in terms of application, represents a broader concept than individual computer models. It enables the simulation of diverse modeling objects while incorporating service functions that support model implementation and analysis.

1.2.2 Fundamental properties of models

Models possess several critical properties that determine their successful application. Below we outline the most essential ones.

Completeness is one of the characteristics of a model. It can be argued that any object of study is infinitely complex and is characterized by an infinite number of parameters. When constructing a model, the researcher always proceeds from the set goals and takes into account only the most significant factors. Therefore, any model is not identical to the original, and is thus inherently incomplete. It should be noted that other factors, despite their relatively small influence on the behavior of the object compared to the selected factors, can collectively lead to significant differences between the object and its model.

Accuracy makes it possible to ensure an acceptable match between the actual values of the system's output parameters and those found using the model.

Adequacy is the ability of a model to reflect the properties of a system with a relative error not lower than a specified one. In a general sense, adequacy of a model is understood as a correct qualitative and sufficiently accurate quantitative description of precisely those characteristics of a system that are important in a particular case. A model that is adequate when choosing some characteristics may be inadequate when choosing other characteristics of the system. In a number of applied areas that are not yet sufficiently prepared for the use of quantitative mathematical methods, models are primarily qualitative in nature. In such cases, adequacy of a model should be understood as only a correct qualitative description of the behavior of the systems being studied. Thus, adequacy is the degree of correspondence of a model to the real object being studied. It can never be complete. In practice, a model is considered adequate if it allows one to achieve the research goals with satisfactory accuracy.

Simplicity of the model can be one of the characteristics of the model. The more properties of the object the model describes, the more complex it is. A complex model is not always adequate. Obviously, of the two models that allow achieving the desired goal and obtaining the required results with the specified accuracy, preference should be given to the simpler one. At the same time, adequacy and simplicity of the model are not always contradictory requirements. Considering the infinite complexity of any object of study, one can assume the existence of an infinite sequence of its models, differing in the degree of completeness, adequacy and simplicity.

Efficiency is assessed by the costs of computing resources required to implement the model on a computer. These costs depend on the number of arithmetic operations when using the model, on the features of the computer used, and other factors. It is obvious that the requirements of efficiency, high accuracy, and adequacy of the model are contradictory and in practice can only be satisfied based on a reasonable compromise. A model's efficiency is often linked to its simplicity. Certain simplified model versions may even permit quantitative analysis without modern computing tools. However, such results typically offer only limited utility during algorithm or program debugging stages.

The *robustness* of a model characterizes its stability with respect to errors in the initial data, the ability to level out these errors and prevent their excessive influence on the result of a computational experiment. The model's low robustness may stem from the necessity to subtract closely approximate values during quantitative analysis or to divide by a small absolute value, as well as the incorporation of functions that exhibit rapid variation within the interval where the argument's value is known with low accuracy.

The *productivity* of the model is related to the reliability of the initial data. If they are the result of measurements, the accuracy of their measurement should be higher than for those parameters that are obtained using the model. Otherwise, the model will be unproductive and its use for analyzing a specific system will lose its meaning. It can only be used to evaluate and characterize a certain class of systems with hypothetical initial data.

The *clarity* of the model is a desirable but not mandatory property. However, the use of the model and its modification are simplified if its components (for example, individual terms of the equations) have a clear substantive meaning. This allows one to predict the results of the computational experiment and facilitate the control of their correctness.

Another property of a model is its potentiality, or predictability, regarding the model's capacity to generate new knowledge about the studied object. In scientific research, models lacking a definitive

“predictability” cannot be considered satisfactory. Hence, *potentiality* (*predictability*) refers to a model’s capacity to yield novel insights about the research object, as well as to forecast its behavior or properties.

1.2.3 Objectives of modeling

Let us consider the primary objectives of modeling. An important and prevalent function of models is to analyze and predict the behavior of complex processes and phenomena.

Models are often used to identify the most significant factors that shape certain properties of an object. Naturally, a model of any real process or phenomenon is “poorer” than the process itself in terms of its objective existence (process, phenomenon). At the same time, a good model is “richer” than what is comprehended by reality, as it is typically impossible for an individual to comprehend the entire network of connections in intricate systems. A model allows one to “play” with it, enabling or disabling specific connections and altering them to gain a better understanding of their significance in the system’s overall behaviour.

It is essential to recognise that certain things and phenomena cannot be explored directly. For instance, extensive “natural” studies involving a country’s economy or the health of its population are inadmissible. Investigations into the historical context of any state or people are fundamentally impractical. Conducting an experiment to directly explore the structure of stars is unfeasible. Numerous experiments are unrealistic due to their exorbitant costs or potential hazards to individuals or the environment.

Currently, detailed preliminary examinations of several models of the phenomenon precede any complex assessments. Furthermore, computer simulations enable the formulation of a comprehensive strategy for large-scale experiments, identification of necessary specifications for measurement apparatus, scheduling of observations, and cost estimation for the experiment.

When an object’s attributes evolve over time, anticipating its states in response to multiple influences becomes particularly critical. In the design and operation of complex technical devices, it is essential to anticipate variations in the reliability of both individual subsystems and the overall device.

The model also allows you to master the correct manipulation of the object by experimenting with various control alternatives. Employing a real object for this purpose is often risky or outright impossible. For example, acquiring initial abilities in operating a contemporary aircraft is safer, more expedient, and less costly through a simulator (i.e., a model) than by risking both oneself and an expensive aircraft.

So, modeling is conducted for the following purposes:

1) to investigate the mechanisms of phenomena, specifically to comprehend the structure of a particular object, including its composition, internal relationships, fundamental properties, developmental laws, self-evolution, and interactions with the environment;

2) to facilitate the management of the object, which involves identifying effective strategies for controlling an object or process in alignment with specified objectives and criteria.

1.2.4 Principles of modeling

Modeling is based on several fundamental principles.

The *principle of information sufficiency*. Without any information regarding the system under examination, constructing its model is unfeasible. When complete information about the system is available, its modelling becomes meaningless. A specific threshold of a priori information concerning the system, known as the level of information sufficiency, must be attained to develop an adequate model.

The *principle of feasibility*. The model under development must guarantee the attainment of the established research objective with a probability markedly distinct from zero, within a finite timeframe. A specific criterion for the probability of attaining the modeling objective is typically established, along with an acceptable time frame for its realization.

The *principle of multiple models*. This principle is fundamental. The point is that the model being created should primarily reflect those properties of the real system (or phenomenon) that affect the selected efficiency indicator. Accordingly, when using any specific model, only some aspects of the real system are studied. For a more complete study, a number of models are needed that allow the system under consideration to be reflected from different sides and with varying degrees of detail.

The *aggregation principle*. In most cases, a complex system can be represented as subsystems, for the adequate description of which certain standard mathematical schemes are suitable. In addition, the aggregation principle allows for a fairly flexible restructuring of the model depending on the research objectives.

The *parameterization principle*. In some cases, the modeled system includes relatively isolated subsystems characterized by certain parameters, including vector parameters. Such subsystems can be replaced in the model by corresponding numerical values, rather than describing the process of their functioning. If necessary, the dependence of the values of these values

on the situation can be specified in the form of a table, graph, or analytical expression (formula). The parameterisation principle facilitates the reduction of both the volume and duration of modelling. It is important to note that parameterisation diminishes the model's adequacy.

Self-Assessment Questions

- 1 What is the primary application of the term “model”?
- 2 What terminology from regulatory papers pertaining to computer modeling are you familiar with?
- 3 What are the key attributes of a model that influence the efficacy of its application?
- 4 What are the primary objectives of modeling?
- 5 Enumerate several fundamental principles underlying modeling.

Topic 1.3 Methodology of mathematical and computer modeling

Elements of mathematical modeling have been integral to the exact sciences since their inception, and it is not coincidental that certain computational methods are named after prominent figures such as Newton and Euler, while the term *algorithm* is derived from the name of the medieval Arab scholar Al-Khwarizmi.

The second “birth” of this methodology occurred in the late 40s – early 50s of the 20th century, and was due to at least two reasons. The first of them is the appearance of computers. The second is the implementation of national programs of the USSR and the USA to create a missile-nuclear shield, which could not be implemented by traditional methods.

Computer modeling addressed this task: nuclear explosions, as well as rocket and satellite flights, were initially “simulated” within computers using mathematical models before being executed in practice. This success largely determined the further achievements of the methodology, which is currently deemed essential for any large-scale technological, ecological, or economic initiative.

Computer modeling is now entering the third stage of its development, “integrating” itself into the structures of the information society. Without possession of information resources, it is impossible to even think about solving the increasingly large and increasingly diverse problems facing the world community.

The methodology of computer modeling is expressed by the triad “model – algorithm – program”. This concept posits that the formulation of a mathematical modeling query for any object establishes a definitive course of action, which can be conditionally divided into three stages: model – algorithm – program (Figure 1.2).

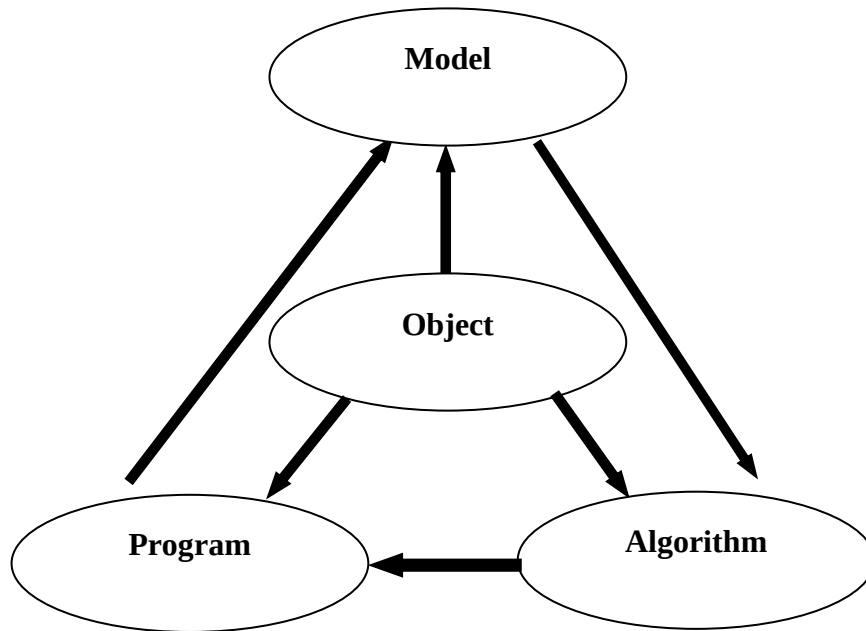


Figure 1.2 – Triad “model – algorithm – program”

At the first stage, an “equivalent” of the object is selected (or constructed), reflecting in mathematical terms its most important properties – the governing laws, the connections inherent in its constituent parts, etc. The mathematical model (or its fragments) is analyzed using theoretical approaches, enabling the acquisition of important preliminary insights on the object.

The model is presented in a form convenient for the application of numerical methods. The sequence of computational and logical operations that must be performed to find the desired values with a given accuracy is determined. Computational algorithms must not distort the basic properties of the model, and therefore, the original object, while also being economical and adaptable to the specific characteristics of the problems addressed and the computers employed.

At the third stage, programs are created that “translate” the model and algorithm into a computer-readable language.

Having created the triad “model – algorithm – program”, the researcher gets a universal, flexible and inexpensive tool, which is debugged and tested in “trial” computational experiments. After that, subject to the adequacy (sufficient correspondence) of the triad to the original object, various

“experiments” are carried out with the model, providing all the required properties and characteristics of the object. It is important that the modeling process be accompanied by the improvement and clarification of all links of the triad. This cognitive approach integrates the benefits of both theoretical and experimental methodologies. Indeed, working with a computer model rather than the object itself (phenomenon or process) enables the efficient and cost-effective examination of its characteristics and behavior under many hypothetical scenarios. This is the benefit of theory. Simultaneously, computational experiments utilizing object models offer a comprehensive examination of things, leveraging the capabilities of computational methods and technology, which is unattainable through only theoretical approaches. This is already the advantage of experimentation.

Computer modeling as a methodology does not replace mathematics, physics, biology and other scientific disciplines, but rather serves a synthesising function. The creation and application of the triad is impossible without relying on a variety of methods and approaches, ranging from qualitative analysis of nonlinear models to modern programming languages. However, addressing the challenges of the information society necessitates more than just reliance on computational capabilities. Continuous enhancement of the triad of mathematical modeling is essential.

Self-Assessment Questions

- 1 What stages of advancement in mathematical and computational modeling may be identified?
- 2 What is the fundamental nature of the computer modeling technique, represented by the triad “model – algorithm – program”?
- 3 What is the flexibility and multifunctionality of the triad “model – algorithm – program”?

Topic 1.4 Model classification

As noted earlier, modeling refers to general scientific methods of cognition. The use of modeling at the empirical and theoretical levels of research leads to the division of models into *material* and *ideal*.

Considering that ideal modeling precedes material modeling (first, an ideal model is conceived in an individual’s mind, followed by the construction of a material model based on it), existing types of modeling can be presented as shown in Figure 1.3.

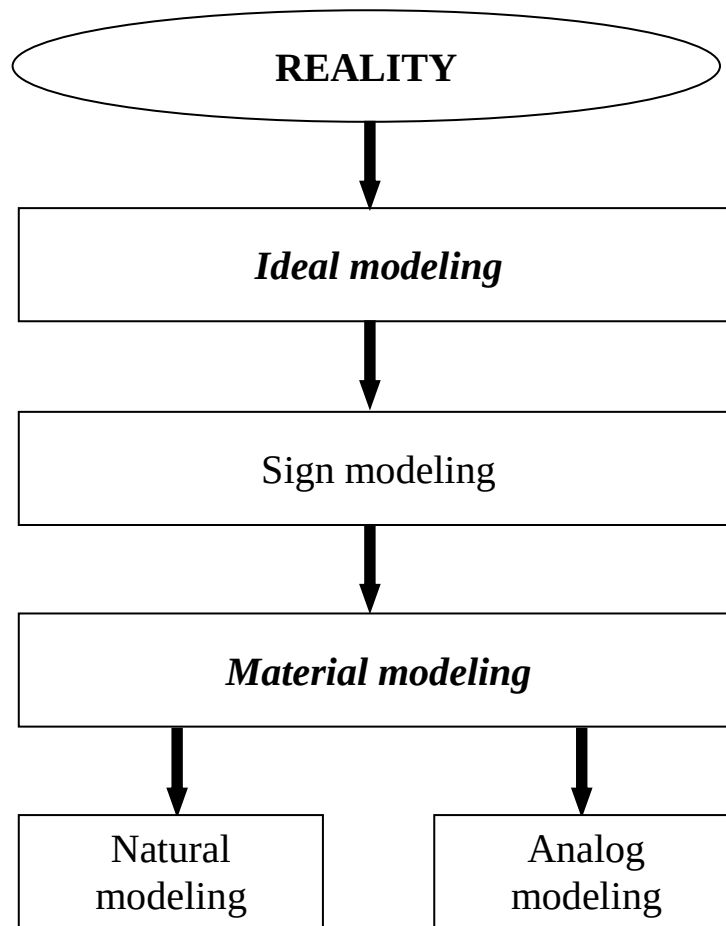


Figure 1.3 – Types of modeling

1.4.1 Material modeling

Material modeling involves the examination of an object through its material counterpart, replicating the fundamental physical, geometric, dynamic, and functional attributes of the object.

The main types of material modeling are *natural* and *analog*. Both types of modeling are based on the properties of geometric or physical similarity.

Natural modeling represents an experimental approach where a real object is compared with its enlarged or reduced material analogue, facilitating research (typically in laboratory settings) through the subsequent transfer of properties from the model to the object, grounded in similarity theory.

Examples of full-scale models include mock-ups in architecture and ship models in shipbuilding. It should be noted that it was with full-scale models of ships in the mid-19th century that modeling began to develop as a scientific discipline, and the models themselves began to be actively used in the design of new technical devices. At present, full-scale modeling methods are widely used in shipbuilding, aircraft manufacturing, automobile manufacturing, rocketry and other areas. For example, when developing a new aircraft, experiments with

full-scale models tested in a wind tunnel are of great importance. The studies conducted allow us to examine the features of air flow around the fuselage, to find the most rational shape of the hull and individual units.

Analog modeling refers to the representation of processes and phenomena with distinct physical characteristics that are described using a uniform formalism.

Analog modeling is based on the coincidence of mathematical descriptions of different objects. Examples of analog models include electrical and mechanical oscillations, which, from a mathematical perspective, are characterized by identical relationships yet relate to qualitatively different physical processes. Under certain assumptions, the processes of heat propagation within a body, impurity diffusion, and liquid infiltration can be considered analogous.

Models of physical and analog types serve as material representations of real objects, closely connected with them by their geometric, physical and other characteristics. In fact, study of such models essentially involves executing a series of natural experiments, utilising a physical or analog model in place of the real object.

The peculiarities of material modeling include the difference between the conditions for implementing the model process and the conditions inherent in the original process, which are selected based on the convenience and simplicity of the study. Since there is no need to preserve the dimensions of structures, time, and intensity of the influencing factors during material modeling, there is a real opportunity to intensify the research process and obtain the necessary results with lower material and time costs. However, the conditions of material modeling cannot be chosen absolutely arbitrarily. Certain similarity relationships must be preserved between the original process and the model process, guaranteeing the possibility of using the information obtained by modeling to adequately assess the properties of the original being studied. Thus, material modeling is still associated with significant costs.

1.4.2 Ideal modeling

Ideal modeling differs from material modeling in that it relies not on a materialized analogy of an object and a model, but on an analogy of an ideal model and is always theoretical in nature.

Sign modeling involves the use of symbolic images, diagrams, graphs, drawings, hieroglyphs, runes, sets of symbols as models, which also include a set of laws and rules that govern the operation of specific symbolic formations and elements.

Examples of such models include any language, such as oral and written human communication, algorithmic and chemical formulas, paintings, musical notation, etc.

Modeling using mathematical relationships (mathematical modeling) is also an example of sign modeling.

Mathematical modeling is an ideal symbolic modeling, in which the description of an object is carried out in the language of mathematics, and the study of the model is performed using certain mathematical methods.

In fact, all modern sections of physics are devoted to the construction and study of mathematical models of various physical objects and phenomena.

Mathematical modelling approaches are frequently the sole viable options. This is currently one of the most effective and commonly employed methodologies of scientific inquiry.

It is important to highlight the benefits of mathematical modeling relative to a natural experiment:

- cost-effectiveness;
- the capability to model hypothetical (not existent in nature) objects;
- the ability to execute scenarios that are hazardous or challenging to replicate in nature;
- the potential to alter the temporal scale.

The aforementioned definition of mathematical modeling cannot be regarded as the only interpretation. Any mathematical model designed for scientific inquiry enables the determination of parameter values of the object or phenomenon under investigation, based on the provided initial data. Consequently, it can be assumed that the core of any such model involves translating a certain set Ω_X , representing the input parameters X , onto the set of values Ω_Y of the output parameters Y . This situation enables the consideration of a mathematical model as a specific mathematical operator A , leading to the formulation of the following definition.

We define a mathematical model as any operator A that, based on the corresponding values of the input parameters X , determines the output values of the parameters Y of the modeled object:

$$A: X \rightarrow Y, X \in \Omega_X, Y \in \Omega_Y,$$

where Ω_X and Ω_Y are sets of admissible values of input and output parameters for the modeled object. Depending on the nature of the modeled object, the elements of the sets Ω_X and Ω_Y can be any mathematical objects (numbers, vectors, tensors, functions, sets, etc.).

The concept of an operator in the given definition can be interpreted quite broadly. It can be either a function that connects input and output

values, or a mapping that represents a symbolic record of a system of algebraic, differential or integral equations. Finally, it can be an algorithm, a set of rules or tables that ensure the finding (or establishment) of output parameters based on given initial values.

1.4.3 Classification of mathematical models

It is possible to divide mathematical models into different classes depending on:

- the complexity of the modeling object;
- the model operator (submodel);
- the input and output parameters;
- the method of implementing the model.

All modeling objects can be divided into two groups: simple and system objects. In the first case, the internal structure of the object is not considered during modeling, and its constituent elements or subprocesses are not identified.

The division of research objects into “simple” and “complex” is arbitrary. Since it is impossible to single out “elementary building blocks” for any known processes, phenomena, or material bodies, any research object can be considered infinitely complex. Simplification of its structure during the development of a model is accomplished by discarding connections between the elements that make up the object that are insignificant for achieving the research goals set at the given moment.

The classification of mathematical models depending on the methods of model implementation is shown in Figure 1.4.

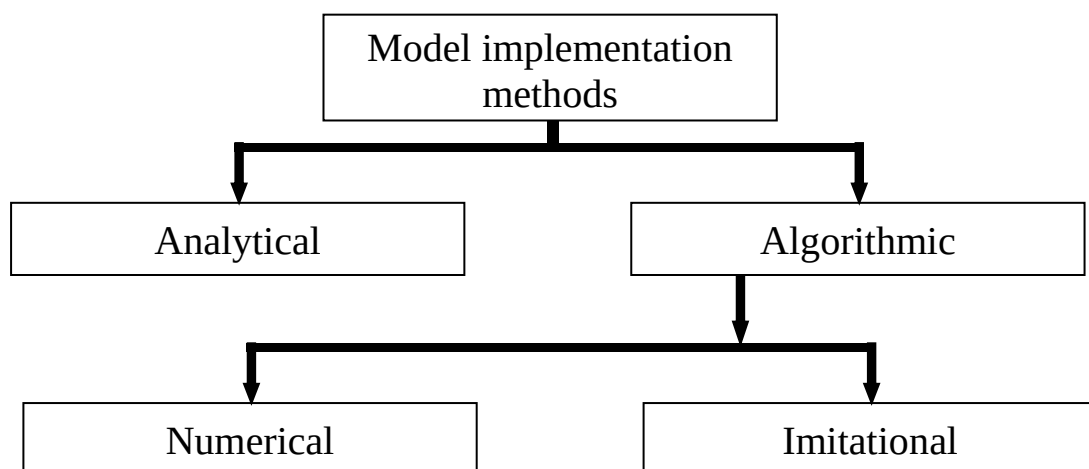


Figure 1.4 – Classification of mathematical models depending on the implementation methods

It was noted above that any mathematical model can be considered as a certain operator A , which is an algorithm or is defined by a set of equations – algebraic, ordinary differential equations (ODE), systems of ODEs (SODEs), partial differential equations (PDEs), integro-differential equations (IDEs), etc. (Figure 1.5).

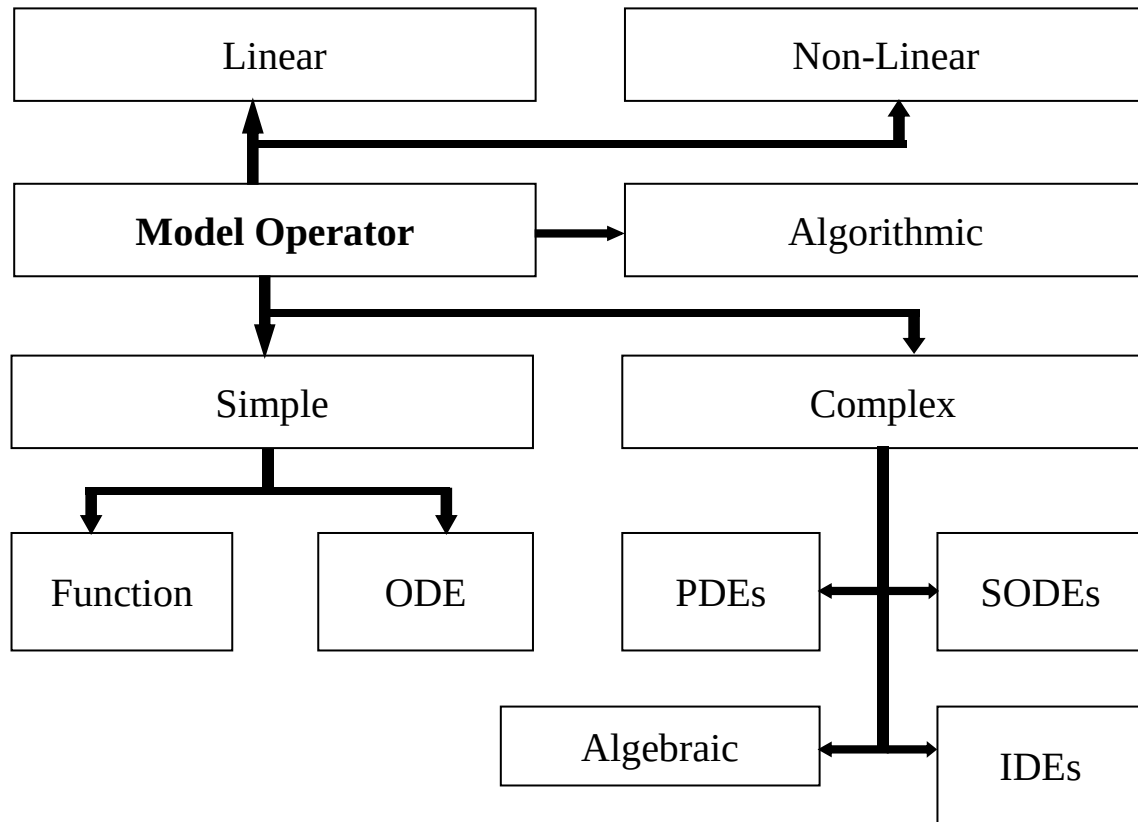


Figure 1.5 – Classification of mathematical models depending on the model operator

A model implementation method is classified as analytical if it yields output parameters as analytical expressions, defined as expressions utilising no more than a countable set of arithmetic operations and limit transitions.

Analytical methods of model implementation are advantageous as they provide the examination of the properties of the modeling object with reduced computational costs, employing established mathematical techniques for analysing analytical functions. The application of analytical methods must be feasible without a computer. Furthermore, understanding the analytical expression for the desired parameters enables the examination of the object's fundamental features, its qualitative behavior, and the formulation of new hypotheses regarding its internal structure.

Figure 1.6 illustrates the classification of mathematical models based on the model parameters.

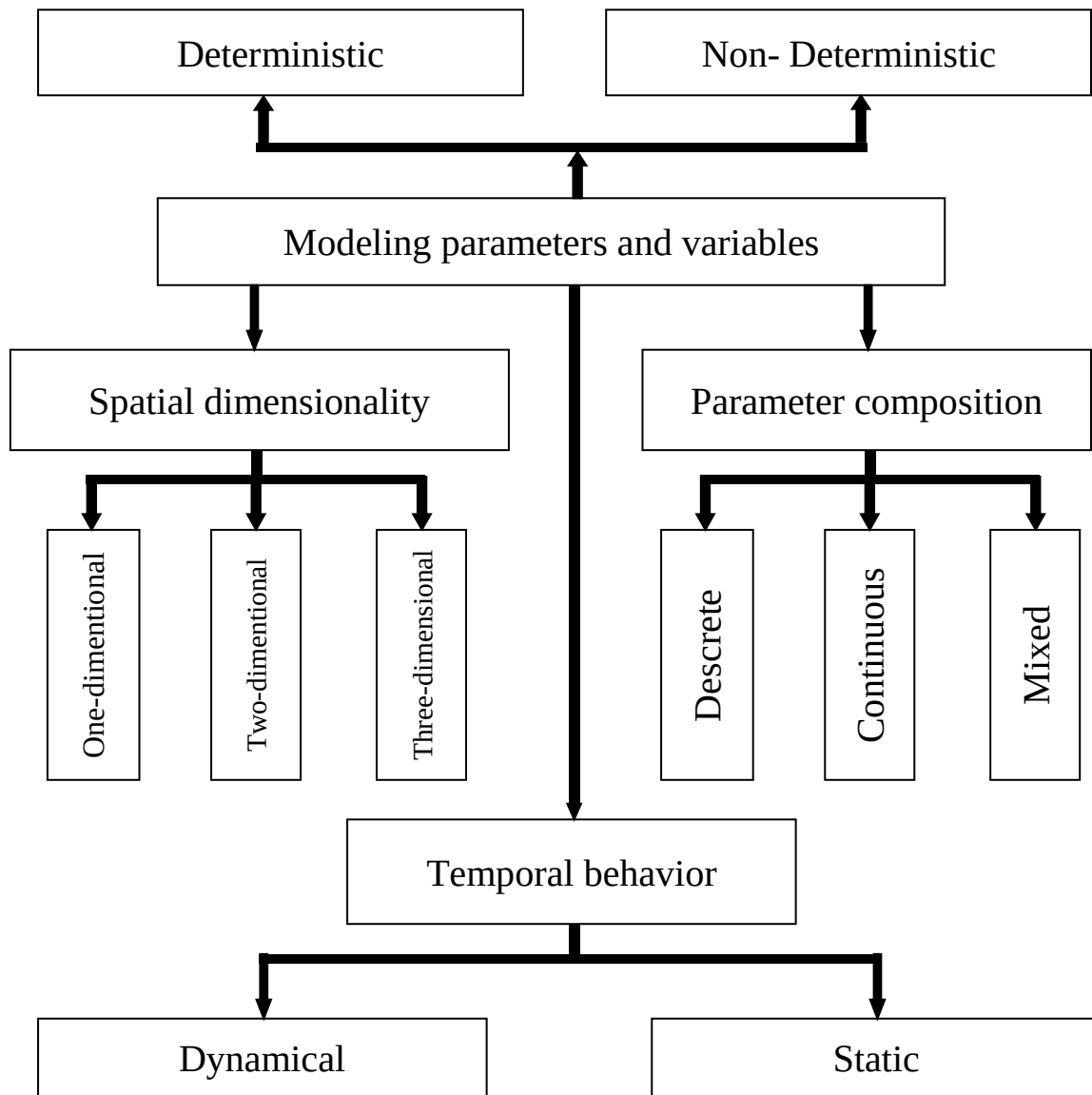


Figure 1.6 – Classification of mathematical models depending on the model parameters

It is noteworthy that the effectiveness of analytical methods is heavily influenced by the advancement of the relevant areas of mathematics.

The current increase in interest in analytical methods for model implementation is linked to the advent of mathematical computer software (Maple, MathCAD, MATLAB, Mathematica, etc.). The scope of issues addressed by these packages is extensive and continually broadening, encompassing elementary mathematics, symbolic manipulation of polynomials, derivatives and integrals, vector and matrix operations, field theory, vector analysis, the finite element approach, among others. The utilization of such software not only streamlines the process of acquiring an analytical answer but also enhances the following analysis of the derived solution through numerous visualizers.

Unfortunately, the mathematical methods available at present permit the derivation of analytical solutions only for relatively straightforward mathematical models within a limited spectrum of parameter values. In many instances, when examining models, it is essential to employ algorithmic approaches that yield only approximate values of the desired parameters.

The numerical approach involves substituting the model's set of mathematical relationships with a finite-dimensional analogue. This is typically accomplished through the discretization of the initial relationships, meaning the shift from functions of a continuous argument to functions of a discrete argument. Following the discretization of the initial problem, a computational algorithm is developed, consisting of a series of arithmetic and logical operations executed on a computer, enabling the solution to the discrete problem to be achieved in a finite number of steps. The solution identified for the discrete problem serves as an approximate solution to the original mathematical problem.

A fundamental requirement for a computational algorithm is to achieve a solution to the original problem with specified accuracy within a finite number of steps.

The numerical approach discretizes the established system of mathematical relationships, while the simulation approach deconstructs the subject of study into distinct components. The comprehensive system of mathematical relationships for the entire object-system is not recorded; instead, it is substituted by an algorithm that simulates its behavior, considering the interactions among the models of the individual system elements. Models of particular elements may be either analytical or algorithmic.

Algorithmic models using both numerical and simulation approaches fail to produce solutions in analytical form, hence complicating the analysis of the modeling results. The efficacy of such models is reliant upon the availability of computing equipment, specifically the power and speed of the computer. An obvious advantage of algorithmic models is the lack of fundamental restrictions on model complexity, enabling their application to the analysis of systems of arbitrary complexity.

The application of a mathematical model developed by algorithmic techniques resembles performing experiments on a real object; however, instead of an actual experiment, a computer experiment is executed with the model. Based on a defined set of starting parameter values of the model, the computational experiment yields a specific set of approximation values for the desired parameters. To analyze the behavior of an object given a novel set of initial data, it is necessary to do a new computational experiment.

1.4.4 Classification of computer models

Computer models are classified according to the following criteria:

- structure;
- purpose;
- mathematical model employed;
- degree of approximation to the real-world object;
- set of properties under investigation;
- spatial dimension of the modeling area;
- method of discretization concerning spatial variables;
- techniques for solving differential equations;
- temporal dependence of properties;
- physical processes being simulated;
- modeling speed;
- application of modeling outcomes.

Depending on the mathematical model used in the computer model, a distinction is made between:

- analytical computer models describing the properties of the modeling object through a system of equations for which an analytical solution can be derived in explicit form (for example, individual models of solid mechanics);
- numerical computer models that depict the properties and behavior of the modeling object via a system of equations for which solutions are obtained using computational mathematical techniques (such as difference methods, finite element methods, iterative algorithms, etc., applied to problems in solid mechanics, heat transfer, hydrodynamics, electrodynamics, etc.);
- statistical computer models describing the properties and behavior of the modeling object using methods of probability theory and mathematical statistics (for example, mass service models, models describing the dynamics of changes in warehouse stocks).

Based on the structure of the computer model, a distinction is made between simple and composite (complex) computer models, which consist of a set of interconnected models describing a single modeling object.

Depending on the intended use of the computer model, we can categorise them into two types:

- computer models of engineering analysis, which aim to describe the properties and behavior of the modeling object while considering physical processes;
- virtual-simulation computer models, intended to model the processes of interaction of elements of systems of interacting objects in order to study the behavior of these systems.

According to the degree of approximation of a computer model's representation to a real-world object, a distinction is made between:

- simplified computer models used to develop elements of the design of the modeling object;
- precise computer models employed for conducting virtual experiments.

Following the set of properties under study, computer models are divided into:

- simple computer models (e.g., for studying a single property);
- combined computer models that allow studying a set of different properties of the modeling object.

Based on the spatial dimension of the modeling area, computer models can be categorised as follows:

- zero-dimensional computer models with lumped parameters;
- one-dimensional computer models;
- two-dimensional computer models;
- three-dimensional computer models.

The method of discretization by spatial variables classifies computer models into the following types:

- finite-difference;
- finite-element;
- finite-volume.

In the context of solving differential equations, computer models can be divided into:

- deterministic computer models;
- stochastic computer models (Monte Carlo method).

According to the dependence of properties on time, computer models are distinguished as:

- stationary (static);
- non-stationary (dynamic).

Based on the simulated physical processes, computer models can be grouped into:

- aerodynamics;
- hydrodynamics;
- thermal conductivity;
- strength.

The given list of classification features can be expanded depending on the modeled processes relevant to a particular subject area.

By considering the speed of execution in modeling, computer models are categorised into real-time models and non-real-time models. Real-time

computer models are commonly employed in control systems, simulators, and similar applications. Computer models utilized for engineering analysis are employed in a non-real-time context.

Computer models are classified based on their application of modeling results into the following types:

- models used product development and acceptance, including delivery, preliminary, state, qualification, serial, and other testing phases, as well as during subsequent stages of the product life cycle;
- models employed for research aimed at determining development strategies and optimizing the structure of systems involving interacting objects;
- models designed to support decision-making processes when engaging with systems of interacting objects;
- models for training purposes, functioning as simulators to enhance skills in operating systems of interacting objects.

Self-Assessment Questions

- 1 What types of modeling can be distinguished at the empirical and theoretical levels of research?
- 2 What is the difference between natural and analog modeling?
- 3 Classify mathematical models depending on the methods of model implementation.
- 4 Categorize mathematical models depending on the model operator.
- 5 Classify mathematical models based on the model parameters.
- 6 Categorize computer models.

Topic 1.5 Stages of computer mathematical model development

The introduction of computer technology into all spheres of human activity has led to the widespread use of mathematical models. Note that a computer is only “hardware”, and what makes it “smart” and useful are programs, which in most cases are implementations of algorithms of the corresponding mathematical models. Therefore, it is necessary to create a large number of diverse mathematical models with wide capabilities.

The need for mass model building requires the development of a certain set of rules and approaches that would reduce the costs of model development and reduce the likelihood of errors that are difficult to eliminate later. Such a set of rules could be called a technology for creating mathematical models.

At the same time, as already noted, the basis for a computer model is a mathematical model implemented in the form of a program code and data that determine a specific modeling object.

The process of constructing any computer mathematical model can be represented by a sequence of stages, shown in Figure 1.7.

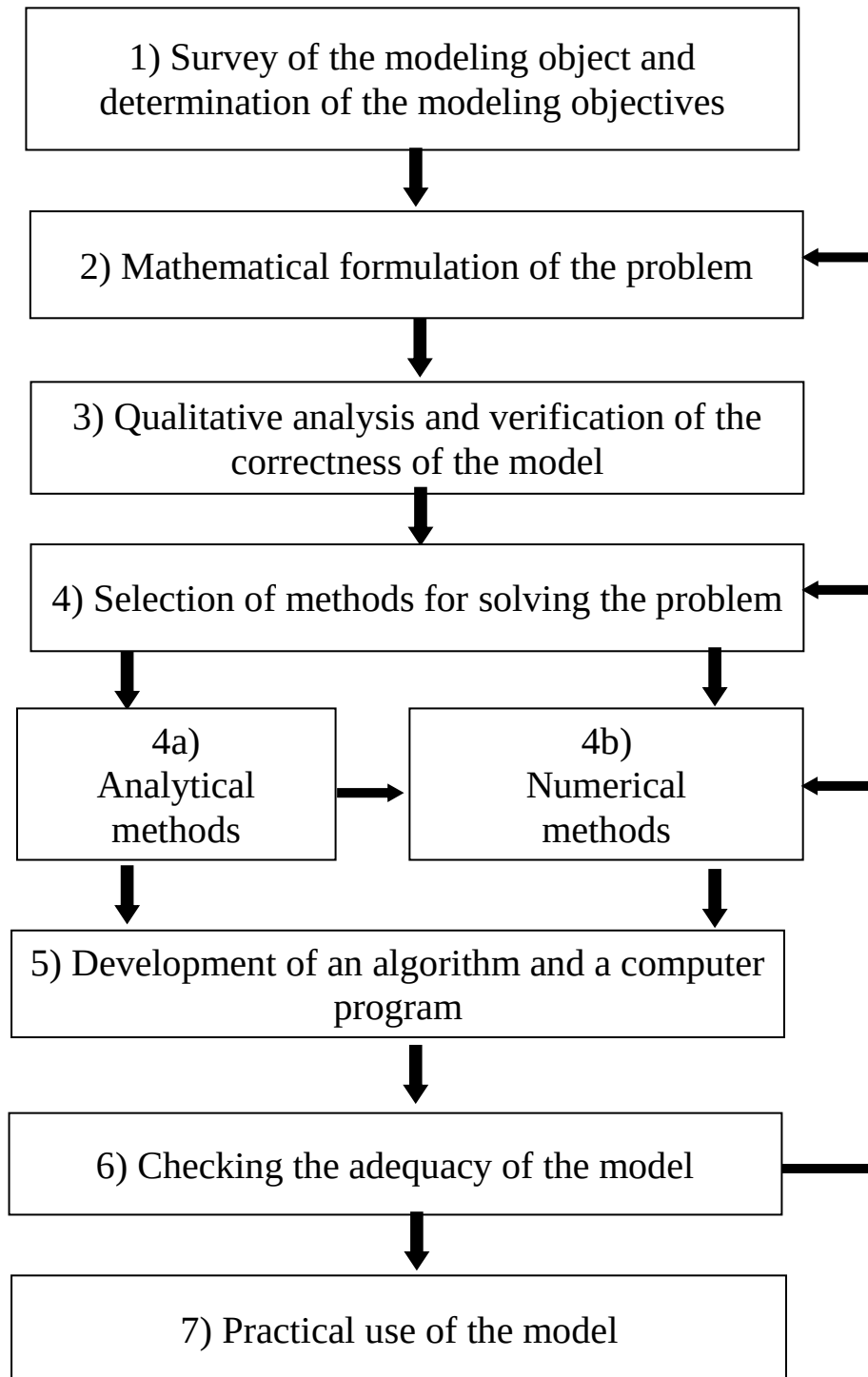


Figure 1.7 – Stages of constructing a computer mathematical model

The *survey stage* includes the following works:

- survey of the modeling object and determination of the modeling objectives to identify the main factors and mechanisms influencing the behavior of the modeling object, determination of the corresponding parameters that allow describing the modeled object;
- collection and verification of available experimental data on similar objects, conducting additional experiments if necessary;
- analytical review of information, analysis and comparison of previously constructed models of this object (or similar to the object under consideration);
- analysis and generalization of all accumulated material, development of a general plan for creating a mathematical model.

All the material collected as a result of the survey about the knowledge accumulated to this point about the object, the requirements for the implementation of the model and the presentation of the results are drawn up in the form of a technical assignment for the design and development of the model.

In the second stage, a conceptual, or “natural science” (physical, chemical, biological, etc.) formulation of the modeling problem is developed. A conceptual formulation of the modeling problem is a list of the main questions to be solved during modeling, formulated in terms of specific disciplines (physics, chemistry, biology, etc.), as well as a set of hypotheses regarding the properties and behavior of the modeling object.

The completed conceptual formulation allows us to formulate a mathematical formulation of the modeling problem, including a set of various mathematical relationships describing the behavior and properties of the modeling object.

The *mathematical formulation* of a modeling problem is a set of mathematical relationships that describe the behavior and properties of the modeling object.

In the third stage, a qualitative analysis and verification of the correctness of the model is carried out. To control the correctness of the resulting system of mathematical relationships, a number of mandatory checks must be performed:

- dimensionality control;
- order control, consisting of a rough estimate of the comparative orders of the added quantities and the exclusion of insignificant parameters;
- control of the nature of dependencies, which entails verifying that the direction and rate of change of the model’s output parameters, derived from the established mathematical relationships, align properly with the “physical” interpretation of the model being analysed;
- extreme situations control – examining the nature of mathematical relationships and modeling outcomes when model parameters or their combinations approach their highest permitted values, typically zero or infinity;

- boundary conditions control, which involves verifying that the boundary conditions are actually imposed, are used in the process of constructing the desired solution, and that the values of the output parameters of the model actually satisfy these conditions;

- control of physical meaning – assessment of the physical or alternative, based on the task's nature, significance of the initial and intermediate relationships that emerge during the model's construction.

In the fourth stage, methods for addressing the problem are selected. When using developed mathematical models, it is generally necessary to determine the relationship of certain previously unknown parameters of the modeling object that fulfil a specific system of equations. Therefore, the search for a solution to the problem is simplified to identifying certain dependencies of the desired values on the initial parameters of the model. As previously mentioned, all approaches for addressing issues that constitute the “core” of mathematical models can be categorised into analytical and numerical methods. The process of developing a software implementation is just as intricate as all the earlier stages of creating a mathematical model.

In the fifth stage, the creation of a software implementation of the mathematical model takes place, which can be broken down into several steps:

- designing the structure;
- coding the algorithm;
- testing and debugging.

Most programs that implement mathematical models consist of three main parts:

- preprocessor (preparation and verification of the initial data of the model);
- processor (solving the problem, implementing the computational experiment);
- postprocessor (displaying the results obtained).

In the sixth stage, the model's adequacy is evaluated. The adequacy of the mathematical model will be interpreted as the extent to which the results derived from the developed model align with the experimental data or test task outcomes. Before proceeding to the adequacy check of the model, it is essential to ensure the proper integrated functioning of all algorithms and programs within the model. This includes conducting independent testing and debugging of each individual algorithm, such as the software modules that implement the numerical method employed.

The verification of the model's adequacy aims to achieve two objectives:

- 1) to ensure the validity of the set of hypotheses formulated at the stages of conceptual and mathematical formulations. It is necessary to

proceed to testing the hypotheses only after checking the used solution methods, comprehensive debugging and eliminating all errors and conflicts associated with the software;

2) to establish that the accuracy of the results corresponds to the accuracy specified in the technical task.

The developed mathematical model is verified by comparing it with available experimental data from the real object or with results from other established and reliable models. In the first instance, verification is discussed through comparison with the experiment, while in the second instance, it is through comparison with the outcomes of solving the test problem.

The accuracy of modeling is determined by the specific requirements placed on the model and its intended purpose. In this situation, it is important to consider the precision of acquiring experimental results or the characteristics of the test task formulations. In models designed for evaluation and rough calculations, an accuracy of 10–15 % is considered acceptable. In models utilized in control and monitoring systems, the necessary accuracy can range from 1 to 2 %.

The final stage involves the practical application of the developed model and the analysis of the results obtained from the modeling process.

Self-Assessment Questions

- 1 Describe the process of building any computer model.
- 2 What are the stages involved in constructing a computer model?
- 3 What are the objectives of assessing the model's adequacy?
- 4 What methods are used to verify the developed mathematical model?

2 CALS/PLM, AND CAE SYSTEMS

Topic 2.1 CALS/PLM systems

2.1.1 Key terms and definitions

SRI – scientific research institute.

R&D – research and development work.

CNC (computer numerical control) – a system of digital computing devices for controlling production equipment.

Digital twin – a digital interactive copy of a real physical object, created for its effective management and optimization of business processes of its creation.

BOM (bill of materials) – a list of product specifications, including the materials and components along with their quantities required for the production of a specific product.

CAD (computer-aided design) – computer-aided design system.

CAE (computer-aided engineering) – computer systems for solving various engineering problems: calculations, analysis or simulation of physical processes.

CAI (computer-aided inspection) – computer systems for the quality control of physical models and products. Operates alongside measurement tools.

CALS (continuous acquisition and life cycle support) – continuous information support for product life cycle processes.

CAM (computer-aided manufacturing) – computer systems for preparing control programs for CNC machines.

CPC (collaborative product commerce) – a digital environment for managing corporate information of participants in the production chain to coordinate the processes of design, production and after-sales service throughout the product life cycle.

SCM (supply chain management) – a digital supply chain management system.

ERP (enterprise resource planning) – a digital enterprise resource planning system.

MES (manufacturing execution system) – an operational planning and management system for production processes.

MRP (material requirements planning) – computer systems for planning material requirements.

PCM (product configuration management) – a system designed to digitally define the functional and physical characteristics of a product, ensuring their compliance with specified requirements as well as design and operational documentation throughout the product's service life.

PDM (product data management) – systematic organization and administration of product information.

PLM (product lifecycle management) – systematic management of a product's life cycle.

SCADA (supervisory control and data acquisition) – a digital dispatch control system for production processes. It is designed to ensure real-time operation of systems for collecting, processing, displaying and archiving information about the monitoring or control object.

2.1.2 History of CALS/PLM systems

1960s. The initiation of integrating computing technology into production processes. Computers were used to solve specific problems that related to different stages of the product life cycle. The integration of computing technology facilitated the automation of accounting and management functions within the automated production management system (APMS). Simultaneously, automated process control systems (APCS) emerged.

1970s. The first systems for design engineering emerged as CAD systems, which used computer technology in the design and technological preparation processes, producing two-dimensional product drawings. A system for planning material requirements was developed to enhance the logistics of enterprise resources.

The concept of integrated computer production systems was developed due to the growing number of computer systems within enterprises and the challenges associated with information exchange among them. The USSR State Standard established a method for detailing products within the framework of automated design and production. This initiative aimed to create a cohesive system for the formalized presentation of design and technological information, as well as to develop an information language.

Early 1980s. The advent of 3D CAD systems marked a significant shift from 2D CAD systems, enabling engineers to create complex solid surfaces in three dimensions. Design methods progressed from static 2D drawings featuring various views and sections to dynamic 3D virtual geometric models of products.

The emergence of CAE systems occurred alongside the advancement of 3D CAD systems. CAE systems expanded the functionality of computer

design by assisting engineers evaluating the performance and behavior of a three-dimensional model of a product under conditions that closely resemble real operation, all while minimising significant time and financial investments. The development of CAM systems enabled engineers to create control programs for CNC machines derived from the results of engineering analysis.

1985. CALS technology was developed for the US military-industrial complex, which standardized the electronic representation of products and the exchange of technical and commercial information. CALS technology streamlined processes and reduced costs associated with information support for logistics.

Mid-late 1980s. The emergence of Product Data Management (PDM) systems – software platforms designed to support CAD, CAM, and CAE systems. Early PDM systems focused on providing users with centralized access to product information while ensuring data integrity through a unified repository.

The transition from PDM systems to CPC systems was driven by the expansion of contract manufacturing and advancements in Internet technologies, which established the essential infrastructure for the creation of lightweight and universal user interfaces with minimal support costs. The advancement of Internet technologies allowed for a reduction in the ownership costs of CPC systems for enterprises, facilitating collaboration with international suppliers and customers. CPC systems were limited to managing only technical and design documentation, which proved insufficient for comprehensive product life cycle management.

The Production Resource Planning (PRP) system was replaced by the Enterprise Resource Planning (ERP) system. During this period, two other critical systems emerged: Customer Relationship Management (CRM) and Supply Chain Management (SCM). ERP, CRM, and SCM systems each addressed distinct stages of the product life cycle. However, Product Data Management (PDM) remained separate from ERP and CRM, as it was originally developed to handle technical and design data rather than business or operational processes.

2005. Dassault Systems introduced the PLM concept, marking the evolution of CALS technologies. PLM programs were created to enhance the functionalities of PDM programs. PDM concentrated on information management, while PLM served as a process that facilitated the collection, organization, and reuse of information across the product life cycle.

Second half of the 2000s – present. The integration of PLM and ERP systems has become a key industry trend. This period also saw the

emergence of the digital enterprise concept, where AI-driven integration between PLM and ERP systems forms the operational backbone of modern manufacturing organizations

2.1.3 Structure of CALS/PLM systems

The development and production of modern products requires the use of software of various directions. The stages of the life cycle of industrial products and their automation systems are presented in Figure 2.1.

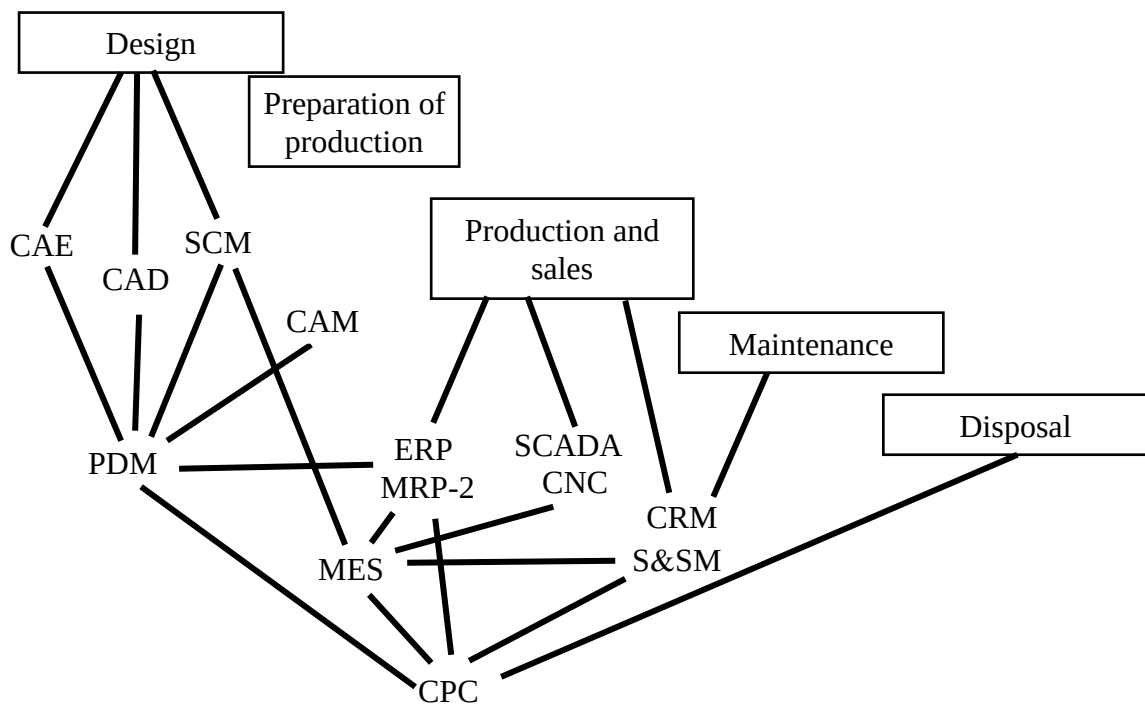


Figure 2.1 – Stages of the life cycle of industrial products and their automation systems

CAD is an automated product design system that utilizes information technologies to perform engineering design functions. It serves as both an organizational framework and technical platform for design automation. The term CAD typically refers to software applications used for automated design processes.

CAE is a program for solving engineering problems: calculations, analysis and simulation of physical processes. CAE algorithms are based on numerical methods for solving differential equations. Modern CAE are used together or integrated into CAD. CAE allows evaluating the behavior of a computer model of a product in real operating conditions and check the operability of the product.

CAM is used to program manufacturing processes and CNC machines. CAM employs models and assemblies developed in CAD to generate tool paths. CAM refers to both the process of computerized production preparation and the software and computing systems used by process engineers.

PDM is a product engineering data management system that ensures coordination of CAE/CAD/CAM systems.

SCM is a digital supply chain management system. SCM automates and manages procurement, product movement within an enterprise, and product distribution. Supply chain management includes product manufacturing, raw material supply, location, inventory, transportation, and information.

ERP is a digital enterprise resource planning system. ERP manages business processes such as accounting, purchasing, project management, risk management, compliance, planning, budgeting, forecasting, and financial reporting. ERP integrates multiple business processes and transfers data between them, eliminates duplication, and ensures data integrity through a “single source of truth.”

MRP is used for enterprise resource planning and includes functions for the automatic creation of purchase orders and the production of essential components. MRP optimizes the efficiency of material and component delivery, ensuring that the necessary materials for production arrive concurrently. MRP accelerates the delivery schedule by prioritizing the materials that are required immediately while postponing unnecessary receipts.

MES is a system for operational management of manufacturing processes that uses data on the status and movement of orders and materials for production at production sites.

CPC is a digital environment for managing corporate information of participants in the production chain for coordinating the processes of design, production and after-sales service throughout the product life cycle.

SCADA is a digital dispatch control system for production processes. It is designed to collect, process, display and archive information about the object of monitoring or control in real time.

CNC – computer numerical control.

CRM is a digital customer relationship management system. CRM generates documents using a template, sets tasks for managers at each stage of the transaction, sends messages and letters to clients, creates reports on specified indicators, calculates the cost of services, tracks important dates and deadlines.

S&SM is a sales and service management system. In addition to those indicated in the diagram, the following automation systems are also used within the PLM systems.

CAI refers to computer systems for quality control of physical models and products, with the digital model acting as a standard. They are used in conjunction with measuring instruments.

BOM includes details regarding the assemblies, parts, components, and raw materials necessary to produce a single unit of the finished product, along with the quantities of each item. The information is organized hierarchically, with the completed product at the apex and the assemblies, parts, components, and raw materials at the subordinate levels. The requisite quantity of each item needed to finalize one unit of the subsequent assembly level is specified.

PCM establishes the functional and physical characteristics of a product, their compliance with specified requirements and design and operational documentation throughout the product's service life.

ECM is a system of digital change management in engineering systems, their planning, implementation and evaluation depending on the influence of various factors.

Thus, PLM systems represent the organization of development, production, after-sales service, and product operation processes through the informational support of life cycle processes, predicated on the standardization of information presentation methods at each life cycle stage and the implementation of paperless electronic data exchange.

Today, modeling tools alone are not enough to develop products, so the PLM concept goes beyond CAD, CAM, CAE, PDM. 70 % of all product costs are spent on the design cycle (PLM), and the rest on operational management and logistics (ERP). The use of ERP platforms begins where technological information is not important: resource management, financial flows, stocks, etc. PLM and ERP complement each other. PLM is responsible for product development. ERP is responsible for after-sales service, the information for which comes from the PLM platform. An important role in this is played by the exchange of the latest information between platforms.

The application of PLM is progressively advancing in the development of complex, science-intensive products, which are designed and produced by integrated industrial entities, including research institutes, scientific and production associations, design bureaus, and enterprises involved in technical maintenance, repair, and disposal. The evolution of the PLM idea has resulted in the construction of a “virtual enterprise” – a novel methodology for executing intricate projects that consolidates players in the product life cycle on a contractual basis inside a unified information framework. This strategy enables the implementation of collaborative projects for the creation, production, sales, and servicing of high-tech items.

2.1.4 CALS/PLM systems: strategic objectives and core functions

Strategic objectives of PLM implementation:

- revenue growth;
- cost reduction associated with the manufacturing and selling of various products.

Core functions of PLM systems in product development:

1 **Product data management.** Centralizes all product-related information for production, logistics, sales, operations, and maintenance applications.

2 **Management of fixed assets.** Ensures a high level of control and accuracy of equipment operation for an uninterrupted production cycle.

3 **Program and project management.** Provides information on interrelated activities for effective project management: the budget of the designed product, planning the necessary production capacities, establishing logistics and communication links.

4 **Collaboration throughout the product life cycle.** Increases the efficiency of product development, reduces its cost and time to market due to continuous feedback between the integrated processes of design, production, sales and service.

5 **Quality control.** Implements comprehensive quality control across all lifecycle stages:

- marketing;
- design;
- development of technical specifications;
- material and technical supply;
- purchasing;
- development of production processes;
- production;
- testing;
- certification;
- installation;
- operation;
- maintenance;
- disposal.

6 **Environmental and labor protection.** Facilitates cost reduction and mitigates production hazards while adhering to environmental and labour protection legislation by decreasing the time allocated to completing safety engineering forms and instructions.

2.1.5 Implementation of CALS/PLM systems

The main industries for implementing PLM systems are:

- aerospace and defense;
- automotive and transport;
- consumer goods and retail;
- electronics and semiconductors;
- energy;
- heavy engineering and industrial equipment;
- shipbuilding;
- medicine and pharmaceuticals;
- construction and urban infrastructure;
- scientific research;
- oil and gas industry.

The array of products using PLM is extensive, encompassing agricultural machinery, aerospace equipment, automobiles, chemicals, computers, consumer electronics and appliances, electrical equipment, elevators, escalators, food, furniture, medical and industrial equipment, medicines, ships, clothing and footwear, software, telecommunications equipment, trains, turbines, watches and more.

2.1.6 Advantages and Limitations of CALS/PLM Technologies

Key advantages PLM implementation:

- accelerated time-to-market;
- cost optimization;
- enhanced quality assurance;
- rapid exchange of electronic data and documents;
- simultaneous execution of intricate projects by multiple teams;
- diminished errors in product development;
- decreased production defects;
- comprehensive reliability in production accounting;
- assurance of superior product quality;

– within the framework of PLM, it is possible to develop a “digital twin” that combines the geometry of mechanical, electronic and software components with the physics of processes that model the behavior of a physical object with a high degree of reliability. Using a digital twin, you can perform complex engineering calculations and search for the optimal design. Previously, performing such tasks required a large number of intermediate actions, which greatly complicated the process of finding the

optimal engineering and scientific solution. Today, with the use of PLM and digital twins, the search for a solution occurs in a single environment, in a short time and with a given result.

The main disadvantages of using the PLM concept.

Additional costs. The implementation of a complex system may necessitate the discontinuation of existing enterprise systems, resulting in additional costs. Consequently, the customer may become more dependent on the sole supplier of the chosen solution.

The problem of data integrity. Given the discussion on information management across the whole life cycle, integration is essential – specifically among the design, technological preparation of production systems, and the enterprise management system. To address this issue, it is essential to establish a singular product classification system, standardised reference materials (including components) across the entire organization, and facilitate operations in a distributed mode. Current observations indicate that this issue remains unresolved in the majority of domestic enterprises. Furthermore, there is typically a gap in the data flow between the design and technological subsystems, excluding the preparation of programs for CNC machines. As a result, an initial step in the implementation of PLM systems must involve the establishment of standardized reference materials within the organization.

Data traceability. An important problem of data traceability of a product throughout its life cycle is that different systems (design, technological, production) use different sets of data. Currently, the systems provided by developers typically lack comprehensive end-to-end traceability for a specific product throughout all phases of its development, production, and support. This is primarily true except for instances of single and small-scale production, which are mainly associated with defence applications and particularly critical civilian products. In the context of serial and mass-produced products, the discussion primarily revolves around the traceability of a product batch.

Openness. As domestic enterprises and design organizations increasingly participate in global projects, they pay special attention to cooperation with suppliers and customers. At the same time, the use of modern communication technologies has both a positive and negative effect on enterprises. On the one hand, using the capabilities of the Internet is very attractive for creating product catalogs and online stores, on the other hand, there is a need for reliable protection of confidential information from outside penetration.

Organizational problems. The implementation of enterprise-wide systems invariably impacts the interests of many employee groups, while the integration of design, technological, and management data within

a singular system necessarily results in conflicts of interest. Practical evidence indicates that conflicts predominantly arise from the allocation of access rights to information among services (the “who is in charge” issue) and the transition of data, once owned by individual employees, to the corporate level (the “irreplaceable people” issue). In addition, when implementing systems of this level, enterprises encounter the necessity to establish their own standards, methodologies, and protocols. In this context, unlike the implementation of CAD, replicating another’s experience is virtually unattainable for several reasons, including variations in work specifics, business processes, infrastructure, and the reluctance of colleagues who have already navigated this path to share information. Moreover, the introduction of a new system frequently generates resistance from staff accustomed to previous systems.

Without solving the above-mentioned problems, the implementation of a complex information system is practically impossible. Furthermore, efforts to consolidate all functional capabilities into a single system result in the formation of a cumbersome information entity, which is impractical for specialized user groups to solve professional issues.

Self-Assessment Questions

- 1 What terms and definitions related to CALS/PLM systems do you know?
- 2 What are the key stages of CALS/PLM system development that you can identify?
- 3 Describe the structure of a typical CALS/PLM system.
- 4 What are the main goals and objectives of implementing CALS/PLM systems?
- 5 What advantages and disadvantages of CALS/PLM technology can you identify?

Topic 2.2 SAE systems

According to established terminology, science-intensive software systems are classified as CAE technologies (CAE, Computer Aided Engineering), which constitute the natural science core of the CAD-CAE-CAM triad in the CALS/PLM technology system. CAE systems can be conditionally divided into 2 groups:

- 1 Full-featured engineering analysis systems with powerful tools, large libraries of finite elements, and a multi-purpose focus on solving engineering problems. They have their own geometry modeling tools.

In addition, there is the possibility of importing through industrial standards Parasolid, ACIS, etc. Full-featured CAE systems do not have an associative connection with CAD. Therefore, if during the calculation process there is a need to change the geometry, the user will have to re-import the geometry and enter the data for the calculation. The most famous such system is ANSYS from the American company ANSYS.INC. To create a geometric model, ANSYS uses the Parasolid kernel, integrated into many other commercial software products.

2 Engineering analysis systems, united by a single interface with CAD systems into an integrated design system. They have less powerful analysis tools, but they support associativity with geometry, so they track changes in the model. In addition, such systems include functionality for generating design documentation, i.e., drawings, specifications, etc. These include the CAD/CAM/CAE system NX from Siemens PLM Software, Creo Elements/Pro (ProEngineer) from PTC, CATIA and SolidWorks from Dassault Systèmes, etc.

2.2.1 Capabilities of SAE systems

The benefits of CAE systems include:

- the reduction of development costs by enabling model testing on a computer rather than through costly operational tests;
- the acceleration of product market introduction by minimizing the number of development cycles;
- the improvement of products through the rapid evaluation of numerous concepts and scenarios simultaneously prior to final decision-making, thereby affording additional time for consideration of new designs.

SAE facilitates the execution of:

- strength analysis of components and units based on the finite element method;
- frequency studies;
- thermal and hydrodynamic analysis;
- kinematic studies;
- modeling of processes such as injection molding;
- optimization of products or processes, etc.

The predominant SAE are found in the following sectors:

- mechanical engineering and machine tool manufacturing;
- defence and aerospace industry;
- energy and shipbuilding;
- semiconductor production;
- telecommunications;

- chemical, pharmaceutical and medical sectors;
- construction;
- production of heating, air conditioning, ventilation systems;
- automotive industry.

2.2.2 Market leaders in CAE systems

ANSYS (stock index ANSS) was founded in 1970. The company employs almost 3,000 professionals, with its the headquarters located in Canonsburg (Pennsylvania, USA). ANSYS has always been the predominant player in the CAE business. It took the leading position in 2006. Each year, ANSYS, in a state of dynamic development, enhances its performance.

Since 2009, Dassault Systèmes (stock index DASTY), a leader among PLM companies, holds the second position in the CAE industry. The domain of CAE technology operates under the SIMULIA brand, established with the acquisition of ABAQUS in 2005. The CATIA and SOLIDWORKS packages also offer engineering analysis tools. Consequently, this French corporation, alongside its headquarters in Velizy-Villacoublay, operates two additional geographic centers that impact the advancement of CAE technologies: the SIMULIA division is based in Providence (Rhode Island, USA), and SOLIDWORKS is located in Waltham (Massachusetts, USA) (Table 2.1).

Table 2.1 – CAE market leaders: key software products

Company	Specialization			
	CAD	CAM	CAE	PDM/PLM
ANSYS	SpaceClaim		ANSYS Mechanical, ANSYS CFD	
Dassault Systemes	SolidWorks, CATIA	CATIA	SIMULIA (ABAQUS, fesafe, Isight, TOSCA) SolidWorks Simulation	Enovia (Smarteam), SolidWorks Enterprise PDM
Siemens PLM Software	NX, Solid Edge	NX-CAM	NX-CAE	Teamcenter
PTC	CREO(PRO/Engineer)	CREO	CREO (Pro/Mechanica)	Windchill
Autodesk	AutoCAD, Inventor	Inventor		PLM 360

Siemens PLM Software has the third position in the PLM market. The corporation, a subsidiary of the European conglomerate Siemens, is based in Plano, Texas, USA. Siemens PLM Software's CAE product range features NX CAE, which comprises engineering analysis tools including the primary calculation modules NX Nastran, NX Thermal, and NX Flow.

2.2.3 Main development trends in CAE systems

In the process of CAE development, developers strive to increase their capabilities and expand the scope of implementation. The main areas of CAE development are:

- improving the methods for solving interdisciplinary modeling problems;
- developing new platforms for integrating various CAE systems, as well as for integrating CAE systems into PLM solutions;
- increasing the functional compatibility of CAE and CAD systems;
- improving the methods for constructing computational grids, describing boundary conditions, parallel computing, etc.;
- improving the characteristics of models used to describe the properties of materials;
- optimizing CAE systems for computer platforms with 64-bit and multi-core processors, and thereby improving the conditions for modeling complex structures with a large number of degrees of freedom.

Self-Assessment Questions

- 1 In which industries are SAE systems used most effectively?
- 2 List the leaders of the SAE systems market.
- 3 Name the main development areas used by SAE systems?
- 4 What advantages does the use of SAE systems provide?

Topic 2.3 Computational methods for engineering analysis in CAE systems

Usually, the initial problem of engineering analysis is formulated in partial differential equations together with initial and boundary conditions. According to the mathematical apparatus used, methods for solving problems in partial derivatives are divided into two groups: *analytical* and *numerical*.

2.3.1 Analytical methods

Deriving a solution by the explicit formulation of functional analytical dependencies enables the assessment of the impact of diverse elements on the subject of investigation; yet, this is possible only for a limited category of research objects characterized by the most elementary dependencies in the boundary conditions.

The structural diagram in Figure 2.2 provides a concise overview of the analytical methods.

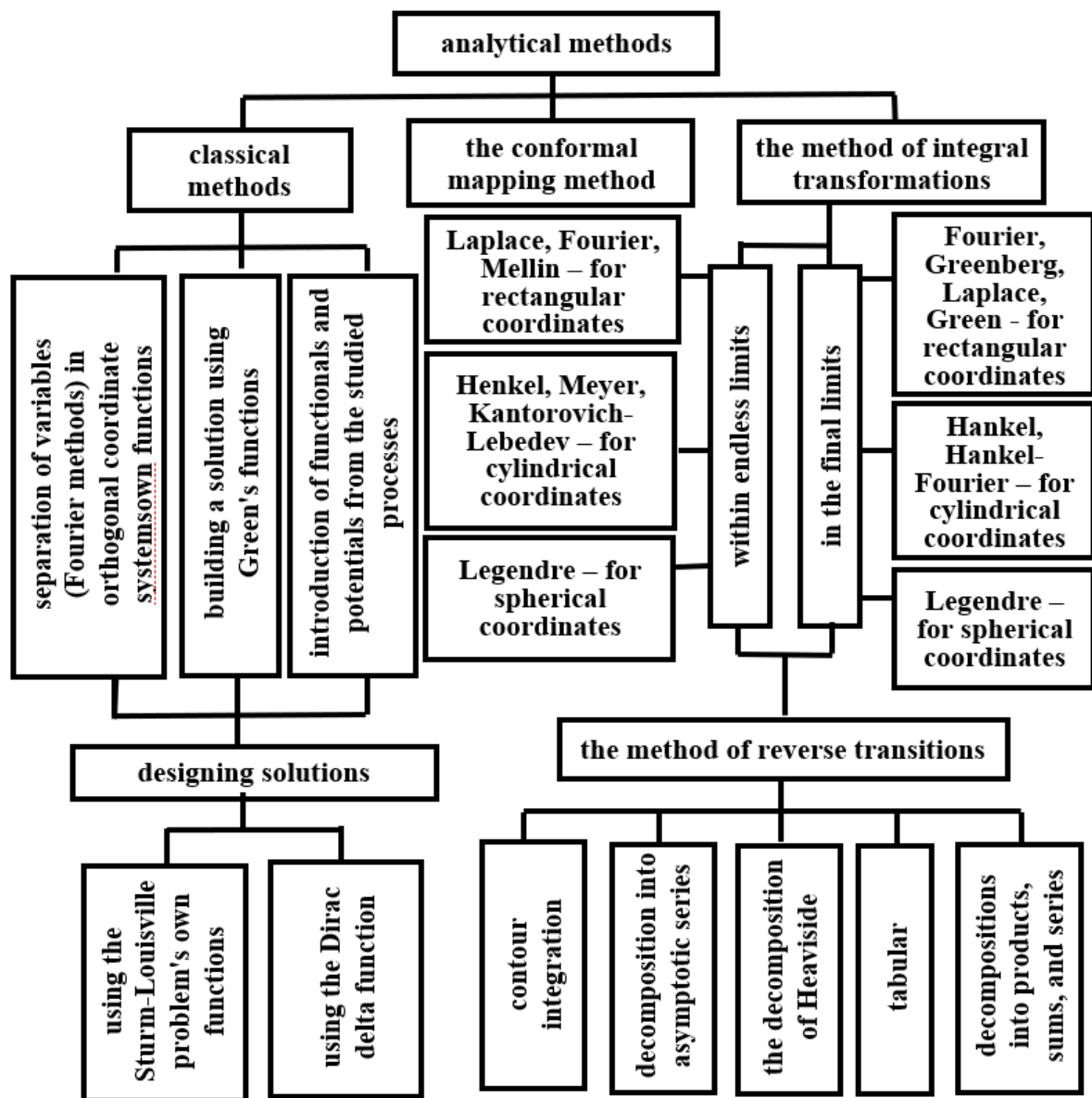


Figure 2.2 – Schematic overview of analytical methods

With the help of various substitutions in analytical methods, techniques are used to reduce the original problem to simpler ones, for

example: solving a partial differential equation to solving ordinary differential equations, solving ordinary differential equations to solving algebraic equations, etc.

The applicability of each method depicted in Figure 2.2 is limited by the domain type, boundary conditions, and the coordinate system employed. The separation of variables method (Fourier method) is applicable to classical problems with homogeneous boundary conditions; Green's functions method eliminates these restrictions, though constructing these functions necessitates ingenuity and can be challenging; conformal mappings are utilized for any domain shape, but exclusively for stationary problems; integral transformation methods demand considerable effort for the inverse transition from images to originals.

All of the above analytical methods are used to to varying extents in practice to achieve ultimate estimates and initial development of design solutions.

It is important to notice that methods for solving direct nonlinear problems are poorly developed. The most common method is linearization, which entails simplifying a nonlinear problem into a linear format whenever possible. This can be achieved through techniques such as integral transformations or by employing a Taylor series expansion of nonlinear dependencies around the mean integral value, retaining only the linear components or specific segments of the expansion.

The greatest difficulties arise when solving nonlinear nonstationary problems in a three-dimensional formulation for bodies of arbitrary shape with variable characteristics in time. We emphasize that the analytical theory for solving these problems has not yet been developed to a sufficient degree.

2.3.2 Numerical methods

For more accurate and complex models, analytical solutions are obtained comparatively seldom. Many important technical problems cannot be solved analytically due to the complexity of the geometry of the structure and the boundary conditions. Currently, numerical methods serve as the primary tool for addressing issues related to complex geometry.

Numerical methods are methods of approximate solution of mathematical problems, which are reduced to performing a finite number of arithmetic operations on numbers, usually performed approximately. The output is presented as a numerical value, accompanied by an error margin that is considered acceptable for the specific problem at hand. Distinction from analytical approaches:

- numerical methods allow obtaining only an approximate solution to the problem;

– numerical methods yield a solution to a problem based on specific parameter values and initial data (i.e., rather than a general formula for an analytical solution, one receives specific values of the sought quantities for a particular set of initial data).

To analyze the dependence of the solution on the parameters and initial conditions, it is necessary to carry out a whole series of calculations. In numerical methods for solving partial differential equations (as opposed to analytical methods), the values of the dependent variable (temperature, displacement, stress) at a certain finite number of points in the initial region (at the nodes of the computational grid) are used as unknowns. Discretization of differential equations is performed. As a result of special transformations, the system of partial differential equations is replaced by a system of linear algebraic equations for unknown values at the nodes of the grid.

A schematic overview of the most developed numerical methods used in engineering calculations is presented in Figure 2.3.

The *finite difference method* (FDM) is based on replacing derivatives in differential equations and boundary conditions with approximate values expressed through differences in function values at a finite set of individual discrete points (nodes) of the region under study, for example, using the simplest linear approximation relationships between adjacent nodes and time instants. In the general case, finite difference approximation also uses the leading terms of the Taylor series when expanding the sought quantities at the nodes of the grid region, or introduces the required number of real parameters (weights). It should be noted that simpler and more visual explicit schemes require certain relationships between the step sizes in time and space to ensure convergence and stability of the solution, while more complex implicit schemes are absolutely stable.

Despite the relative simplicity, versatility and possibility of obtaining a priori estimates of accuracy, the practical use of the FDM for problems with discontinuous boundary conditions and complex domain boundaries is difficult.

The *Monte Carlo method* is a numerical method for solving various problems by modeling random events, based on obtaining a large number of realizations of random variables, which are formed in such a way that their probability characteristics coincide with similar values of the problem being solved. The main idea of the method is to use a sample of random numbers to obtain the desired estimates. Instead of describing the process using analytical apparatus (differential or algebraic equations), a “draw” of a random phenomenon is performed using a specially organized procedure that yields a random result. In essence, any probability problem can be solved by the Monte Carlo method, but it becomes justified only when the “draw” procedure is simpler, and not more complex than analytical calculations.

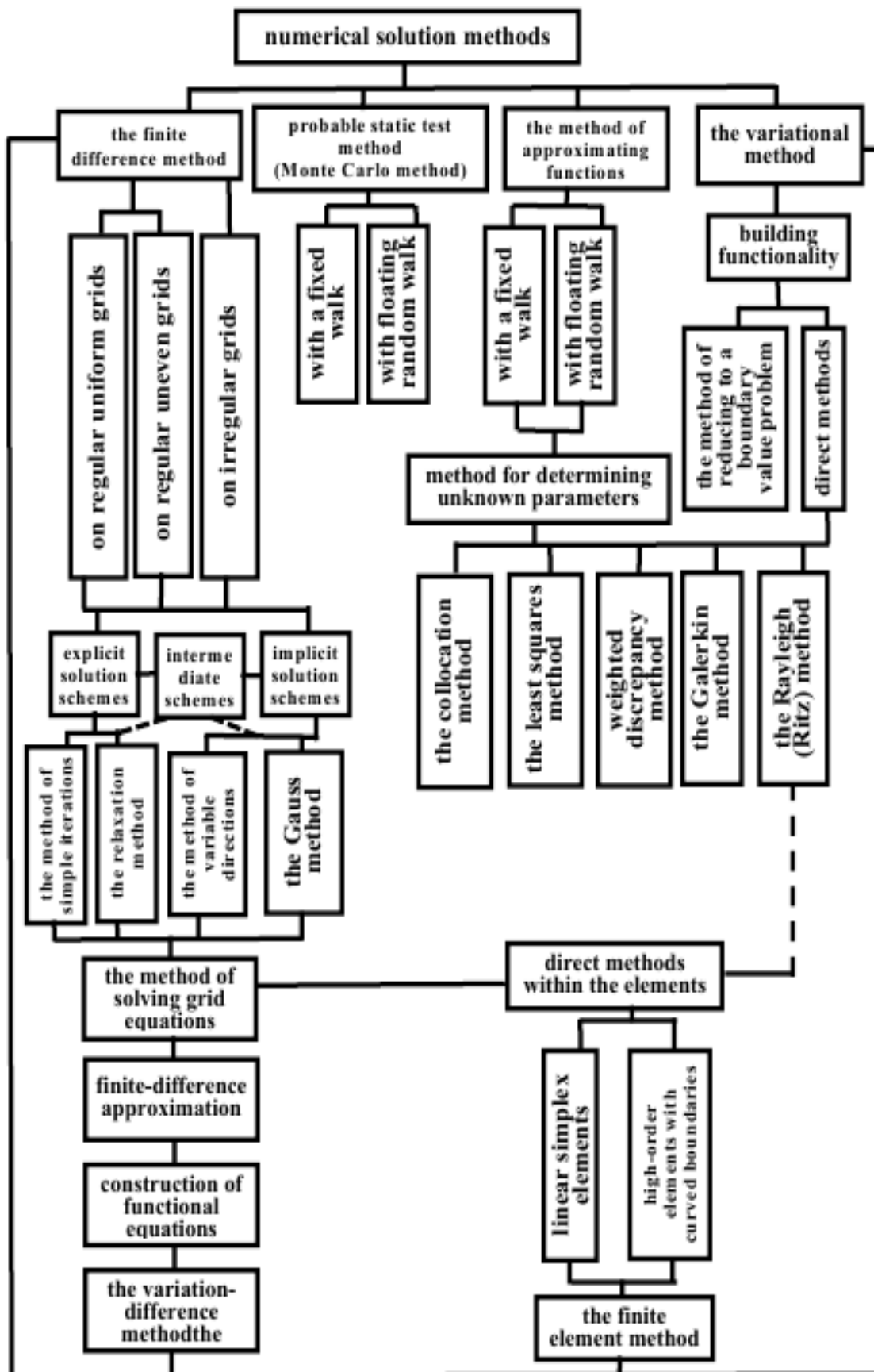


Figure 2.3 – Schematic overview of numerical methods

Methods of approximating functions allow one to search for an approximate solution to a boundary value problem in analytical form, for example:

$$x \approx x_0 + \sum_{i=1}^n A_i \cdot x_i.$$

The approximating system of coordinate functions is constructed from the condition of exact satisfaction of either the boundary conditions or the differential equation, and the incoming unknown parameters are determined by common methods of collocations (coincidence), mean square approximation, and the Galerkin method.

Variational methods require the substitution of the differential boundary value problem with a corresponding variational problem, leading to the development of specialized techniques (such as energy and least squares) for constructing the functional. This enables the search for approximate solutions in analytical form, similar to the method of approximating functions, utilizing well-established direct methods such as the Rayleigh-Ritz method, the weighted residual method, and the Galerkin method.

When applying the last two methods in the case of complex boundary conditions, significant mathematical difficulties arise associated with the choice of a system of coordinate functions.

2.3.3 Finite element method

In CAE systems, modeling programs that employ the finite element method (FEM) hold a pivotal role. The finite element method emerged in the 1950s, pioneered by experts in structural mechanics and elasticity theory.

In the 1960s, a relatively simple method for using FEM for strength analysis through potential energy minimization was introduced, promoting its extensive application in several technological domains. Simultaneously, the first software packages were developed that implemented the finite element method. The emergence of these universal software systems, attributable to the characteristics of the finite element method, was preceded by the development of high-performance electronic computers.

In the 1970s, the finite element method evolved from a numerical procedure for solving problems in structural mechanics into a general method for numerically solving differential equations. In the 1980s, graphic pre- and post-processors and solvers for nonlinear problems were developed.

In the 1990s, FEM tools were integrated into CAD software.

FEM combines finite-difference and variational methods into a single variational-difference method based on the approximation of a continuous

function (for example, displacements, stresses, temperatures, etc.) by a discrete model, which is built on a set of piecewise continuous functions (finite elements).

The basic idea of the finite element method is that any continuous quantity, such as temperature, stress, or displacement, can be approximated by a discrete model that is constructed on a set of piecewise continuous functions defined on a finite number of subdomains. Piecewise continuous functions are defined using the values of the continuous quantity at a finite number of points in the region under consideration. In a general way, the continuous quantity is not known in advance and it is necessary to determine the values of this quantity at some interior points of the region.

To construct a discrete model of a continuous quantity in the implementation of FEM, follow these steps:

- 1 A finite number of points are fixed in the region under consideration. These points are called nodal points, or simply nodes.

- 2 The value of a continuous quantity at each nodal point is considered a variable that must be determined.

- 3 The domain of definition of a continuous quantity is divided into a finite number of subdomains called elements. These elements have common nodal points and together approximate the shape of the region.

- 4 A continuous quantity is approximated at each element by a polynomial that is determined using the nodal values of this quantity. A separate polynomial is determined for each element, but the polynomials are selected in such a way that the continuity of the quantity is preserved along the boundaries of the element (this polynomial is called the element function).

Note that mathematicians and practitioners have differing interpretations of the essence of the method. While the former prioritize substituting the functions of one class with the aggregate of functions from another, the latter concentrate primarily on partitioning the domain into subdomains that define the basis functions. These approaches are not in conflict; rather, the outcome is a mere coincidence in the naming of the subdomains and the basis functions linked to them – both are referred to as finite elements.

In this case, the process of finite element analysis includes a certain number of successive steps based on two approaches: mathematical and physical (Figure 2.4), which are not mutually exclusive, but complement each other.

The values of functions at the nodes of elements at an early stage of the development of FEM were determined by minimizing the functional associated with the differential equation under consideration. Later, more general theoretical justifications made it possible to eliminate the need for a variational formulation of physical problems and to use weighted residual methods to derive systems of equations that determine nodal values.

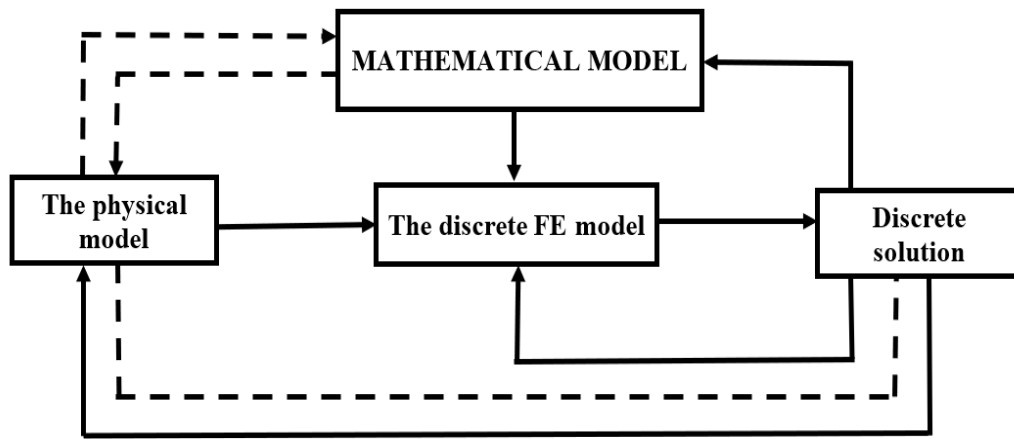


Figure 2.4 – Finite element analysis scheme

General scheme of the calculation algorithm using FEM.

The sequence of procedures in the FEM algorithm can be presented as follows.

1 Discretization of the area under consideration.

Discretization of the problem definition area is the replacement of the continuous medium by a set of finite elements of a given shape, connected to each other at nodes. This stage, despite its apparent simplicity, is of great importance, although it is not determined by strict theoretical recommendations and is largely determined intuitively. Usually, when constructing a finite element model, preliminary ideas about the nature of the expected result are used, and in places of high gradients of the sought quantities, the finite element grid is thickened.

2 Selection of the variational principle.

The choice of the variational principle determines the main unknown functions, through which the remaining unknowns are subsequently established. For example, in problems of solid mechanics, the following variational principles are used: the Lagrange principle, according to which displacements vary; the Castigliano principle, according to which stresses vary.

3 Selection of approximating functions. In piecewise continuous approximation, it is assumed that the displacements inside the element can be expressed through the displacements at its nodes. This relationship is described using so-called shape functions, which approximate the actual displacement field inside the element. The accuracy of the solution depends to a large extent on the choice of approximating functions. These functions must satisfy the following criteria:

- the criterion of completeness: when the dimensions of an element tend to zero, the selected form functions must ensure any simple values;

- the criterion of compatibility: the form functions must ensure the continuity of displacements and their derivatives at the boundary between elements.

Upon fulfilment of these criteria, an increase in the number of finite elements modeling the structure leads in the computation outcomes monotonically converging to the exact solution.

4 Implementation of the variational principle.

At this stage, the stiffness matrices of the elements are calculated, and the global matrix of the system of algebraic equations, as well as the nodal force vector, are constructed. The global stiffness matrix can be obtained using several methods:

- by the direct stiffness method;
- using finite difference operators.

5 Accounting for boundary conditions.

The stiffness matrix obtained on the basis of the specified methods is singular. Correction of this matrix considering boundary conditions leads to a non-singular system of linear algebraic equations.

6 Solution of a system of algebraic equations.

The key advantages of the FEM include:

- accounting for temperature-dependent material properties in numerical modeling;
- simplified problem formulation with mixed boundary conditions;
- applicability to multi-material bodies;
- capability to analyze objects of arbitrary geometric shapes.

The main sources of error in finite element method solutions are:

- discretization error;
- rounding error;
- mathematical model error.

Another source of error lies in the initial data, which is commonly derived experimentally. In general, the accuracy of the computed results cannot exceed the accuracy of this input data.

Self-Assessment Questions

- 1 What challenges emerge when applying analytical methods?
- 2 What analytical methods are you familiar with?
- 3 What are the practical applications of analytical methods?
- 4 What are the key differences between numerical and analytical methods?
- 5 Outline the procedural sequence of the FEM algorithm.
- 6 What are the primary sources of error in FEM?
- 7 Enumerate the main advantages of FEM.

3 ANSYS MECHANICAL APDL

Topic 3.1 Basic information about ANSYS MECHANICAL APDL

3.1.1 Software package organization

The ANSYS environment comprises two main levels:

- entry level;
- modular level.

When launching the software package, the initial level is activated. This interface provides access to common system operations such as renaming the project, clearing the database, and other administrative functions.

At the modular level, multiple specialized modules are available, each providing distinct functionalities and command sets. The complete list of modules, their respective functions, and the corresponding activation commands are detailed in Table 3.1.

Table 3.1 – ANSYS Modules

Module	Function	Call from menu	Command
1	2	3	4
PREP7	Creating a model (building geometry, specifying materials, etc.)	Main Menu =>=> Preprocessor	/PREP7
SOLUTION	Applying loads and obtaining the FEM calculation	Main Menu Solution	/SOLU
POST1	View the results obtained for a specified point in time	Main Menu =>=> General Postproc	/POST1
POST26	Viewing the obtained results for the specified points of the model as a function of time	Main Menu =>=> TimeHist Postpro	/POST26
OPT	Improving the initial model	Main Menu =>=> Design Opt	/OPT
PDS	Determination of the influence of the scatter and distribution of initial data on the results of the final calculation	Main Menu => Prob Design	/PDS

End of the table 3.1

1	2	3	4
AUX2	Converting binary files into human-readable form	Utility Menu => => File => List => => Binary Files Utility Menu => => List Files => => Binary Files	/AUX2
AUX12	Determining emissivity coefficients and creating radiation matrices for thermal calculations	Main Menu => => Radiation Matrix	/AUX12
AUX15	Import of files created by CAD tools or other FEM systems	Utility Menu => => File => Import	/AUX15
RUNSTAT	Preliminary determination of calculation time (CPU), stiffness matrix width and other characteristics for calculation	Main Menu => => Run-Time Stats	/RUNST

Modules are typically launched from the ANSYS Main Menu in the graphical user interface (GUI). For example, selecting Main Menu => Preprocessor activates the PREP7 module. Alternatively, the module can be accessed by entering slash commands in the format /name, where name represents the the target module's identifier.

To return to the initial level from a module, select Main Menu Finish from the on-screen menu or enter the FINISH (or /QUIT) command. It is possible to switch directly between modules without returning to the initial level. To accomplish this, merely indicate the required module or execute the relevant command.

To completely close the ANSYS environment and return to the operating system, select Exit from the Utility Menu File drop-down menu and then click the Exit button in the ANSYS dialog box or use the /EXIT command. By default, the ANSYS automatically saves and reload the model, storing it under the Jobname.DB database file.

3.1.2 Software database

The ANSYS MECHANICAL APDL system stores all initial data (model dimensions, material properties, load data, etc.) and calculated results (displacements, stresses, temperatures, etc.) in a single, specially organized database. The primary benefit of a single database is the capacity to swiftly and effortlessly access, modify, and eliminate any designated data in both textual and graphical formats.

Irrespective of the module employed, operations are conducted using the same database. This facilitates direct access to the model within the database and its loads throughout the environment. The phrase “direct access” refers to the capability to choose, view a list, and exhibit objects.

The database, containing all original data, should be routinely saved to a file. Select Utility Menu => File => Save as Jobname.DB from the drop-down menu or use the **SAVE** command. In either case, the database will be recorded under the Job-name.DB file. The **SAVE** command offers the following options:

- save only model data;
- save model and calculation data;
- save model, calculation and preprocessor data.

To provide an alternative file name, navigate to Utility Menu => File => Select “Save As” from the drop-down menu or use the necessary parameters of the **SAVE** command. Any database save operation initially writes the contents of the backup database file (if it exists) to the Job-name.DB file. If the Jobname.DB file is already present, it will be overwritten.

To restore data from a database file, navigate to Utility Menu => File => Resume Job-name.DB in the drop-down menu or execute the **RESUME** command. This will lead to the reading of the Jobname.DB file. To specify a file with an alternative name, navigate to Utility Menu => File => Resume in the drop-down menu or use the relevant arguments of the **RESUME** command.

Database operations (saving or reading) can be performed at any level in ANSYS, including the base interface. Reading a database file overwrites all current in-memory data with the contents of the file. Combining these operations is particularly useful for testing commands or verifying model behavior, as it allows rapid resetting to a known state.

During model construction, certain scenarios may require resetting the database while maintaining the active session. This is achieved by either selecting Utility Menu > File > Clear & Start New or entering the **/CLEAR** command. Both methods fully initialize the in-memory database (equivalent to restarting ANSYS) but preserve the current session, eliminating the need to exit and relaunch the software.

3.1.3 Graphical user interface

ANSYS Mechanical APDL (the classic ANSYS environment) offers two operating modes: **interactive** and **command**. The interactive mode

provides real-time visualization of each operation's results through the graphical interface, while the command mode processes command files or command sequences in the background without visual feedback.

The easiest way to work with the ANSYS software is to use the software's interactive menu system, called the **Graphical User Interface (GUI)**.

The program is launched via the command sequence Start => Programs => ANSYS => ANSYS Product Launcher. The **Product Launcher** window appears (Figure 3.1). In this window, you must select the working directory, job name, etc.

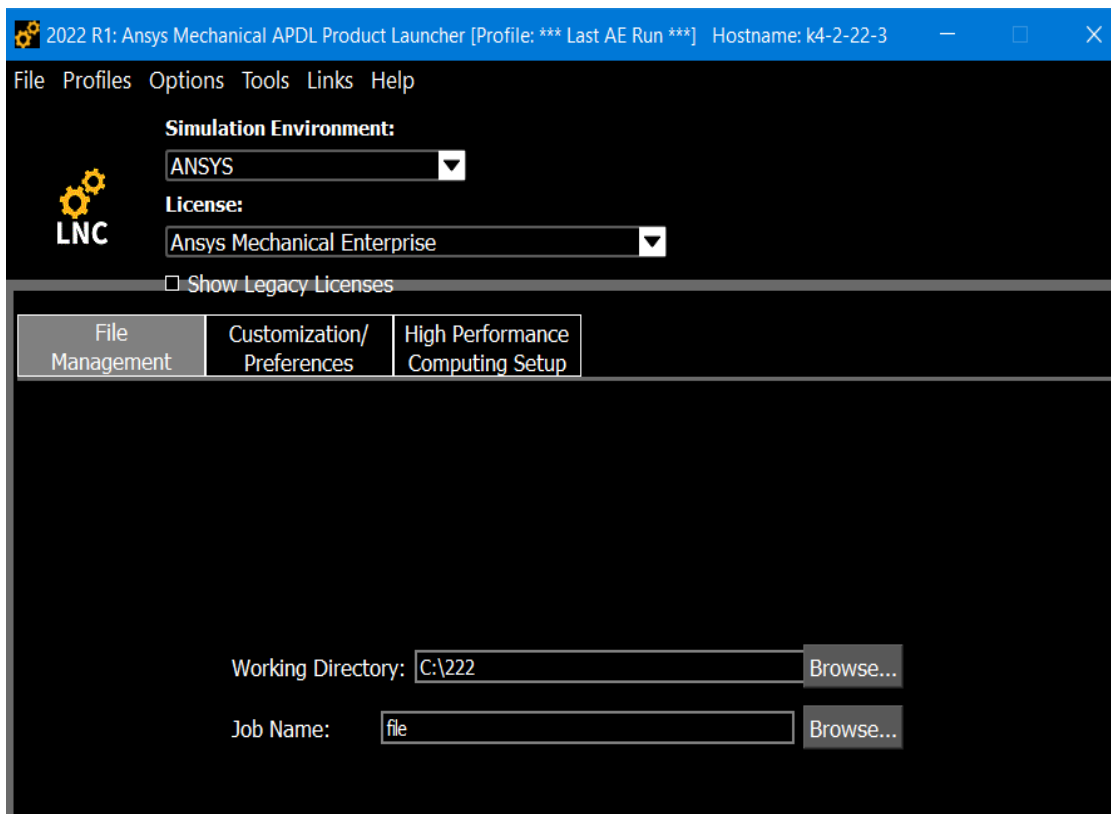


Figure 3.1 – Product Launcher window

For the ANSYS complex, the ANSYS GUI tools are divided into several areas (windows), as shown in Figure 3.2. These include:

1 **Utility Menu** contains service functions used during the entire session of work with the ANSYS complex, such as file control, image control and parameters. This menu is also used to exit the ANSYS complex.

2 **Icon Toolbar Menu** contains graphic buttons that call frequently used ANSYS complex commands.

3 **Input Window** allows direct command input. All previously called commands are contained in a drop-down list for easier viewing and access to them.

4 **Toolbar** is a changeable (supplemented) toolbar containing frequently used ANSYS complex commands and functions. It is possible to add custom buttons by creating abbreviations.

5 **Main Menu** contains the primary functions of the ANSYS complex, divided into modules (preprocessor, calculation, general postprocessor, optimization module, etc.).

6 **Graphics Window** is a graphic representation window.

7 **Status and Prompt Area** is located at the bottom of the GUI tools, it is used to view the hint and view the task status. As a rule, it contains hints on functions that use graphical indication. When reading the hint, you can check that the underlying objects are specified in the established order. This area also contains status information (PREP7, SOLU, etc.).

8 **Output Window** is the window of output messages of the complex. It is usually located behind the other GUI tools, but can be moved to the foreground if necessary.

The above mentioned ANSYS GUI tools are shown in Figure 3.2.

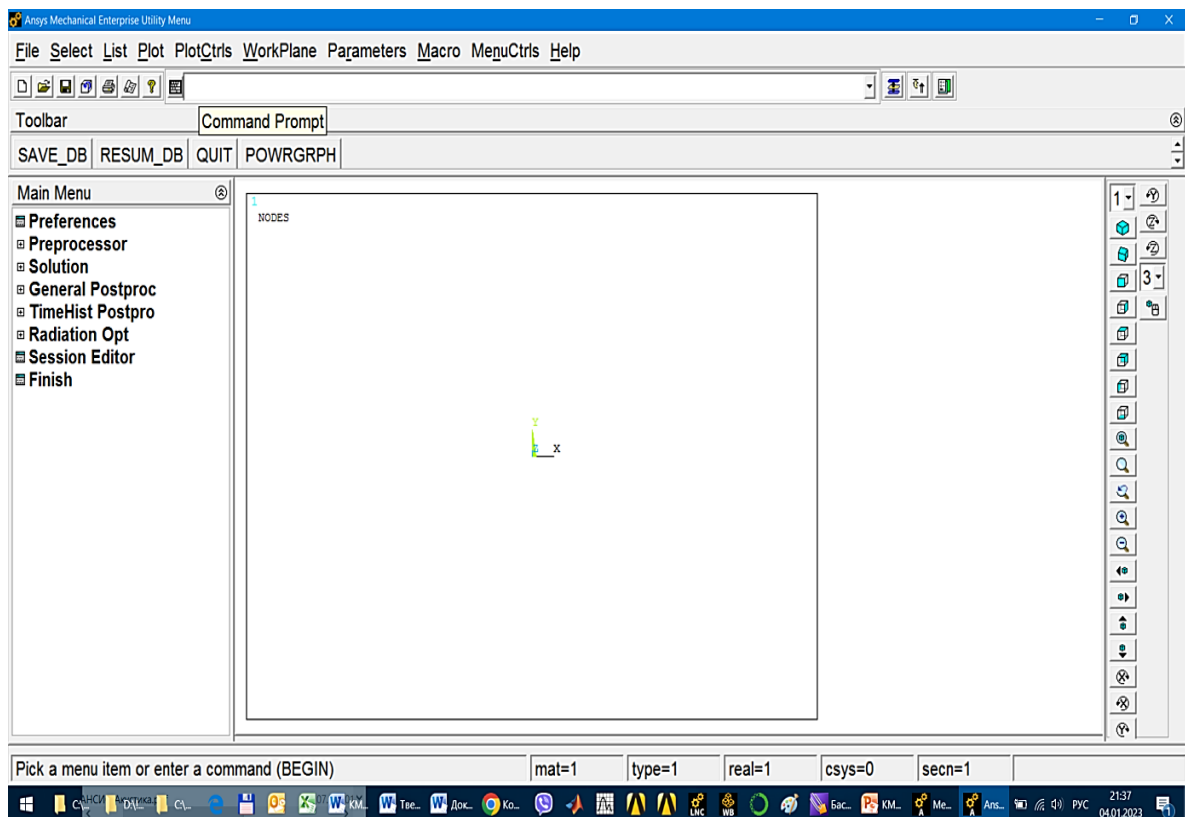


Figure 3.2 – Graphical user interface

The **Utility Menu** drop-down menu contains ANSYS utility functions such as file control, image control, and parameters. Most of these functions can be called at any time during an ANSYS session.

Each section of the **Utility Menu** calls a drop-down menu item that is either a cascading menu or a direct action. Actions can be as follows:

- immediate function call;
- call a dialog box;
- call a cascading menu.

The **Utility Menu** drop-down menu comprises several sections:

- **File** includes functions for working with files and databases, such as clearing a database, saving to a file, and reading from a file. Some of the functions in the **File** section are used only at the initial level. If such a function is called at a level other than the initial level, a dialog box appears, which contains requests to go to the initial level and continue executing the function or canceling the function;

- **Select** contains functions for using the active set of objects and creating components;

- **List** allows to view lists of any objects available in the ANSYS database. In addition, you can get information about the state of various program registration areas and view the contents of files available on the computer;

- **Plot** allows to graphically view points, lines, surfaces, volumes, nodes, elements and other objects that can be displayed graphically;

- **PlotCtrls** includes functions for controlling the image, style and other characteristics of the graphic display;

- **WorkPlane** allows to call or disable the use of a work plane and perform translation, rotation and other operations with the work plane. In this section of the menu, you can also create, delete and switch to another coordinate system;

- **Parameters** includes functions for creating, editing and deleting scalar parameters and parameter arrays;

- **Macro** allows to call macros and data blocks;

- **MenuCtrls** allows to switch from the standard view of the ANSYS complex to the Mechanical Toolbar. – **Help** calls the ANSYS Help System.

The Standard Toolbar contains a set of icons that call the most frequently used functions.

The standard buttons and their respective functions comprise:

- **New Analysis** saves and clears the current calculation information and start a new calculation;

- **Open ANSYS File** accesses the ANSYS database or source file; the operation is determined by the file type;

- **Save Model** saves the current model in the database file;

- **Pan-Zoom-Rotate** calls the Pan-Zoom-Rotate dialog box;

- **Image Capture** calls the image saving tools;

- **Report Generator** calls the report generation tools;

- **Context Help** displays the ANSYS help table of contents, stored in HTML format.

The **Icon Toolbar Menu** contains graphic buttons that call frequently used commands: new task, open file, save database, manage model image, save graphics, create report, help.

The **Input Window** command line provides direct calling of commands in text form. Calling the **Enter** keyboard keys causes the command to be executed. The command text is entered into the called command buffer. The called command buffer contains all previously entered commands. Access to the called command buffer is provided by clicking on the arrow located to the right of the text input field. The drop-down list contains the sequence of called commands. Left-clicking on any line in the history buffer transfers the command to the text input field, where it can be edited and called again. Double-clicking on any line in the called command buffer automatically calls the given command.

The **Toolbar** consists of buttons that call the most frequently used functions of the ANSYS suite. A number of buttons (for example, **SAVE_DB** and **RESUM_DB**) are predefined. It is possible to create other buttons.

The **Main Menu** screen menu contains the main functions of the ANSYS complex, such as the preprocessor, calculation and postprocessor functions. All screen menu functions are modular, i.e., they correspond to the module relative to each other. This means that to call the next function, it is necessary to complete the execution of the previous one.

The **Main Menu** on-screen menu consists of several sections:

- **Preferences** calls the dialog box that allows changing the controls for contextual filtering of menu sections;

- **Preprocessor** enters the preprocessor (by calling the /PREP7 command) and contains such PREP7 preprocessor functions as model creation, mesh creation, and load application;

- **Solution** enters the SOLUTION calculation module (by calling the /SOLU command) and contains such SOLUTION module functions as specifying the calculation type, applying loads, specifying load step options, and performing the calculation;

- **General Postproc** enters the general postprocessor module (by calling the /POST1 command) and contains such POST1 module functions as viewing results in graphical and text modes;

- **TimeHist Postpro** enters the time-based postprocessor module (by calling the /POST26 command) and calls up viewing of results using the Time History Variable Viewer. Contains such functions of the POST26 module as creating and viewing variables in text and graphic modes;

- **Session Editor** calls the session editor;

– **Finish** causes an exit from the current module and transition to the initial level by calling the FINISH command.

The **Graphics Window** is designed for graphic display and execution of all instructions in graphical mode. It is the largest of all GUI windows.

The **Output Window** contains all text output information of the complex: command responses, notes, warnings, errors and other messages. It is usually located behind the GUI tools, but can be moved to the foreground if necessary.

3.1.4 Graphical indication of objects

Many commands of the complex require specifying various objects as arguments. For example, to plot a line by points, it is necessary to specify the numbers of the latter. The ANSYS program allows selecting objects used as command arguments directly on the display screen by specifying them with the mouse. Special panels, called indication panels, are provided for these purposes. They have a standard appearance.

An example of such a panel is shown in Figure 3.3.

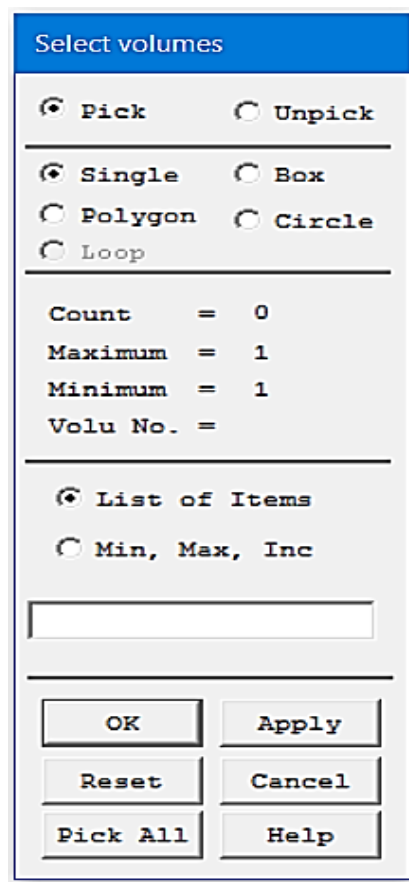


Figure 3.3 – Instruction panel

When the **Pick** button is enabled, objects can be selected here using the left mouse button. Another button, **Unpick**, is used to remove the selection. You can also use the right mouse button to switch between these operations. When selecting objects, the mouse cursor looks like an upward arrow, and when removing the selection, it looks like a downward arrow. The concept of a “hot spot” is used to identify objects when they are selected. For example, if two objects are adjacent, the one with the “hot spot” closest to the mouse cursor is selected. For surfaces, volumes, and finite elements, this point is located near their center. Lines have three “hot spots” – at the ends and in the middle. If the “hot spots” of two or more objects coincide, the “Multiple Entities” panel appears when they are selected. In this case, you must select the desired object using the **Next** or **Prev** button and click **OK**. Activating the **Single** button allows you to select individual objects. Enabling the **Box**, **Polygon** or **Circle** button allows you to select several objects at once, located inside the rectangle, polygon or circle, respectively, constructed by the mouse cursor. The **Loop** button allows you to select a closed sequence of lines or surfaces by specifying only one object from this set. And finally, the **Pick All** button is used to select all objects of the required type.

3.1.5 File management

During operation, the system creates temporary and permanent files. Temporary files are distinguished from permanent files by their automatic deletion upon regular exit from the system. During emergency exit from the system, temporary files are preserved, and certain files must be removed by the user manually.

The permanent files created and used during the operation of the ANSYS complex include the files listed in Table 3.2.

The file name (Jobname) is usually common to all temporary and permanent files and is specified when invoking a session with the complex.

Files are designated with names in the format filename.ext, where filename defaults to jobname, and ext is a distinctive extension comprising two to four characters that delineate the file’s contents. In this instance, jobname refers to the designation assigned during the initiation of the ANSYS session, either through the **/FILENAME** command or by utilizing the Utility Menu File Change Jobname sequence. If the job name is unspecified, the default designation is FILE.

Depending on the application of the files, the system records them in either text (ASCII) or binary format. ERR and LOG files are textual, whereas DB, EMAT, and RST files are binary. Typically, files requiring reading and editing are stored in text format, whereas all others are in binary format.

Let us examine the LOG file independently. While working with the program, whether through the graphical user interface or command inputs, ANSYS records all operations in the LOG file. The LOG file serves as a documented history of interactions with the software. The file possesses a LOG extension and utilizes an ASCII recording format, allowing it to be examined with any external editor. This file denotes an executable program.

Table 3.2 – Files created by ANSYS

File type	Extension
Log-file	Jobname.LOG
Error file	Jobname.ERR
Output file	Jobname.OUT
Database file	Jobname.DB
Backup database file	Jobname.DBB
Results file: – mechanical analysis; – thermal analysis; – magnetic field analysis; – FLOTRAN	Jobname.RST Jobname.RTH Jobname.RMG Jobname.RFL
Step load file	Jobname.SN
Graphic file	Jobname.GRPH
Element matrix file	Jobname.EMAT
Element data file	Jobname.ELEM
File with geometric model data	Jobname.IGES
File with material data	Jobname.MP
File with node coordinates	Jobname.NODE
Output text file	Jobname.OUT

To access the complete contents of the file, execute Utility Menu => List => Files => Log File. Errors that occurred during operation can be rectified using the LOG file. The subsequent activities must be executed:

- generate a LOG file via Utility Menu => File => Generate a DB log file with a .txt extension to facilitate editing;

- open the saved file, make the necessary changes and save. This file represents a program written using commands;
- clear the contents of the ANSYS database using Utility Menu => File => Clear & Start New;
- execute the edited program using Utility Menu => File => Read Input from.

Topic 3.2 Problem Analysis in ANSYS Mechanical APDL

Analysis of any task inside a system consists of the following stages:

- preliminary settings;
- model construction;
- load assignment and analysis execution;
- post-processing of results.

3.2.1 Preliminary settings

Assigning a name to the database file. This operation assigns a new name to the database file and all associated files. If this is not done, the files will have the default name file with the appropriate extension. To assign a name, follow the path in the graphical user interface (GUI) menu and enter your file name: Utility Menu => File => Change Jobname. You can also use the appropriate command in the input line: **/FILENAME,filename.**

Defining the title. To define the title, follow the path in the GUI menu: Utility Menu => File => Change Title. When you perform this operation, the title name will appear in the graphic window. To achieve the same goal, you can use the corresponding command: **/TITLE,filename.**

Defining units of measurement. To switch to the SI system of units, you must specify the **/UNITS** command. This command is not available from the GUI and must be entered directly into the command window: **/UNITS, SI.**

3.2.2 Model construction

Modeling of an object is typically the most labor-intensive phase of problem-solving. Modeling is conducted in the PREP7 preprocessor. At this stage, using mathematical models, a geometric model of the object is constructed, the types of finite elements employed are identified, a finite element model is developed, and the properties of the material and boundary conditions are specified.

Construction of a geometric model. The system has the ability to use the following methods for creating geometric models:

- creation of geometric models by means of the system;
- import of models created by means of CAD.

Below is a brief overview of the methods for creating geometric models using the system.

Building a model from the bottom up. The points that define the corners and edge ends of a model are the lowest-level geometric objects. When building a model, you first create points that are then used to create higher-order geometric objects (i.e., lines, surfaces, and volumes). This process is called bottom-up modeling.

Creating a model from the top down. The complex provides the ability to create a model using geometric objects that completely define the lines, surfaces and volumes being created. Upon the creation of a primitive, the system autonomously generates all associated lower-level objects. If modeling begins with creating primitives, such a process is called creating a geometric model from the top down, or top-down modeling.

The user can combine both methods to create any model.

Employing **Boolean operations.** A geometric model can be constructed with intersection, subtraction, and other Boolean operations. Boolean operations enable direct manipulation of high-level geometric objects to generate complex geometry. Boolean operations can use objects derived from both top-down and bottom-up modeling approaches.

Coordinate systems. The following types of coordinate systems are used in the package:

- global and local coordinate systems used to define the location of geometric objects (nodes, key points, etc.) in space;
- nodal coordinate system defining the direction of the degrees of freedom for each node and the orientation of the components of the result vector at the nodes;
- element coordinate system defining the orientation of the direction-dependent material characteristics and the components of the result vector for the element;
- results coordinate system used to transform the results at nodes or elements into a separate coordinate system for listing printouts, graphical output on the screen, or basic operations in a general-purpose postprocessor.

Let us consider each of the coordinate systems in greater detail.

A **global coordinate system** is a universal or absolute system. The ANSYS program provides for the use of three predefined global systems: Cartesian, cylindrical, and spherical. All three systems are right-handed, have the same origin, and are identified as follows:

- 0 – Cartesian,
- 1 – cylindrical,
- 2 – spherical.

Local coordinate system. In many cases, the user must establish a personal coordinate system, the origin of which does not align with that of the global coordinate system, or whose orientation diverges from that of the specified global system.

The local coordinate system can be Cartesian, cylindrical or spherical. In addition, a toroidal local coordinate system can be defined. Once a local coordinate system is defined, it becomes active. When creating a local system, it must be assigned an identifying number, which must be greater than 10. Local coordinate systems can be created in several ways. For example, defining a local system using the global Cartesian coordinate system: Utility Menu => WorkPlane => Local Coordinate Systems => Create Local CS => AT Specified Loc (Figure 3.4).

Figure 3.4 shows the 'Create Local CS at Specified Location' dialog box. The dialog box contains the following fields and values:

Field	Value
KCN Ref number of new coord sys	11
KCS Type of coordinate system	Cartesian 0
XC,YC,ZC Origin of coord system	0.05, 0.15, 0
THXY Rotation about local Z	
THYZ Rotation about local X	
THZX Rotation about local Y	
PAR1 First parameter	1
PAR2 Second parameter	1

Figure 3.4 – Defining a local system using a global Cartesian coordinate system

In the **Create Local CS at Specified Location** dialog box, you must specify the following parameters.

In the KCN Ref number of new coord sys field, enter a coordinate system number that exceeds 10. If a coordinate system with this number already exists, it will be redefined.

In the KCS Type of coordinate system field, you need to select the type of coordinate system (Cartesian 0, Cylindrical 1, Spherical 2, and Toroidal 3).

In the XC, YC, ZC Origin of coord system field, enter the coordinates of the origin of the coordinate system.

In the fields THXY Rotation about local Z, THYZ Rotation about local X, THZX Rotation about local Y, we input the angular values required to rotate the system concerning the axes of the global Cartesian coordinate system.

Local coordinate systems may be established or removed at any point during program usage. To delete a local coordinate system, proceed as follows: Utility Menu => WorkPlane => Local Coordinate Systems => Remove Local CS.

In the **Delete Local CS** dialog box, in the KCN1 Delete coord system from line, you must specify the number of the local coordinate system to be deleted.

To view the status of all global and local coordinate systems, do the following: Utility Menu => List => Othebnr => Local Coord Sys.

There can be as many coordinate systems as required, but only one can be active at a time. The global Cartesian system is initially active. A newly entered system automatically becomes active. If you need to activate one of the global coordinate systems or one of the preset ones, you must do the following: Utility Menu => Work Plane => Change Active CS to.

While global and local coordinate systems provide the ability to create geometric objects, the **nodal coordinate system** orients the directions of the degrees of freedom at each node. Each node has its own nodal coordinate system, which is parallel to the Cartesian global system by default (regardless of the active coordinate system in which the node was created).

Rotation of the nodal coordinate system, as a result of which the axes of the nodal coordinate system become parallel to the corresponding axes of the active coordinate system, is performed as follows: Main Menu => Preprocessor => Create => Nodes => Rotate Node CS => To Active CS or Main Menu => Pre-processor => Move/Modify => Rotate Node CS => To Active CS.

Rotation of the nodal coordinate system by angles specified by the user is conducted as follows: Main Menu => Preprocessor => Create => Nodes => Rotate Node CS => By Angles or Main Menu => Preprocessor => Move/Modify => Rotate Node CS => By Angles.

Viewing the rotation angles of the nodal coordinate system relative to the axes of the global coordinate system: Utility Menu => List => Nodes.

Each element has its own coordinate system, **the elemental co-ordinate system**, that determines the direction of the axes pertaining to the properties of orthotropic materials, the applied pressures and the results (stresses and strains) for that element.

First, a local coordinate system is created with the required rotation angles relative to the axes of the global coordinate system, and then the number of the created local coordinate system is indicated in the attributes of the elements: Main Menu => Preprocessor => Modeling => Move/Modify => Elements => Modify Attrib – click the **Pick All** button and enter the data (Figure 3.5).

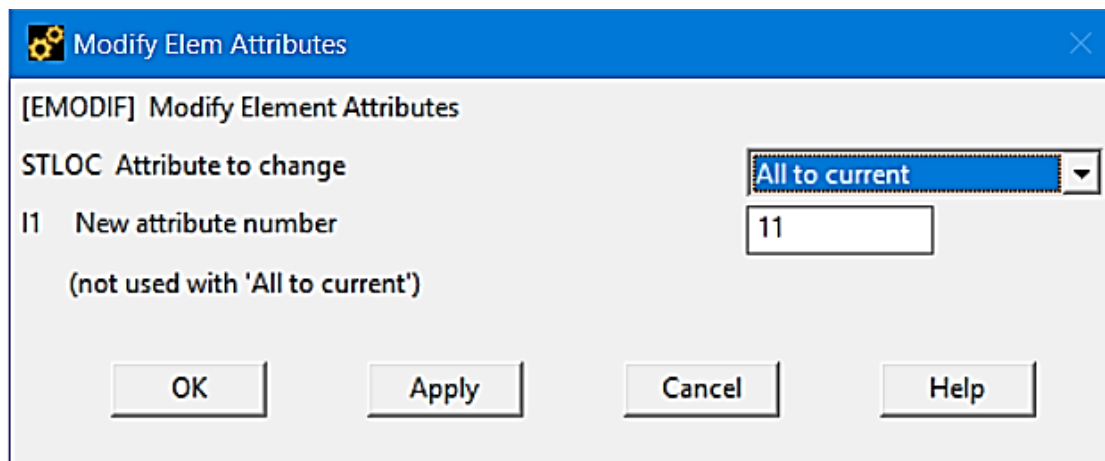


Figure 3.5 – Setting the coordinate system of elements

The output data (results) calculated during the solution are stored in the database and in the results file in the nodal coordinate system and in the element coordinate system. When the results are presented, they are translated into the active results coordinate system (which is the global Cartesian system by default) for display, viewing in text form, and storing in tables.

Changing the results coordinate system: Main Menu => General Postproc => Options for Output. It is possible to change the active results coordinate system to another type (for example, to a global cylindrical system, to a local coordinate system, or to a results coordinate system).

Working plane. Note that although the cursor is a point on the display screen, it actually represents a line in space perpendicular to the plane of the screen. In order to use the cursor to specify the location of points, it is necessary to define an imaginary plane, the intersection point of the cursor line with which will give a single point in space. This imaginary plane is called the Working Plane. In the ANSYS program, it plays an important role. For example, the construction of geometric primitives is always performed relative to the working plane.

By default, the work plane coincides with the XY plane of the global Cartesian coordinate system. However, using the commands concentrated in the Utility Menu => Work-Plane => Align WP with submenu, you can define a new work plane. For example, the Keypoints command allows you to build a work plane using three points that define the origin, the x-axis, and the xy plane.

To move the working plane, use the commands in the submenu: Utility Menu => WorkPlane => Offset WP to.

In this case, the origin of the coordinate system of the work plane can be moved to a specified point, node, or origin of the active coordinate system. In addition, the sequence Utility Menu => WorkPlane => Offset WP by Increments allows you to move and rotate the work plane by specifying three offsets and three rotation angles relative to the axes of its coordinate system.

Construction of a finite element model. To solve the problem as a whole, a computational (finite element) model is required, and geometric information is not used at all in the calculations.

The construction of a FEM model is the process of obtaining nodes and elements that adequately describe the geometry of a body. For these purposes, the complex provides two methods:

- a method based on geometric modeling (solid modeling);
- direct generation.

In the first case, the geometry of the body is first described, and then the geometric objects are automatically divided into finite elements by the program. In this scenario, it is possible to control the size and shape of the elements created in this way.

In the second case, it is necessary to manually determine the position of each node and the connections of each element.

It is obvious that the method based on geometric modeling is more powerful and versatile than direct generation, and is more often used in practice. However, the direct generation method provides more complete control over the geometry and numbering of each node and element. Therefore, there are situations where this method becomes more preferable.

Defining the element type. The ANSYS element library contains more than 100 different types of finite elements. Each element has its own name, which describes the family of elements needed to model the corresponding object, and a number. Table 3.3 lists some of them.

The element type determines:

- degrees of freedom of the element, which also influence the type of analysis (mechanical, thermal, magnetic, electrical);
- model of the object (one-dimensional, two-dimensional or three-dimensional).

To select the required element type, follow the path in the GUI menu: Main Menu => Preprocessor => Element Type => Add/Edit/Delete => Add or use the command in the input line: ET, ITYPE, ENAME.

Table 3.3 – Basic types of ANSYS finite elements

Element Name	Modeling
LINK	Modeling of truss structures, cables, ropes, etc.
SHELL	Modeling of thin-walled structures
PLANE	Modeling of two-dimensional problems (plane problem, plane stress state, axisymmetric problem)
SOLID	Modeling of three-dimensional objects
CONTAC	Modeling of contact conditions

Let us give as an example some types of elements available for thermal analysis (Table 3.4).

Table 3.4 – Element Types for Thermal Analysis in ANSYS

Element	Dimension	Type, characteristic
PLANE35	2D	Triangle, six knots
PLANE55	2D	Quadrilateral, four knots
PLANE77	2D	Quadrilateral, eight knots
SOLID70	3D	Block, eight knots
SOLID87	3D	Tetrahedron, ten knots

Defining Element Options. Each element type typically requires options to be defined. These options allow you to control various aspects of the element.

For example, the SOLID95 element has the following options:

- selection of a local coordinate system associated with the element;
- selection of points at which data (e.g., stresses) are calculated within the element;
- selection of points at which data are calculated on the surface of the element;
- numerical integration rule for construction.

For more information on what options the corresponding element type allows, please refer to the help section for each element.

To select the required element options, follow the path in the GUI menu: Main Menu => Preprocessor => Element Type => Add/Edit/Delete => Options or use the command in the input line: KEYOPT, ITYPE, KNUM, VALUE.

Definition of element constants. Constants are mainly defined for elements that are used to model three-dimensional models of continuous media with lower-dimensional models, for example, in the case of truss, beam and shell elements.

Element constants depend on the element type. For example, the constants for BEAM3, a 2D beam element, are the section area (AREA), moment of inertia (IZZ), section height (HEIGHT), shear constant (SHEARZ), initial strain (ISTRN), and added mass (ADDMAS). Not all elements require constants to be defined.

To set element constants, you need to follow the path in the GUI menu: Main Menu => Preprocessor => Real Constants => Add/Edit/Delete or use the command in the input line: R, NSET, R1, R2, R3, R4, R5, R6.

Generation of finite element meshes. The procedure for constructing nodes and finite elements consists of the following main steps:

- assigning element attributes to objects;
- setting parameters for controlling the finite element mesh;
- generating the mesh of elements.

It is possible to assign different element attributes to different parts of the model. This is done using appropriate reference numbers, including the material number (MAT), the number of the set of “real” constants (REAL), the number of the element type (TYPE), the number of the local coordinate system used as the element coordinate system (ESYS), and the section number (SECNUM). Here, it is possible either to assign attributes to selected geometric entities of the body model (before they are broken down into elements), or to define an active set of attributes (by default MAT = 1, REAL = 1, TYPE = 1, SECNUM = 1) that will be used for all finite elements created in subsequent meshing operations.

To assign attributes directly to geometric objects (points, lines, surfaces and volumes), use one of the following methods:

Main Menu => Preprocessor => Meshing => Mesh Attributes =>

All Keypoints (all points);

Picked KPs (specified points);

All Lines (all lines);

Picked Lines (specified lines);

All Areas (all surfaces);

Picked Areas (specified surfaces);

All Volumes (all volumes);

Picked Volumes (specified volumes).

There are two options available for defining the active set of attributes:

Main Menu => Preprocessor => Meshing => Mesh Attributes => Default Attribs;

Main Menu => Preprocessor => Modeling => Create => Elements => Elem Attributes.

It should be noted that attributes assigned to geometric objects suppress currently active attributes. When generating finite element meshes, if attributes are assigned to an object, these attributes will be automatically assigned to elements created there.

The basic premise of FEM calculations is that as the number of elements (grid density) increases, the solution converges to the exact one. However, this increases the required computer resources. Here, it is necessary to find a balance between accuracy and speed of solving the problem. Thus, in practical use of FEM, the question frequently emerges regarding the requisite number of finite elements to achieve satisfactory accuracy. In the absence of experience, one has to experiment here, starting with a coarse grid and then thickening it and investigating the convergence of the solution.

It is important to remember that FEM is an approximate method, the accuracy of which depends on the correct choice of finite element types and sizes. For example, a finer mesh is required where large stress gradients are expected. At the same time, a coarser mesh can be used in areas with more or less constant stresses, as well as in areas of no particular interest. In this regard, the user must anticipate areas of stress concentration.

It is worth noting that the shape of elements needs to be specified only for those types of elements that allow for the use of more than one shape.

For example, many surface elements can have either a triangular or a quadrangular shape, and volumetric elements can have either a tetrahedron or a hexahedron shape.

The size of the finite elements generated by the program is determined by specifying either the element edge length or the number of line divisions into elements (No. of element divisions).

Defining the global size of elements:

Main Menu => Preprocessor => Meshing => Size Cntrls => ManualSize => Global => Size.

Determining the size of elements near points:

Main Menu => Preprocessor => Meshing => Size Cntrls => ManualSize => Keypoints => All KPs (Picked KPs).

Determining the size of elements on lines:

Main Menu => Preprocessor => Meshing => Size Cntrls => ManualSize => Lines => All Lines (Picked Lines).

The commands listed above can be used together. However, there is a certain hierarchy to resolve conflicts. For example, local settings suppress global ones. The global element size is used only for those objects for which no other splitting parameters are defined.

If the grid control parameters are entirely unspecified, the system determines the dimensions of the elements based on the minimum and maximum number of divisions, the aspect ratio, and primarily, the order of the element.

Automatic division of geometric objects into finite elements is carried out using mesh generation commands (Meshing).

Generation of point elements:

Main Menu => Preprocessor => Meshing => Mesh => Key-points.

Generation of line elements:

Main Menu => Preprocessor => Meshing => Mesh => Lines.

Generation of surface elements:

Main Menu => Preprocessor => Meshing => Mesh => Areas => Mapped => 3 or 4 sided.

Main Menu => Preprocessor => Meshing => Mesh => Areas => Free.

Generating volume elements:

Main Menu => Preprocessor => Meshing => Mesh => Volumes => Mapped => 4 or 6 sided.

Main Menu => Preprocessor => Meshing => Mesh => Volumes => Free.

It is important to state that the ANSYS program employs two distinct types of meshes for surfaces and volumes: free and mapped.

In this case, free grids can be generated on any surfaces or volumes without restrictions. When partitioning surfaces, a free grid can consist of quadrangular and triangular elements (mixed grid) or only triangular elements. However, when partitioning volumes, only tetrahedron-shaped elements are used here (mixed grid is not allowed). Generating regular grids requires that surfaces and volumes be “correct”, i.e., satisfy certain criteria. Thus, surfaces must be three- or four-sided, and volumes must be hexagonal (hexahedron), pentahedral (wedge, prism) or tetrahedral (tetrahedron). The specified number of divisions on the sides must also satisfy certain criteria. In this case, a regular grid on surfaces consists only of quadrangular elements, and in volumes – of hexagonal ones. Note that a regular grid is more economical and more convenient when analyzing calculation results.

However, in many instances, the model's geometry reveals that its constituent surfaces possess more than four sides, while the volumes exhibit more than six faces.

To convert objects to regular ones, it may be necessary to employ one of the following techniques:

- dividing surfaces (volumes) into smaller parts of simpler shape using the Boolean operation of division. In this case, it is convenient to use the working plane as a cutting tool, which is easily set to the desired position;
- using the concatenate operation to reduce the number of sides (faces) by “fictitious” combining (only for the purpose of breaking up) two or more lines (surfaces): Main Menu => Preprocessor => Meshing => Concatenate => Lines (Areas).

For the convenience of the user, all the above commands are combined in one dialog panel “Mesh Tools” (Figure 3.6), which appears when executing the following command sequence: Main Menu => Preprocessor => Meshing => Mesh Tools.

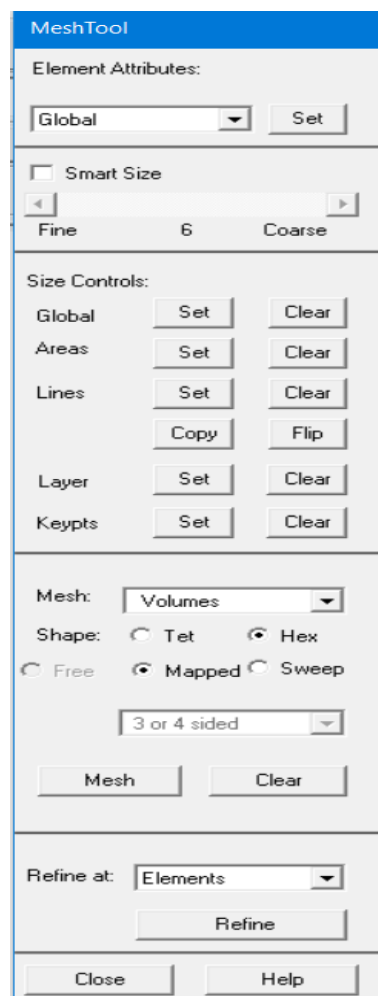


Figure 3.6 – Mesh building tools

Defining material properties. Depending on the task, the following material properties can be specified in ANSYS:

- linear or nonlinear;
- isotropic, orthotropic, and anisotropic;
- temperature-dependent or independent.

To set the material properties, you need to follow the path in the GUI menu: Main Menu => Preprocessor => Material Props or use the command in the input line: MP, Lab, MAT, C0, C1, C2, C3, C4.

The **MP** command allows you to specify a temperature-dependent property as a polynomial. The polynomial can be linear, quadratic, cubic, or fourth-order:

$$\text{Property} = C0 + C1 \cdot T + C2 \cdot T^2 + C3 \cdot T^3 + C4 \cdot T^4$$

If only C0 is specified, the material property will be constant; if C0 and C1 are specified, the material property will depend linearly on temperature, etc.

When a temperature-dependent property is specified in the above manner, the program calculates a polynomial at discrete temperature points with linear interpolation between the points.

The second way to specify temperature-dependent properties is to use a combination of the **MPTEMP** and **MPDATA** commands.

MPTEMP specifies a series of temperatures, and **MPDATA** specifies the corresponding material property values. For example, the following commands define the temperature-dependent thermal conductivity for material 1:

```
MPTEMP,1,10,20,30, , , ,  
MPDATA, KXX, 1,1,1.1,1.2,1.3, , ,
```

3.2.3 Setting loads and performing analysis

Selecting the analysis type and setting the solution specification.

At this stage of solving problems in the complex, the solution block is used. It is necessary to select the analysis type and set its options.

The system enables the execution of various types of analyses:

- Static – static analysis (set by default);
- Modal – modal analysis (i.e., calculation of frequencies and forms of natural oscillations);
- Harmonic – harmonic analysis (i.e., calculation of forced oscillations of the structure under the action of loads changing according to the sinusoidal law);
- Transient – analysis of transient dynamic processes (usually short-term, for example, of an impact nature);

- Spectrum – spectral analysis (i.e., analysis of random oscillations);
- Eigen Buckling – analysis of initial stability (linearized approach);
- Substructuring – analysis of substructures (superelements).

To select the analysis type, follow the path in the GUI menu: Main Menu => Preprocessor => Loads => Analysis Type => New Analysis or use the command in the input line: ANTYPE, Antype, Status.

At this stage, based on the chosen solution and the nature of the problem, the following parameters are established:

- selection of the method for solving the resulting systems of equations;
- setting the solution parameters (load step, number of steps, integration step, number of eigenmodes to be determined, etc.);
- setting the accuracy of the solution;
- setting the parameters for recording the results in a file, etc.

Analysis options allow you to specify the parameters of the calculation being performed. To set the analysis options, you can use the universal dialog panel “Solution Controls”, which is called up by the path in the GUI menu: Main Menu => Solution => Analysis Type => Sol’n Controls (Figure 3.7).

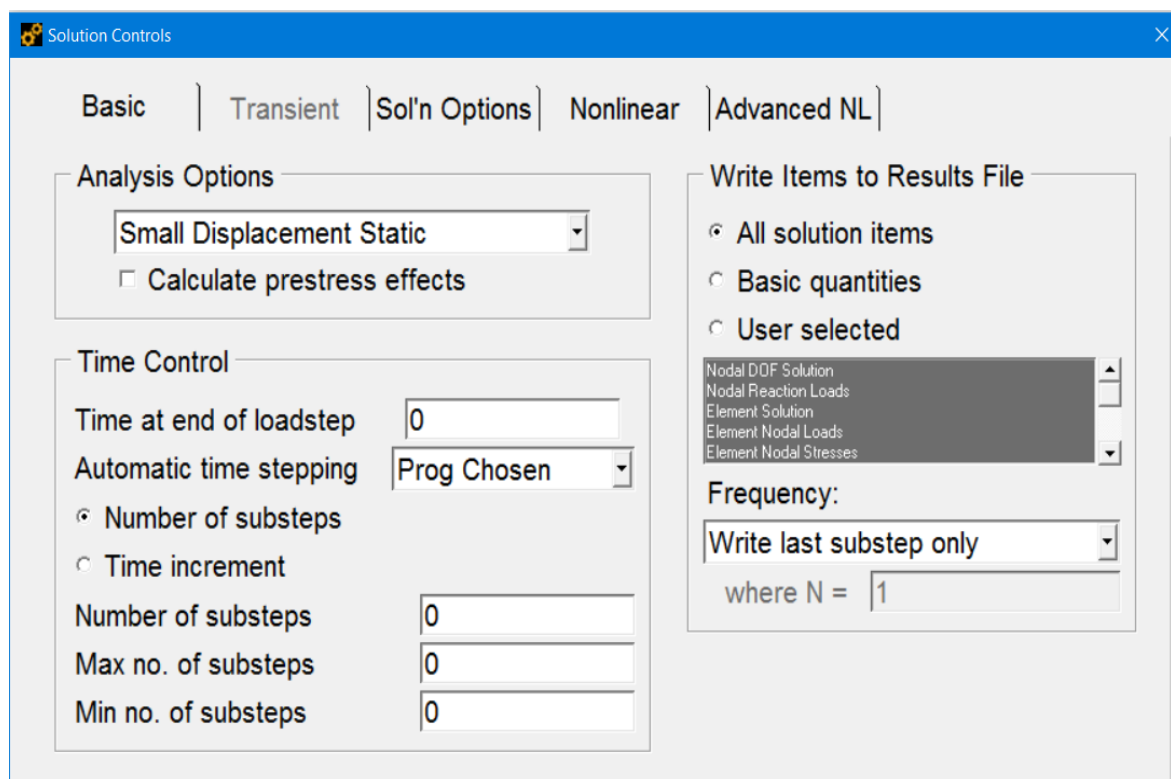


Figure 3.7 – Solution Controls Dialog Box

It should be noted that solving a system of linear algebraic equations can take from fractions of a second to several hours, depending on the size

of the problem and the characteristics of the computer. A linear static calculation requires one such solution, while a nonlinear or dynamic calculation may require tens, hundreds, or even thousands of solutions. Therefore, it is important to choose the solution method correctly.

To solve systems of equations, both direct and iterative methods are used in combination. Each approach has certain advantages and disadvantages, but most often at present, direct methods based on Gaussian elimination are used.

The methods for solving systems of equations available in the complex are divided into three groups:

1) direct: Frontal direct – frontal method, Sparse direct – for sparse matrices, i.e., containing many zeros (selected by default);

2) iterative: PCG – preconditioned conjugate gradient method, ICCG – conjugate gradient method with incomplete Cholesky decomposition, JCG – Jacobi conjugate gradient method;

3) for parallel computing: AMG – method used in multiprocessor computers, DDS – method used in the presence of a distributed computing system.

In most cases, the default Sparse direct method provides good reliability and speed of solution. The Frontal direct method is recommended when reliability is important or when the computer memory is limited. A distinctive feature of the frontal method is that it does not require the construction of a complete stiffness matrix of the structure. When using it, only those equations are formed that are necessary to exclude the next unknowns (degrees of freedom). The iterative PCG method provides high speed of solution for large systems, especially when using volumetric elements. The ICCG method is effective in multiphysics problems, and JCG is recommended for use in cases where only one degree of freedom is determined in each node (as in thermal analysis).

Application of loads. Loads in the complex are divided into six categories:

– DOF Constraints – limitations of degrees of freedom (boundary conditions);

– Forces – concentrated forces and moments;

– Surface Loads – surface loads (e.g., pressure);

– Body Loads – volumetric loads (e.g., temperature);

– Inertial Loads – inertial loads associated with the mass of the structure (arise under the action of: acceleration of gravity, overloads, angular velocity and angular acceleration);

– Coupled-field Loads – loads of coupled fields (used to transfer the calculation results of one analysis as a load for another analysis).

Most of the loads listed above can be applied either to geometric objects (i.e., points, lines, surfaces, and volumes) or directly to the finite

element model (i.e., nodes and elements). The first method seems more convenient, since loads can be defined on geometric objects both before and after breaking the model into finite elements. In addition, loads applied in this way will not disappear when modifying the element mesh. It should be noted that when starting a task on the count, these loads will be automatically transferred to the corresponding nodes and finite elements.

For instance, consider the many categories of thermal loads (Table 3.5).

Table 3.5 – Thermal Load Types in ANSYS

Load type	Units of measurement	Family of commands	Menu path
Temperature	K	D	Main Menu => Solution => Define Loads => Apply => Thermal => Temperature
Heat Flow	J/s	F	Main Menu => Solution => Define Loads => Apply => Thermal => Heat Flow
Convection	W/(K·m ²)	SF	Main Menu => Solution => Define Loads => Apply => Thermal => Convection
Heat Flux	J/(s·m ²)	SF	Main Menu => Solution => Define Loads => Apply => Thermal => Heat Flux
Heat Generat	J/(s·m ³)	BF	Main Menu => Solution => Define Loads => Apply => Thermal => Heat Generat

Loads can be entered in the Solution module. As an example, we will give the path in the GUI menu for setting temperature values:

Main Menu => Solution => Loads => Define Loads => Apply => Structural => Temperature =>

- On Keypoints;
- On Lines;
- On Areas;
- On Nodes;
- On Elements;
- Uniform Temp.

In order to check the applied loads, it may be necessary to re-draw the corresponding symbols. This can be done by activating the necessary options in the “Symbols” dialog box, which appears when you select the menu item Utility Menu => PlotCtrls => Symbol.

To display a list of specified loads, use the menu items Utility Menu => List => Loads.

Launching a task on the calculation. To launch a task on the calculation, use the following GUI path: Main Menu => Solution => Solve => Current LS. In this case, the complex takes information from the database on the finite element model and the current loading step and calculates the solution.

If, after completing the calculation, without leaving the solution block, you change the loading conditions and send the problem for solution again, then a solution for the next step will be obtained, etc. The calculation results are written to the results file (Jobname.rst), as well as to the database. In this case, only one set of results can be stored in the database, while the results for all solution steps can be written to the file.

If you first write the loading steps sequentially into text files (named Jobname.s01, .s02, .s03, etc.) using the menu items Main Menu => Solution => Load Step Opts => Write LS File, then you can get solutions for several loading steps at once: Main Menu => Solution => Solve => From LS Files.

3.2.4 Postprocessing of results

Upon completion of the solution, the results can be retrieved via two post-processors, POST1 and POST26, depending on the nature of the problem.

The POST1 postprocessor (General Postprocessor) is used to review the results in a stationary problem or during a single step of the solution in a non-stationary problem. Path in the GUI: Main Menu => General Postproc or the **/POST1** command.

The POST26 (Time History Postprocessor) is used to review time-dependent results. Path in GUI: Main Menu => Time Hist Postpro or command **/POST26**.

The general postprocessor provides the following types of graphic images of output values:

- Contour Plot – contour image;
- Deformed Shape – image of a deformed state;
- Vector Plot – vector image;
- Path Plot – image of a change in value along a user-defined trajectory;
- Reaction Force Display – image of reaction forces.

Let us consider some of them. Thus, the contour image is used for multi-color representation on the model of the output quantity field (for example, the displacement vector component, stress tensor, etc.). The following commands are available here: Main Menu => General Postproc => Plot Results => Contour Plot =>

- Nodal Solu (nodal solution);
- Element Solu (element solution);
- Elem Table (results from the element table);
- Line Elem Res (results of line elements).

In this case, the Nodal Solu menu item is used to draw continuous contour lines.

The Element Solu line is used to draw contour lines that are discontinuous at the element boundaries. This command is usually used to draw fields of derived quantities.

The next menu item, Elem Table, is used to construct outline images of the quantities stored in the element table.

In practice, when processing calculation results, it is often necessary to construct two-dimensional graphs of various output values. The complex allows you to construct such graphs along a user-defined trajectory. To create a trajectory by directly specifying the nodes lying on it, the following menu item is available: Main Menu => General Postproc => Path Operations => Define Path => By Nodes.

Here, after specifying the nodes, a dialog panel appears (Figure 3.8), where in the Define Path Name field you must specify the name of the trajectory being created, in the Number of data sets field you define the maximum possible number of output values displayed on the trajectory, in the Number of divisions field you specify the number of intermediate segments created between each pair of nodes defining the trajectory (the more intermediate points, the smoother the graph).

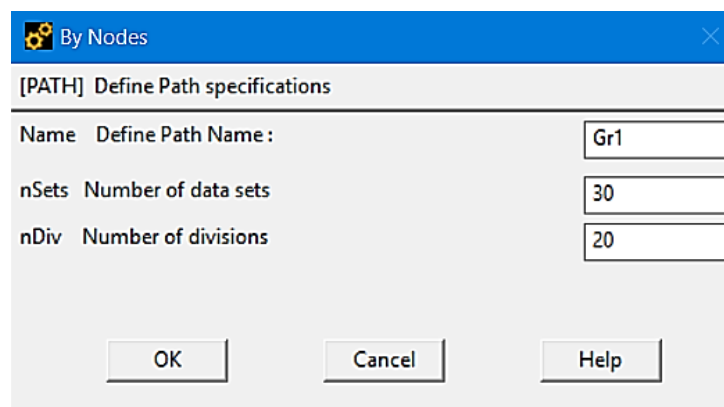


Figure 3.8 – Path Definition Dialog Box

Next, you need to select the output data to be displayed along the created path: Main Menu => General Postproc => Path Operations => Map onto Path.

Here each value must be assigned a name (the reserved names XG, YG, ZG and S cannot be used). Once the value of interest is attached to the curve, the dependence of this variable on the selected path can be obtained in tabular form: Main Menu => General Postproc => Path Operations => Plot Path Item => List Path Items or in the form of a two-dimensional graph: Main Menu => General Postproc => Path Operations => Plot Path Item => On Graph.

A number of mathematical operations (such as integration, differentiation, multiplication, scalar and vector products) can be applied to the value arrays selected in this way.

Presentation of results in the loading history postprocessor. In general, in the loading history postprocessor, all operations are called from the corresponding dialog panel (Figure 3.9), which appears automatically when entering the corresponding section of the program.

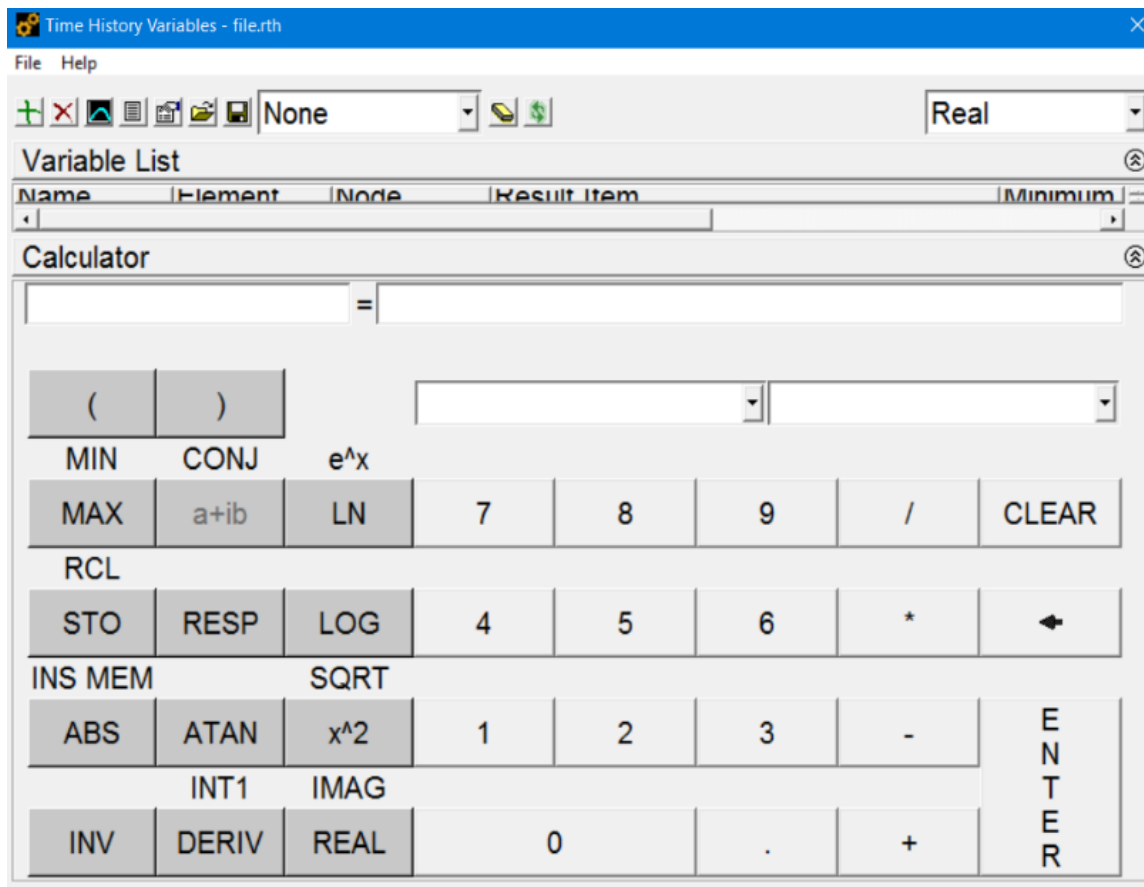


Figure 3.9 – Dialog panel for working with variables in the load history postprocessor

Working in a temporary postprocessor involves defining so-called variables (Variables), which describe the behavior of the model at specific points throughout the entire loading history, and further operations with these variables.

The dialog box for working with variables includes a toolbar, a list of variables, and a calculator.

The toolbar contains the following icons:

- Add Data – creating a new variable;
- Delete Data – deleting a previously created variable;
- Graph Data – displaying a variable on the screen as a graph;
- List Data – displaying a list of variable values in text form;
- Data Properties – specifying variable properties;
- Import Data – importing variable values from an external file;
- Export Data – exporting variable values to an array or external file;
- Overlay Data – replacing data;
- Clear Time-History Data – deleting all variables;
- Refresh Time-History Data – updating the list of variables.

The Variable List contains the actual names of the variables (defined by the user), as well as the numbers of nodes and elements on the basis of which these variables were obtained, the type of variable (displacement, stress, etc.), the minimum and maximum values of the variable, and the flag for using the variable as the abscissa axis when constructing the graph.

The Calculator contains buttons that allow you to perform various operations to create new variables based on existing ones. For example, after defining a variable for displacement in a transient dynamic process analysis, you can calculate the velocity or acceleration by applying the operation of differentiation with respect to time.

To obtain a new variable based on the existing ones, it is necessary to assign a unique name to it in the upper field of the calculator and enter the corresponding expression in the adjacent field (after the equal sign) using the calculator buttons. In this case, the selection of existing variables is carried out from the second list. Entering a new variable is completed by pressing the **ENTER** button.

Topic 3.3 Automated modeling in ANSYS MECHANICAL APDL

When starting to work with the system, users usually deal with the interactive mode, where all tasks are conducted via the menu. This option

allows access to most functions and is sufficient for most tasks. However, some operations and entire calculations are easier to perform using the built-in command language. APDL is an abbreviation for ANSYS Parametric Design Language. APDL is a programming language that allows you to create parametric models and automate routine operations.

Using APDL you can:

- enter dimensions, properties, etc., using parameters rather than numbers;
- obtain information from the model base;
- perform mathematical operations on parameters;
- create macros with a set of commands;
- create parametric models.

Most of the operations that the user performs when working with the program can be replaced by commands. There are two simplest ways to get acquainted with the language.

First, when you click the **Help** button in any dialog box, help for the given operation is displayed, including a description of the corresponding command.

The second method involves recording all instructions executed by the user in a log file. This is a text file that can be viewed in any text editor.

While utilizing the application, the user might acquire knowledge of the commands they frequently employ. Commands may be conveyed to the program through two methods.

The first option is appropriate for inputting individual commands, which are entered via the command line. The alternative is using a batch file. A batch file is a command file executed in a batch mode. This file can be obtained based on a log file. To use a batch file, you need to call the Utility Menu => File => Read Input menu and specify the name of the file that ANSYS should read to obtain input commands and their parameters. In addition to a batch file, it is possible to implement automation of modeling using a mac file. A mac file is a file with sets of APDL commands and macros. When running a macro, a parameter transfer mechanism is used, allowing you to pass arguments inside the macro program. A mac file, like a batch file, can be created in any text editor. To call a macro file, you need to call the Utility Menu => Macro => Execute Macro menu, specify the name of the macro file and its input parameters.

3.3.1 Complex commands

The commands of the complex have a predetermined format. In general, a command consists of the command name, which is the first in the

line, and several fields containing the command arguments. For example, the F command, which applies a concentrated force to a node, has the following form: F, NODE, Lab, VALUE.

To apply a force of 2000 in the X direction to node 376, the F command is: P, 376, FX, 2000.

The command name can be abbreviated to the first four characters. For example, **FINISH**, **FINIS**, and **FINI** all perform the same actions.

Command arguments can be numbers or alphanumeric labels, depending on the purpose of their use. In the F command example shown above, NODE and VALUE are numbers, and the Lab argument is an alphanumeric label.

A number of commands (e.g., **/PREP7**, **/POST1**, **FINISH**, etc.) do not have arguments, resulting in the full command consisting only of the name of the command itself.

The following rules are used to write a program in the system:

- each line must contain one command;
- the maximum number of characters in one line can reach 640;
- when calling a command, its arguments must be placed in the appropriate fields. To skip arguments, use consecutive commas. In this case, the Ansys system will apply default values to the omitted fields, as defined in the command's documentation;
- the values of real numbers must be between $\pm 10^{-60}$ and $\pm 10^{60}$. The absolute value of integers must be between 0 and 99,999,999;
- the Ansys system interprets values provided in the Angle arguments as degrees by default. Note that certain functions within the software also support radians, depending on the command used;
- the following special characters cannot be used in alphanumeric arguments: "!", "@", "#", "\$", "%", "&", "^", "*", "(", ")", "_", "-", "+", "=", "|", "\", "{", "}", "[", "]", ":", ";", "/", "<", ">", "~";
- the Ansys system does not have variable typing, therefore no description of variable types is required. All variables organized by the user are perceived as real;
- the decimal point is used to specify real numbers. For numbers in exponential form, you can use the notation forms with E and D. For example, the number 25000 can be written as 25E3 or 25D3;
- comments on the command line are written using the “!” character.

To minimize the amount of initial information, most commands have default values. There are two types of default values: the default action of the command and the default value of the argument.

The default action of a command specifies the procedure to follow in the event that the command is not called. For example, if the **/FILENAME** command is not called, the default jobname is **FILE**.

The default argument value is the value assumed for an unspecified command argument. For example, if the command **N,10** is called (create a node numbered 10 with X, Y, and Z coordinates that are spaces), the node is created at the origin of the coordinate system; in this case, the X, Y, and Z coordinates are zero by default. Default argument values are usually specified in command descriptions.

3.3.2 Batch File Structure

The structure of the simplest command file will be as follows:

/FILENAME, filename

/TITLE, titlename

/UNITS, units

A block of commands defining variables used in the program.

A block of commands defining functions used in the program.

/PREP7

A block of commands responsible for constructing the model (defining the element type, defining element options, defining element constants, defining material properties, creating a finite element model. Applying loads).

FINISH

/SOLU

A block of commands responsible for the solution (defining the analysis type, specifying the solution, solving the problem).

FINISH

Next comes the postprocessor processing. In this case, depending on the presentation of the results, we use one of two postprocessors.

/POST1

A block of commands of the general postprocessor.

FINISH

/POST26

A block of commands of the temporary postprocessor.

FINISH

3.3.3 Programming elements

Variables and arrays in APDL. APDL contains a set of built-in data types and operations defined on these types. Variables in APDL

terminology are called parameters. The parameter type does not need to be explicitly declared. All numeric values (regardless of their logical meaning – integer or real) are stored as double-precision real numbers.

Any name can be given to a parameter (with some exceptions), but it must start with a letter, contain only letters, numbers, and underscores, and be no more than 8 characters long.

Other rules and guidelines for choosing parameter names are as follows.

Commonly used program identifiers must not be used as names, such as:

- labels of degrees of freedom (TEMP, PRES, etc.);
- labels entered for convenience (ALL, PICK, STAT, etc.);
- labels entered by the user when creating the model (for example, specified by the ETABLE command);
- labels of array types (for example, CHAR, ARRAY, TABLE, etc.).

The names ARG1 through ARG9 and AR10 through AR29 are reserved for local parameters that are used when creating macros. Using these names is not recommended.

Parameter names must not match abbreviations entered with the ***ABBR** command. Parameters beginning with an underscore are used by the graphical user interface and the complex's own macros. It is not recommended to use such names.

The simplest way to specify a parameter is to use the “=” command. The command format is Name = Value, where Name is the name assigned to the parameter, and Value is the numeric or symbolic value of the parameter. The symbolic value is enclosed in single quotation marks. Up to 8 alphabetic characters can be between these quotation marks. Examples of using the “=” command are the following:

ABC=-24

QR=2.07E11

XORY=ABC

CPARM='CASE1'

***SET** command. The following examples illustrate its use:

***SET,ABC,-24**

***SET,QR,2.07E11**

***SET,XORY,ABC**

***SET,CPARM,'CASE1'**

Those parameters whose values are not specified are considered “infinitesimal” and equal to 2^{-100} . For example, if parameter A is specified as A=B, and parameter B is not specified, parameter A is assigned an “infinitesimal” value.

To assign values to parameters contained in the complex database, use the ***GET** command.

The ***GET** command (Utility Menu => Parameters => Get Scalar Data) retrieves the value of an element from the database (node, element, surface, etc.) and stores it as a user-defined parameter.

Parameters:

par is the name of the parameter for storing the result of the property read request. The array must be defined in advance;

entity specifies the type of object properties that will be saved.

The following keywords can be used for this parameter (basic):

- ACTIVE – current active component;
- AREA – surface parameters and statistics;
- CDSY – parameters and statistics for project coordinate systems;
- CE – parameters and statistics for loads defined in the project;
- CP – parameters and statistics for paired elements;
- EDCC – parameters and statistics for contact objects of the model;
- ELEM – geometric, physical, mechanical parameters for the selected element;
- ETYP – number of the property set of the specified element;
- LINE – parameters and statistics of lines;
- MPLAB – data of elements with the specified temperature;
- NODE – parameters and statistics of nodes;
- PART – parameters and statistics of parts;
- SCTN – geometric parameters of sections;
- SECP – statistics of section elements;
- SHEL – shell parameters and statistics;
- TBFT – parameters and statistics for the material property sets used in the project;
- VOLU – parameters and statistics for volumetric primitives;
- entnum – specifies the component number or the number of a set of components (the set is formed using ANSYS selection commands) about which data is required;
- item1 – the name of a specific element;
- it1num – the number (or label) for a specific element (if defined) item1, for some elements item1 does not require it1num;
- item2, it2num – a secondary set of element descriptors (required only for specific commands).

For a complete list of Item1 values, see the description of the ***GET** command.

After setting the parameters (both those specified by the user and those used by the ANSYS program), you can obtain a list of them using the ***STATUS** command.

Parameters can be deleted by leaving the space to the right of the command symbol “=” blank, for example, the command **QR=** deletes the previously entered parameter QR. Specifying a zero value is not equivalent to deleting a scalar parameter.

If there is a need to use the parameters in another ANSYS session, the parameters can be written to a file and then read (restore) this file. To write the parameters to a file, use the **PARSAV** command (Utility Menu => Parameters => Save Parameters).

You can read parameters from a file using the **PARRES** command (Utility Menu => Parameters => Restore Parameters).

In addition to the usual scalar parameters, the complex also uses array parameters.

Arrays can have different dimensions (1D, 2D, 3D, 4D, 5D). The dimensions of arrays can be interpreted as follows:

- 1D arrays – string arrays. A(5) retrieves the fifth element of the array row;
- 2D arrays – arrays of rows and columns. A(2,5) accesses the fifth column in the second row;
- 3D arrays – arrays of planes, rows and columns. A(4,1,8) locates the eighth column of the first row within the fourth plane;
- 4D arrays – arrays of books, planes, rows and columns. A(3,2,1,6) finds the sixth column in the first row of the second plane within the third book;
- 5D arrays – arrays of rows, columns, pages, books and shelves. A(1,4,3,2,8) points to the eighth column of the second row on the third page in the fourth book of the first shelf.

ANSYS provides three types of arrays:

- **ARRAY**. This type is similar to FORTRAN arrays. As with FORTRAN arrays, the indices for rows, columns, and planes are consecutive integers starting with one. The array elements may be either integers or real numbers.
- **CHAR**. This is a character array with each element consisting of an alphanumeric value not exceeding eight characters. The indices for rows, columns, and planes are consecutive integers starting with one.
- **TABLE**. This is a special type of numeric array that allows ANSYS to calculate values between those array elements explicitly defined in the array. In addition, you can define array indices for each row, column, and plane. These indices are real numbers.

To define an array, you must first declare its type and size using the ***DIM** command (Utility Menu => Parameters => Array Parameters => Define/Edit).

The following examples illustrate the use of the ***DIM** command to determine the dimensions of various types of arrays:

***DIM,XYZ,ARRAY,12** ! type ARRAY, dimension 12[x1x1]

***DIM,FORCE,TABLE,5** ! type TABLE, dimension 5[x1x1]

***DIM,CPARR1,CHAR,5** ! type CHAR, dimension 5[x1x1]

To specify arrays, use the “=” command. The “=” command works similarly to the same command for scalar parameters, except that now a column of data is specified (up to ten values for each “=” symbol).

For example, to set the XYZ parameter to size 12x1, you will need two “=” commands:

***XYZ(1)=59.5,42.494,-9.01,-8.98,-8.98,9.01,-30.6,51**

***XYZ(9)=-51.9,14.88,10.8,-10.8**

The ***VEDIT** command (Utility Menu => Parameters => Array Parameters => Define/Edit), available only in interactive mode, launches a data entry dialog box that you can use to edit ARRAY or TABLE arrays.

To read an array from a file, use the command:

***VREAD** (Utility Menu => Parameters => Array Parameters => Read from File).

As with scalar parameters, it is possible to use the ***STATUS** command to obtain a listing of arrays.

To delete an array parameter (numeric or symbolic), simply leave the right side of the “=” command blank. For example, the **XYZ(1)=** command deletes the XYZ array parameter.

Repetition, branching, and looping in APDL. By default, the commands of a complex are executed sequentially, regardless of whether they are entered from the keyboard or from a file. The repetition, branching, and looping facilities of APDL allow you to change the sequential order of command execution:

- repeating the previous command several times;
- branching – exiting a command block and moving to another when certain conditions are met;
- executing cycles – organizing a cyclical repetition of a sequence of commands several times.

These tools can be used anywhere in an ANSYS program, but are especially useful in macros.

The ***REPEAT** command is used to repeat the execution of a command when the value of any field of this command changes by a constant value.

For example, commands

N,1,0,0,0,,,,

***repeat,10,1,1,1,0**

10 nodes are generated (Figure 3.10).

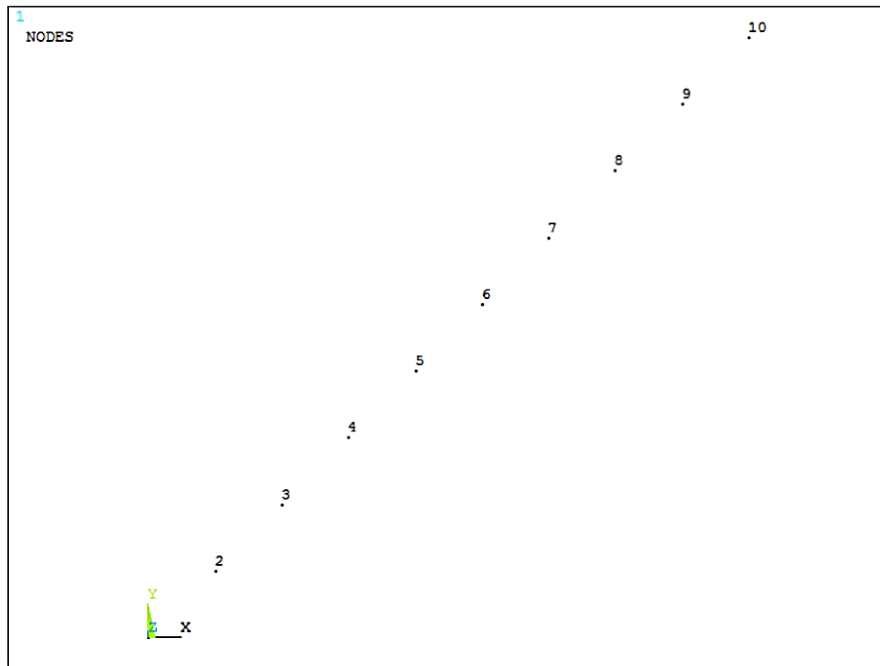


Figure 3.10 – Result of using the ***REPEAT** command

Branching can be implemented using the if-then-else conditional jump construct, using the ***IF**, ***ELSE**, ***ELSEIF**, and ***ENDIF** commands.

The ***IF** command has the following syntax:

***IF**, VAL1, Oper, VAL2, Base,

where

- VAL1 is the first numeric value (or numeric parameter) to compare;
- Oper is the comparison operator;
- VAL2 is the second numeric value (or numeric parameter) to compare;
- Base is the action that occurs if a condition is met.

APDL offers several comparison operators:

- EQ – equal to (for VAL1 = VAL2);
- NE – not equal to (for VAL1 $\neq$ VAL2);
- LT – less than (for VAL1 < VAL2);
- GT – greater than (for VAL1 > VAL2);
- LE – less than or equal to (for VAL1 $\leq$ VAL2);
- GE – greater than or equal to (for VAL1 $\geq$ VAL2).

When you assign the Base argument a THEN value, the ***IF** command becomes the beginning of an if-then-else construct.

The construct consists of an ***IF** command followed by one or more additional ***ELSEIF** commands, an additional ***ELSE** command, and an ***ENDIF** command that marks the end of the construct.

The next example demonstrates a more complex structure, shown in Figure 3.11.

*IF,A,EQ,1,THEN

! Блок 1

*ELSEIF,A,EQ,0

! Блок 2

*ELSEIF,A,EQ,-1

! Блок 3

*ELSE

! Блок 4

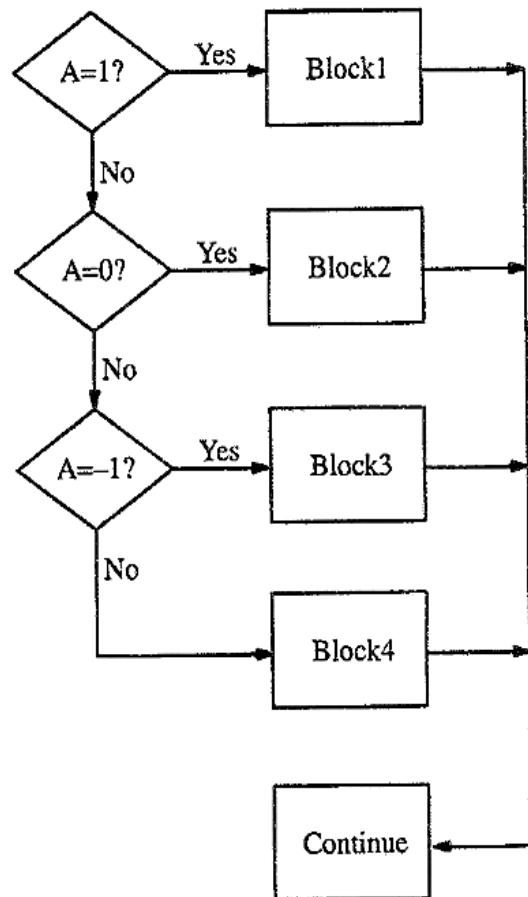


Figure 3.11 – Block diagram

Note that only one block can be executed. If none of the conditions are true, the block after the ***ELSE** command is executed.

The command ***DO,par,ival,fval,inc** organizes a loop and indicates the beginning of the loop body.

Parameters:

- **par** is the name of the scalar parameter that is used as the loop iteration counter. String parameters cannot be used;
- **ival** is the initial value of the counter;
- **fval** is the final value of the counter;
- **inc** is the counter step (default is one).

Negative numbers are allowed for all parameters. Non-integer parameters are allowed.

If **ival** > **fval**, the loop execution is terminated (if this is the case when the loop is first entered, the loop body will not be executed at all).

The ***DO** command can be used on any processor.

The ***ENDDO** command indicates the end of the loop body. This command is paired with the ***DO** command. All commands located between ***DO–*ENDDO** form the loop body. The ***ENDDO** command can be used in any processor.

An example of using the ***DO – *ENDDO** block with the ***IF** command:

C1=1 !initial value of C1

C2=3 !initial value of C3

!loop from 1 to 10 with step 1, start of loop body

***DO**,i,1,3,

C1=C1+1!increment C2 by one

!if C1 > C2 increment C2 by one

***IF**,C1,GT,C2, THEN

C2=C2+1!increment C2 by 1

***ENDIF**

***ENDDO** !end of loop body

4 ANSYS WORKBENCH

Topic 4.1 Basic information about ANSYS WORKBENCH

Currently, the ANSYS system is one of the most powerful software tools for analysis for a wide range of engineering disciplines. The system provides tools for performing the following types of analysis:

- **Structural Mechanics** – modeling of dynamics and strength problems, including linear statistical analysis, spectral and harmonic analysis, buckling assessment, and fracture mechanics. The ANSYS line of modules for solving problems of solid mechanics includes Static Structural, Transient Structural, Rigid Dynamics, Steady-State Thermal, Transient Thermal, Linear Buckling, nCode, Harmonic Response, Random Vibration, Modal, Response Spectrum, Explicit Dynamics/AUTODYN;

- **Fluid Mechanics** – modeling of liquid and gas flows. Allows to study a wide range of problems of fluid dynamics, including stationary and unsteady flows, compressible and incompressible flows, inviscid, laminar and turbulent flows, multicomponent and multiphase flows, flows with chemical reactions, flows through porous media, etc. The line of ANSYS modules for preprocessing, solving and processing results in fluid dynamics problems includes the following modules: Fluent, CFX, Icepak, CFD-Flo, Vista TF, Icem CFD, TurboGrid, TGrid, BladeModeler, CFD-Post;

- **Electromagnetics** – calculation of electronic components and devices, integrated circuits, antennas, electrical machines, drives, power electronics, automation systems, transformers, electric batteries, etc. The line of software products for solving electromechanical problems includes Maxwell, RMxpert, PExpert, Simplorer;

- **Linked calculations** – modeling of problems at the junction of different sections of physics. Allows for multidisciplinary analysis based on a combination of different types of solvers. As a rule, each solver corresponds to a separate application in the program environment.

Access to any solver and cross-application interoperability are efficiently managed through ANSYS Workbench.

Using ANSYS WORKBENCH takes the modeling process to a modern level. Using ANSYS WORKBENCH makes modeling more visual and many user actions easier, which is especially important for novice users.

The following stages of engineering analysis in ANSYS WORKBENCH can be distinguished:

- *model development* (preprocessing). At this stage, the geometric model is prepared, the material and its properties are specified, the finite

element mesh is generated, and the physical conditions of the modeling are determined. The final result of the stage is a model prepared for numerical solution;

- *solver setup and solution*. At this stage, the necessary solver settings are set, parameters that ensure the convergence of the iteration process, and the solver is launched. The final result of the stage is a numerical solution obtained with a given accuracy;

- *processing of results* (postprocessing). At this stage, the obtained numerical solution of the problem is used to visualize the distribution of the necessary physical quantities (stresses, deformations, temperatures, etc.). The final result of the stage is a set of graphs, animations, arrays of values representing the necessary results of the solution to the problem.

In addressing a practical problem, it is typically essential to frequently revisit previous stages, modify the models, reconstruct the finite element mesh, and adjust the solver settings. The availability of efficient tools in ANSYS WORKBENCH is crucial, as it facilitates a substantial acceleration in the solution acquisition process.

ANSYS WORKBENCH provides the engineer with a large number of tools necessary at all stages of analysis, allows for rapid numerical solution by various solvers, and effectively organizes the relationships between individual types of engineering calculations.

It is important to note that ANSYS WORKBENCH does not yet provide the user with all the capabilities of ANSYS Mechanical APDL and automates some important operations, such as selecting the type of finite element, which is not always convenient. ANSYS, Inc. is intensively developing ANSYS WORKBENCH both in terms of improving the graphical interface and in terms of expanding the capabilities of the platform.

Self-Assessment Questions

- 1 List the types of analysis that can be performed in ANSYS WORKBENCH.

- 2 List the stages of performing engineering analysis in ANSYS WORKBENCH.

- 3 Compare the advantages and disadvantages of ANSYS WORKBENCH and ANSYS Mechanical APDL.

Topic 4.2 ANSYS WORKBENCH Graphical Interface

ANSYS WORKBENCH is launched from the main menu Start => Programs => ANSYS => WORKBENCH. After loading, the main program window is displayed, which consists of several windows (Figure 4.1).

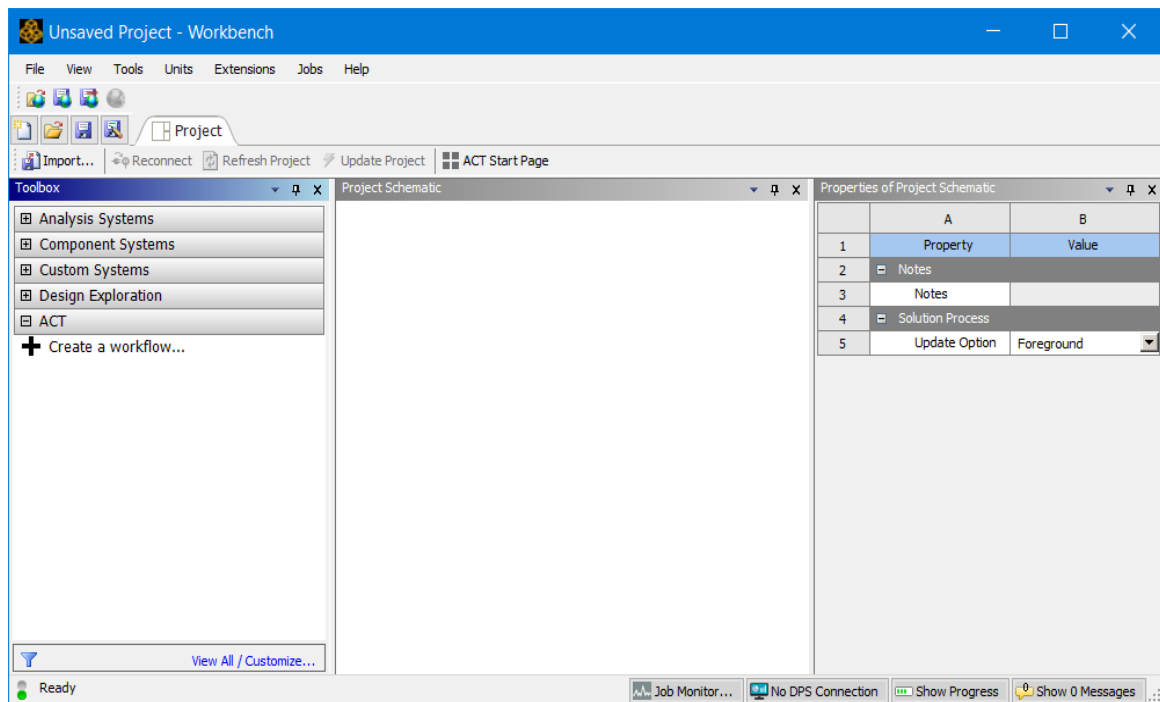


Figure 4.1 – ANSYS WORKBENCH Main Window

Each window can be resized, minimized, or closed using standard Windows tools. Below is a description and functionality of the windows.

The **Project Schematic** window is the main window of the project, located in the center of the main WORKBENCH window (Figure 4.2). It contains the structural components of the project (blocks) and the connections between them. This approach makes it possible to visually represent parts of the project and manage the connections between its individual blocks.

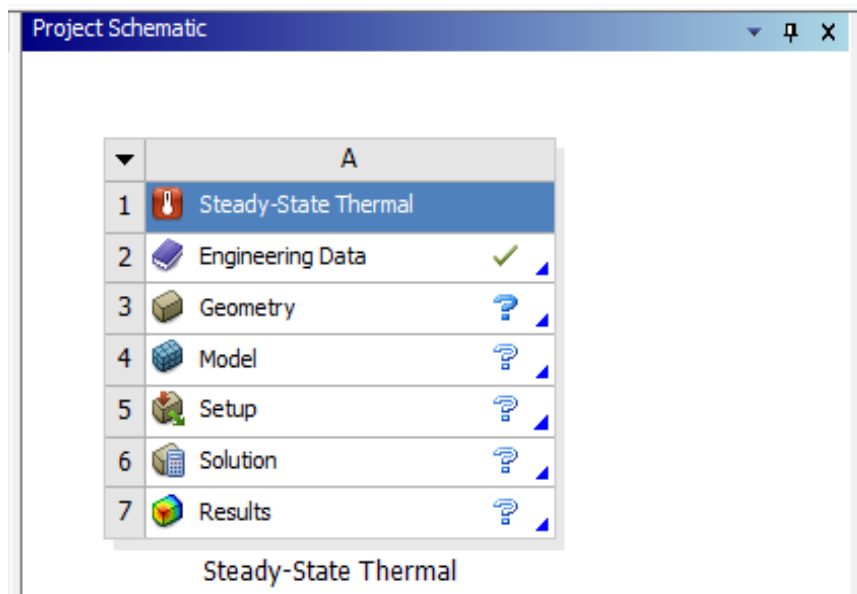


Figure 4.2 – Project Schematic window

The **Toolbox** window is the project tool window, located to the left of the Project Schematic window. It contains several sections displayed as drop-down lists (Figure 4.3).

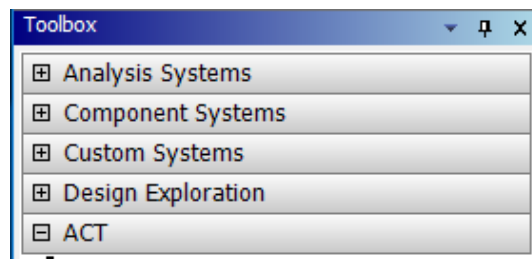


Figure 4.3 – Toolbox window

Sections of the **Toolbox** window:

– **Analysis Systems** contains ready-made templates for various types of numerical analysis (Figure 4.4);

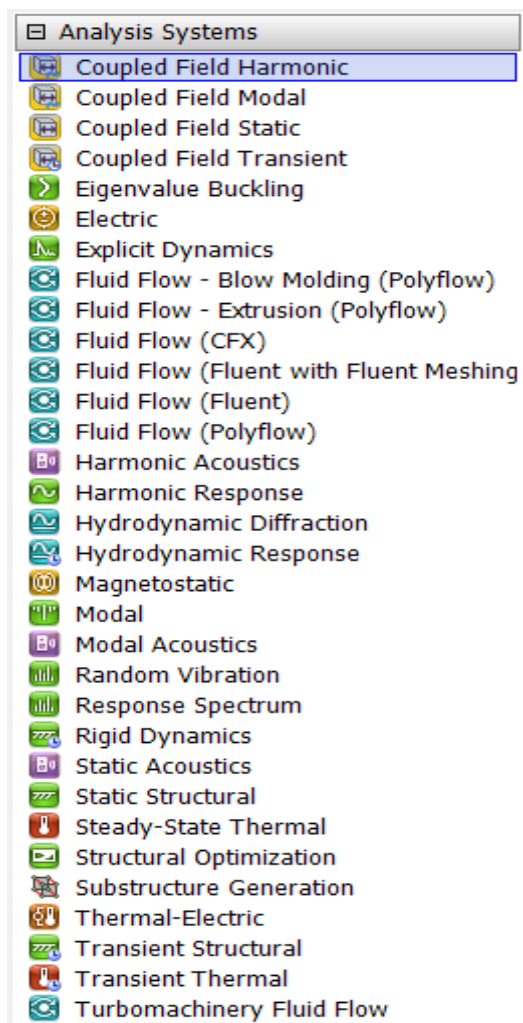


Figure 4.4 – **Analysis Systems** window

– **Component Systems** contains individual engineering analysis components, such as a geometric model, finite element mesh, etc. (Figure 4.5);

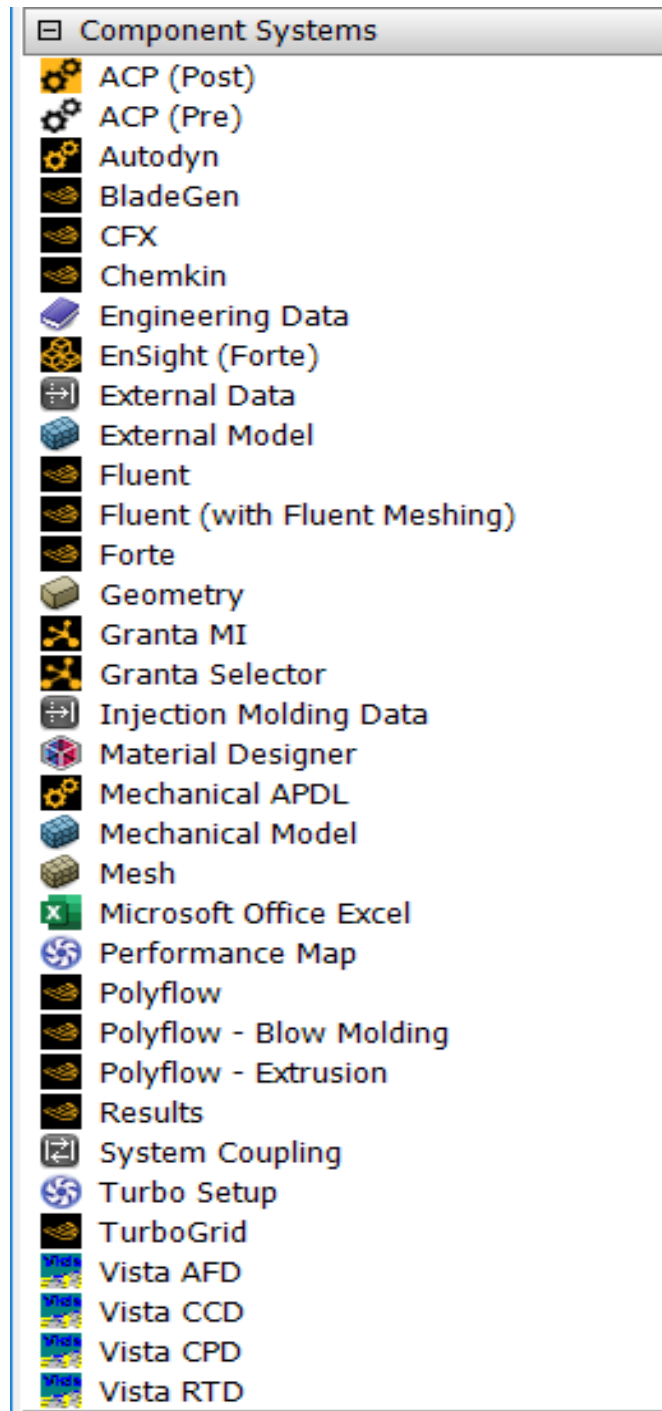


Figure 4.5 – **Component Systems** window

– **Custom Systems** contains ready-made template bundles for solving interdisciplinary problems (Figure 4.6);

– **Design Exploration** contains tools for solving optimization problems (Figure 4.7).

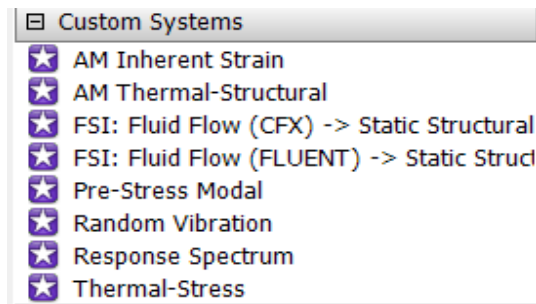


Figure 4.6 – **Custom Systems** Window

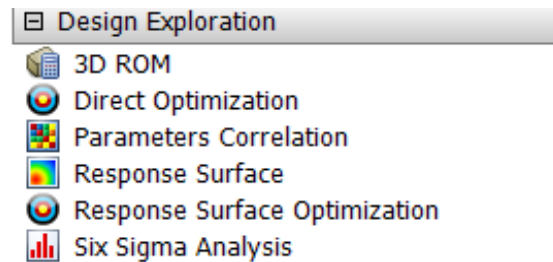


Figure 4.7 – **Design Exploration** Window

The **Properties of Schematic** window is a window that displays the properties of the selected object (Figure 4.8). All the properties of the object in this window are displayed in the form of a table, the first column of which contains the name of the property or parameter, and the second reflects its value.

Properties of Schematic A6: Solution		
	A	B
1	Property	Value
2	General	
3	Component ID	Solution
4	Directory Name	SYS
5	Queue	
6	ServerUrl	
7	UserName	
8	Notes	
9	Notes	
10	Used Licenses	
11	Last Update Used Licenses	
12	System Information	
13	Physics	Thermal
14	Analysis	Steady-State
15	Solver	Mechanical APDL
16	Solution Process	
17	Update Option	Use ap...
18	Solve Process Setting	My Computer

Figure 4.8 – **Properties of Schematic** window

At the bottom of the screen is the **Messages** window, which is used to display service messages, warnings, or error messages.

At the top of the main window is the ANSYS WORKBENCH start window area, which contains the **File, View, Tools, Units, Extension, Jobs, Help** tabs. Let's take a closer look at the **File, View, Tools, Units** tabs.

The **File** tab menu enables the creation of a new project, the loading of an existing project, and the saving of an ongoing project.

The **View** tab menu facilitates the addition or removal supplementary windows in the ANSYS WORKBENCH start-up graphical interface. To do this, you need to check or uncheck the box next to the name of the window you are interested in by left-clicking on the corresponding name. The size and location of all windows can be changed by manipulations similar to those performed when working with Windows windows.

The **Tools** tab menu allows configuring basic settings for ANSYS WORKBENCH.

The **Units** tab menu (Figure 4.9) enables the selection of a measurement system suitable for the current task. One can switch between different systems of units at any point during the project's progression. Select the desired system of units by marking the checkbox adjacent to the relevant system from the available options.

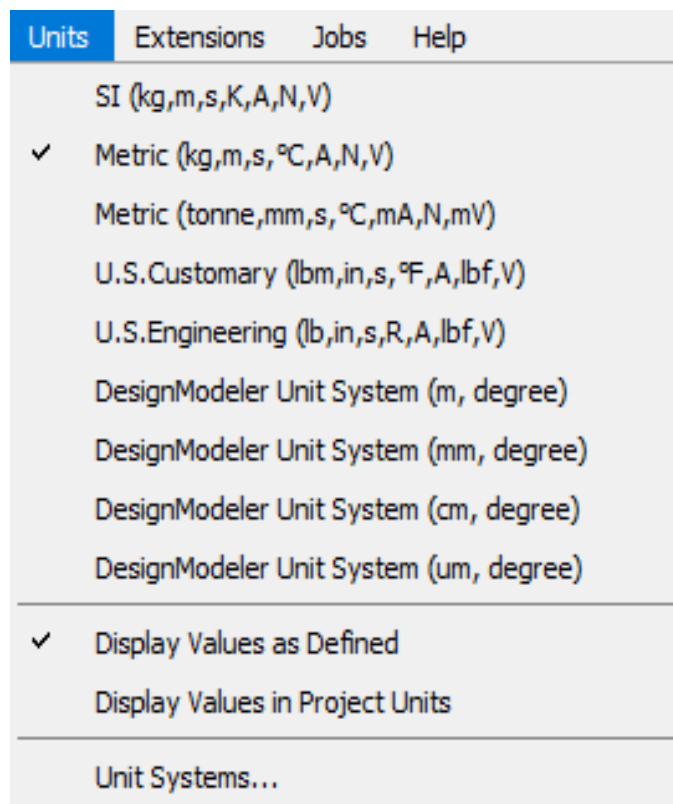


Figure 4.9 – **Units** window

Self-Assessment Questions

- 1 What elements are included in the **Project Schematic**?
- 2 What tools does **the Design Exploration** window contain?
- 3 Which window's capabilities are most effectively used in addressing interdisciplinary issues?

Topic 4.3 Working with a project in ANSYS WORKBENCH

4.3.1 Overview of core modules and applications

Here is a brief overview of the main modules and applications used in ANSYS WORKBENCH to work with a project.

Engineering Data – an interface for managing a database of material properties, as well as input parameters of mathematical models.

Design Modeler and **SpaceClaim** – applications for creating geometric 2D/3D models.

Meshing – a multifunctional mesh preprocessor that allows you to generate high-quality computational meshes in automatic mode for various types of engineering analysis.

Static Structural is designed to solve problems of mechanics of a deformable solid in a static setting.

Transient Structural – a module for solving problems of structural dynamics.

Explicit Dynamics/AUTODYN/LS-DYNA – modules for calculating problems of structural dynamics and modeling fast nonlinear processes: high-speed impacts, penetrations, fragmentation, destruction, etc.

Rigid Dynamics is designed to simulate the dynamics of moving systems and mechanisms.

Steady-State Thermal/Transient Thermal – analysis of a stationary/non-stationary thermal field.

Fluid Flow (CFX)/Fluid Flow (Fluent) are designed to solve problems of hydrodynamics, as well as for problems of conjugate heat transfer.

Electric is designed to simulate electric fields of direct current in conductors.

Thermal-Electric – stationary electrothermal analysis, allowing you to study the processes of heat generation when an electric current passes through a conductor.

Modal – modal analysis, calculation of natural frequencies and vibration modes.

Harmonic Response – harmonic analysis for determining the response of a structure to harmonic loads. Allows you to assess the negative consequences of forced vibrations - resonance, fatigue, etc.

Response Spectrum – analysis of the response of a structure to dynamic loads. Used to calculate the seismic resistance of structures.

Random Vibration – analysis of the response of a structure to random vibration loads.

As an example, we will give the structure of the **Steady-State Thermal** static strength analysis project. Double-click the left mouse button to select Static structural in the Analysis Systems window. A block containing the stages of analysis will appear in the main project window (Figure 4.10):

- setting material properties (**Engineering Data**);
- building and importing geometry (**Geometry**);
- forming a finite element model (**Model**);
- setting boundary conditions and loads (**Setup**);
- performing calculations (**Solution**);
- outputting results (**Results**).

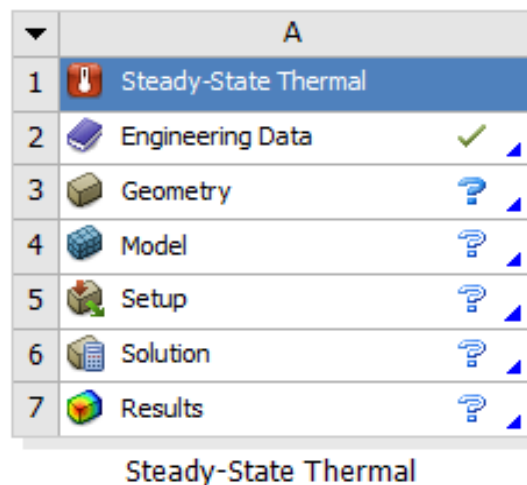


Figure 4.10 – Result of analysis selection

The engineering analysis block control menu in **Project Schematic** is called by pressing the button in the upper left corner of the block and allows you to perform the following actions (Figure 4.11):

- update (**Update**);
- make a duplicate of the block (**Duplicate**);
- change the type of engineering analysis in the block (**Replace With**);
- delete all block information (**Clear Generated Data**);

- delete the block (**Delete**);
- rename the block (**Rename**);
- go to the block properties window (**Properties**).

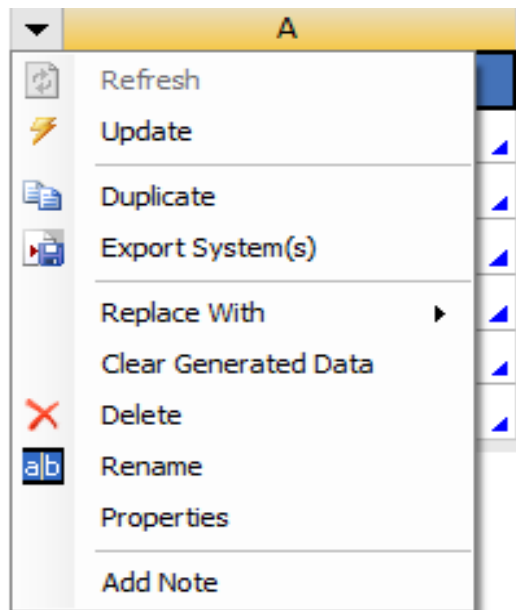


Figure 4.11 – Engineering Analysis Block Control Menu

Each block element has a similar menu. To call it, you need to right-click on the element (Figure 4.12). It displays the operations performed with this element, such as editing the element (**Edit**), updating (**Update**), transferring data from a new block (**Transfer Data From New**), and others.

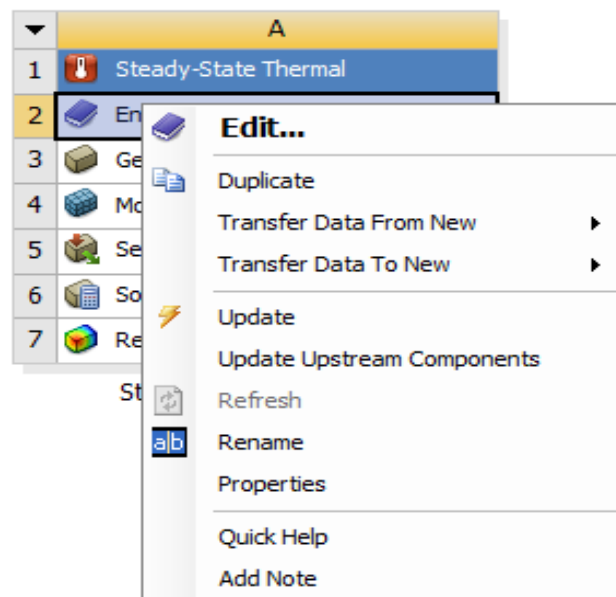


Figure 4.12 – Engineering Analysis Block Elements Control Menu

Each element of the engineering analysis block has a status, which is displayed to the right of its name. If the block has just been inserted, by default only the material properties are specified in the **Engineering Data** element, which is marked with a check mark. As you work with the project, the status of other elements in the block changes. If no properties are defined for an element in the block, its status is marked with a question mark. If calculations are required for an element, its status is marked with a lightning bolt, and if only connections or properties of the element need to be updated, its status is marked with .

When performing modeling in ANSYS WORKBENCH, it is sometimes necessary to establish links between individual engineering analysis blocks. The need to link blocks may be due to the specifics of the analysis, for example, the calculation of thermal stresses. In this case, the linked blocks must exchange information.

ANSYS WORKBENCH supports two types of connections between engineering analysis blocks: a simple connection for transferring data (**Transfer Data**) or a shared connection (**Share**).

With a simple link, data from one block is passed as input to another block. With a shared link, a correspondence is established between the linked cells, which imposes restrictions on editing and resetting data in dependent elements.

Figure 4.13 illustrates the scheme of coupled static thermal-strength analysis. In this problem, the properties of materials, geometric model, and FE mesh are used together, so the connection of these elements is of the **Share** type.

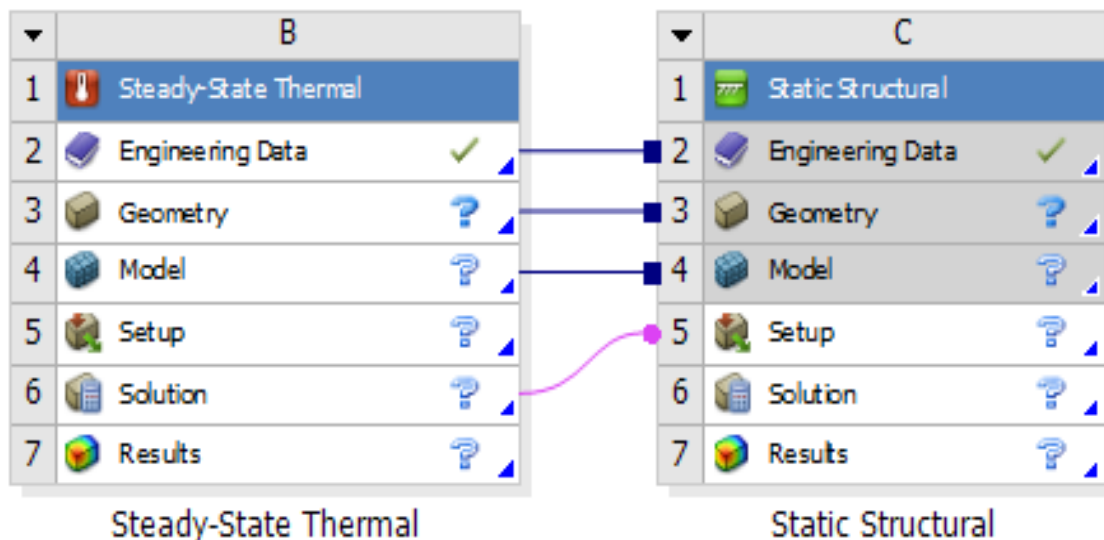


Figure 4.13 – Engineering Analysis Block Element Management Menu

The easiest way to insert a linked engineering analysis block is to click and hold the left mouse button on the desired block in the Toolbox window and move the pointer to the element of the existing block with which you want to establish a link. If you point to the **Geometry** element and release the left mouse button, a link will be established between the **Engineering Data** and **Geometry** elements. If you point to the Model element and release the left mouse button, a link will be established between the **Engineering Data**, **Geometry**, and **Model** elements.

The block elements that can participate in establishing connections are highlighted with a rectangle, and the types of connections between them are shown on the right (Figure 4.14).

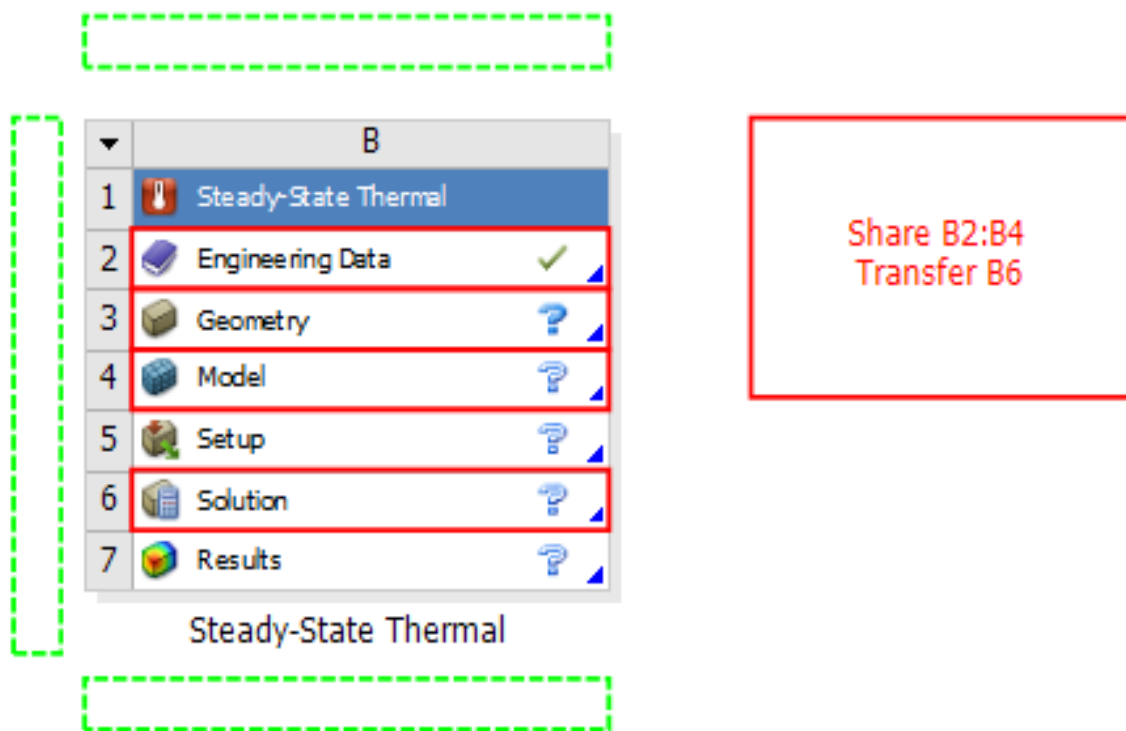


Figure 4.14 – Inserting a linked engineering analysis block

4.3.2 Setting material properties

An important stage of modeling is the selection of materials for the created bodies and the specification of their properties. For these purposes, ANSYS WORKBENCH has a materials management module associated with the analysis block and represented by the **Engineering Data** element.

You can select materials and set their properties using this module both before and after building a geometric model. The materials management module can be used as a repository for material property libraries.

After inserting a new engineering analysis block into the project, the **Engineering Data** element is already marked as defined, since structural steel is selected by default (Figure 4.15). You can change the material or its properties at any stage of modeling.

Outline of Schematic A2: Engineering Data					
	A	B	C	D	E
1	Contents of Engineering Data			Source	Description
2	Material				
3	Structural Steel				Fatigue Data at zero mean stress comes from 1998 ASME BPV Code, Section 8, Div 2, Table 5 -110.1
*	Click here to add a new material				

Properties of Outline Row 3: Structural Steel					
	A	B	C	D	E
1	Property	Value	Unit		
2	Material Field Variables	Table			
3	Density	7850	kg m ⁻³		
4	Isotropic Secant Coefficient of Thermal Expansion				
6	Isotropic Elasticity				
12	Strain-Life Parameters				
20	S-N Curve	Tabular			
24	Tensile Yield Strength	2,5E+08	Pa		
25	Compressive Yield Strength	2,5E+08	Pa		
26	Tensile Ultimate Strength	4,6E+08	Pa		
27	Compressive Ultimate Strength	0	Pa		

Figure 4.15 – Properties of structural steel

To launch the materials management module, you need to call up the context menu by right-clicking on the **Engineering Data** element line and select **Edit**. After loading, the module's workspace will appear, from which you can manage materials, access external sources of material data, and save data for further use (Figure 4.16).

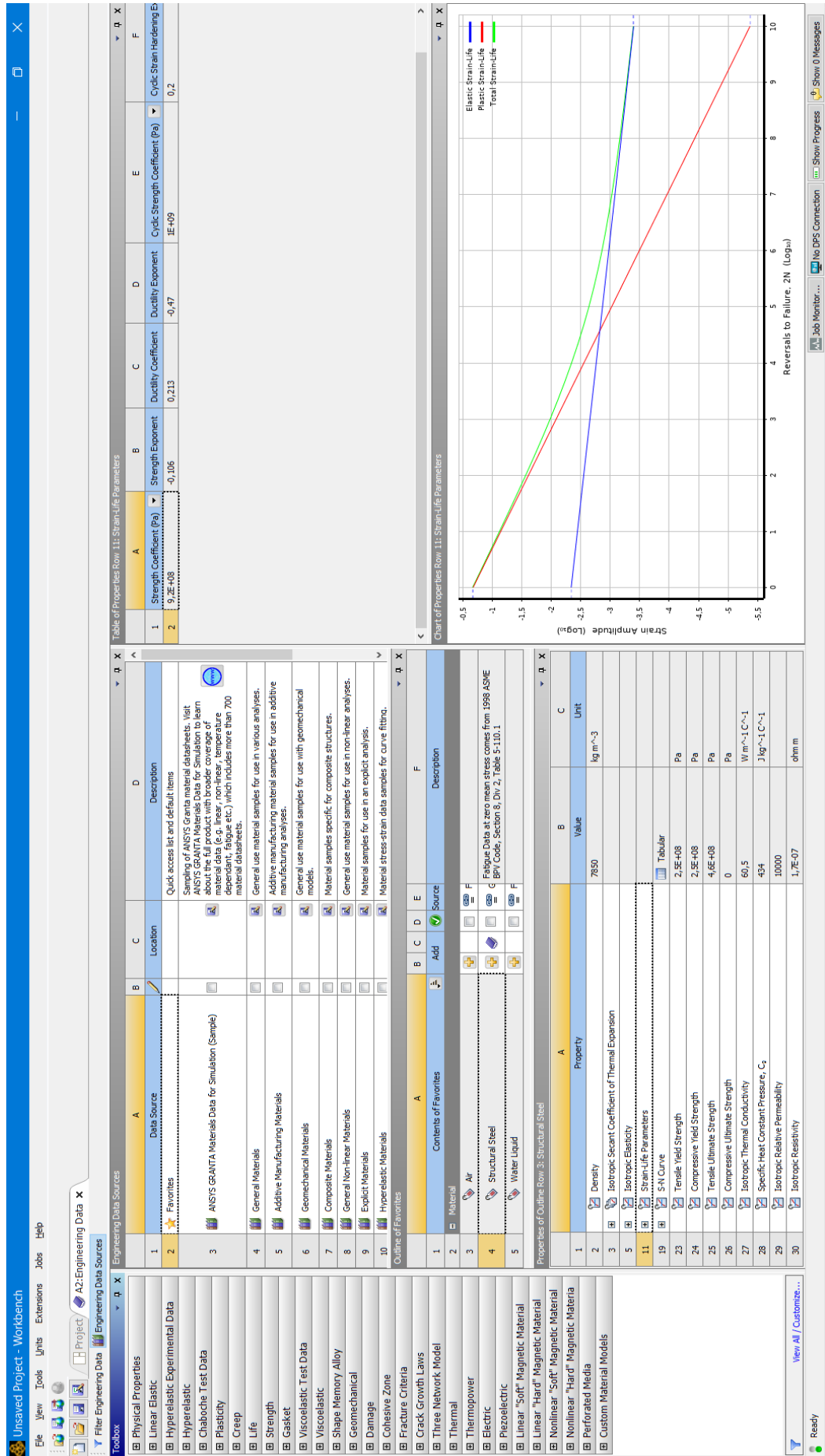


Figure 4.16 – Working area of the Engineering Data module

All work with the materials editor can be divided into two parts: work directly with the data source (with the **Engineering Data Source** tab active) and work with materials embedded in the project (the **Engineering Data Source** tab is not active).

On the left side of the material editor window is the **Toolbox** window, which displays the entire set of properties that can be set for each material.

The **Engineering Data Source** window contains data on material databases. The window becomes available after activating the corresponding tab in the upper left corner of the material editor window.

When working with a source, the **Outline of Schematic Materials** data window displays all materials stored in the specified source.

When working with materials embedded in a project (with the **Engineering Data Source** tab inactive), in the **Outline of Schematic** window you can create, delete or rename a material, add its description, add a material from an external data source or set a “default” material for specific parts of the model.

When working with a data source, the Contents of **Engineering Data** column of the **Outline of Schematic** window displays the names of all materials contained in the selected source. The icons to the left of the material names indicate whether the properties of the material have been entered correctly. The **Add** column is used to add a material from the data source to the current project. If the element is included in the current project, the corresponding icon lights up on the right. The **Source** column indicates the source where the material data is stored, and the **Description** column provides a brief description of the material.

When the **Engineering Data Source** tab in the upper left corner of the **Material Properties Editor** window is inactive, the **Outline of Schematic** window displays only those materials that are included in the current project. If necessary, you can delete or duplicate the selected material, view, update or delete the link to the data source in this window.

There are three types of data sources used in the Material Management module workspace: the current set of materials, the library of materials, and favorites. **General Materials** is a library of general-use materials that can be used in various types of analysis. **Materials** from the library are added to the current set of materials by clicking the button with a yellow plus in the **Add** column of the **Outline Panel**. We will incorporate the silicon properties from the **General Materials** library into the existing materials set as illustrated in Figure 4.17.

Engineering Data Sources				
	A	B	C	D
1	Data Source		Location	Description
2	★ Favorites			Quick access list and default items
3	ANSYS GRANTA Materials Data for Simulation (Sample)			Sampling of ANSYS Granta material datasheets. Visit ANSYS GRANTA Materials Data for Simulation to learn about the full product with broader coverage of material data (e.g. linear, non-linear, temperature dependant, fatigue etc.) which includes more than 700 material datasheets.
4	General Materials			General use material samples for use in various analyses.
5	Additive Manufacturing Materials			Additive manufacturing material samples for use in additive manufacturing analyses.
6	Geomechanical Materials			General use material samples for use with geomechanical models.

Outline of General Materials				
	A	B	C	D
1	Contents of General Materials	Add	Source	Description
7	FR-4		General_Materials.xml	sources and meant for illustrative purposes. It is assumed that the material x direction is the length-wise (LW), or warp yarn direction, while the material y direction is the cross-wise (CW), or fill yarn direction.
8	Gray Cast Iron		General_Materials.xml	
9	Magnesium Alloy		General_Materials.xml	
10	Polyethylene		General_Materials.xml	
11	Silicon Anisotropic		General_Materials.xml	
12	Stainless Steel		General_Materials.xml	
13	Structural Steel			Fatigue Data at zero mean stress comes from 1998 ASME BPV Code, Section 8, Div 2, Table 5-110.1

Properties of Outline Row 11: Silicon Anisotropic		
	A	B
1	Property	Value
2	Isotropic Thermal Conductivity	124

Outline of Schematic A2: Engineering Data				
	A	B	C	D
1	Contents of Engineering Data		Source	Description
2	Material			
3	Silicon Anisotropic		General_Materials.xml	
4	Structural Steel		General_Materials.xml	Fatigue Data at zero mean stress comes from 1998 ASME BPV Code, Section 8, Div 2, Table 5-110.1
*	Click here to add a new material			

Properties of Outline Row 3: Silicon Anisotropic				
	A	B	C	D
1	Property	Value	Unit	
2	Material Field Variables	Table		
3	Isotropic Thermal Conductivity	124	W m ⁻¹ C ⁻¹	

Figure 4.17 – Setting the properties of silicon

Let us set the properties of silicon as a new material. To accomplish this, open **Physical Properties** in the **Toolbox** and set the value of the thermal conductivity coefficient (Figure 4.18).

All properties of the selected material are displayed in the **Properties of Outline Row** window. The Property column contains the names of the properties of this material and its parameters.

Along with the **Properties of Outline Row** window, the data on the material property is also displayed in the form of a table in the **Table of**

Properties window, and the **Chart of Properties** window displays a graph of the change in this property from the specified parameter. When working with materials embedded in the project (the **Engineering Data Source** tab is inactive), you can adjust the material properties in the **Properties of Outline Row** and **Table of Properties Row** windows.

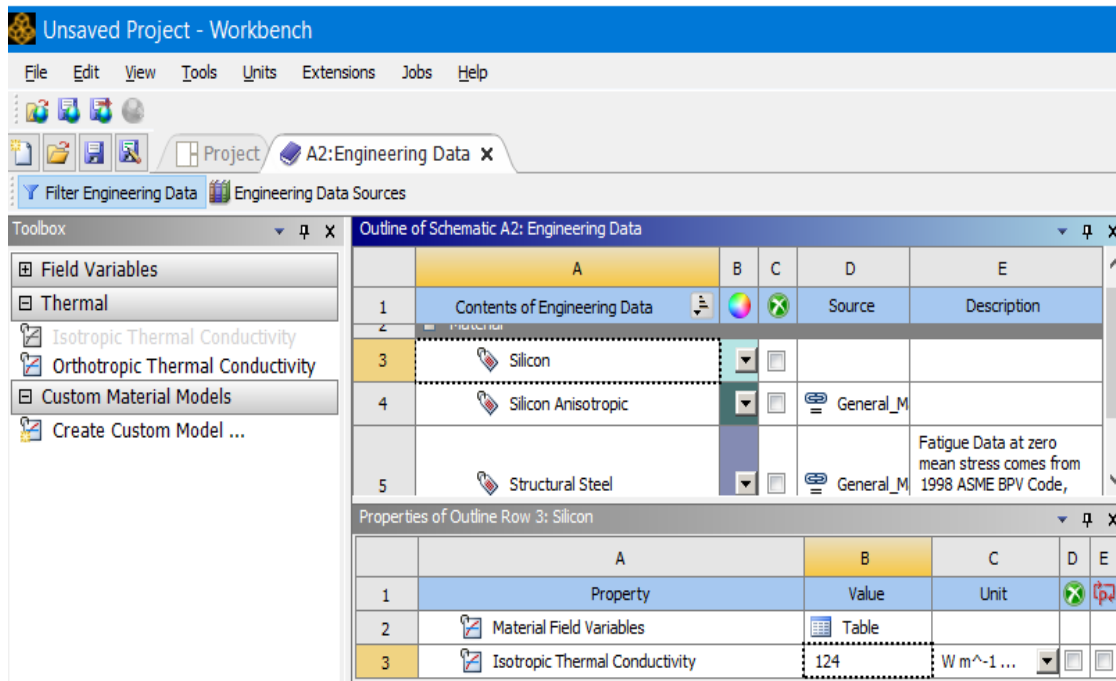


Figure 4.18 – Setting the properties of silicon (second option)

To change the material accepted by default, it is necessary to select the data source in **Engineering Data Source**, then, in the **Outline of Schematic** panel select the material and using the context menu add it to **Favorites** (**Add to Favorites** command). After that, in **Engineering Data Source** it is necessary to display the contents of the **Favorites** group and in the context menu, specify this material as the default material **Default Solid Material**.

4.3.3 Geometry construction and import

ANSYS WORKBENCH provides the following capabilities for geometric modeling:

- creating geometry from scratch;
- importing geometry from third-party CAD systems (AutoCAD, SolidWorks, etc.);
- editing imported geometry;
- bidirectional parametric linking with various CAD systems (import-export of a model).

The main ANSYS software modules used to create geometric models are:

- SpaceClaim Direct Modeler (Figure 4.19);
- Design Modeler (Figure 4.20);
- Mechanical APDL.

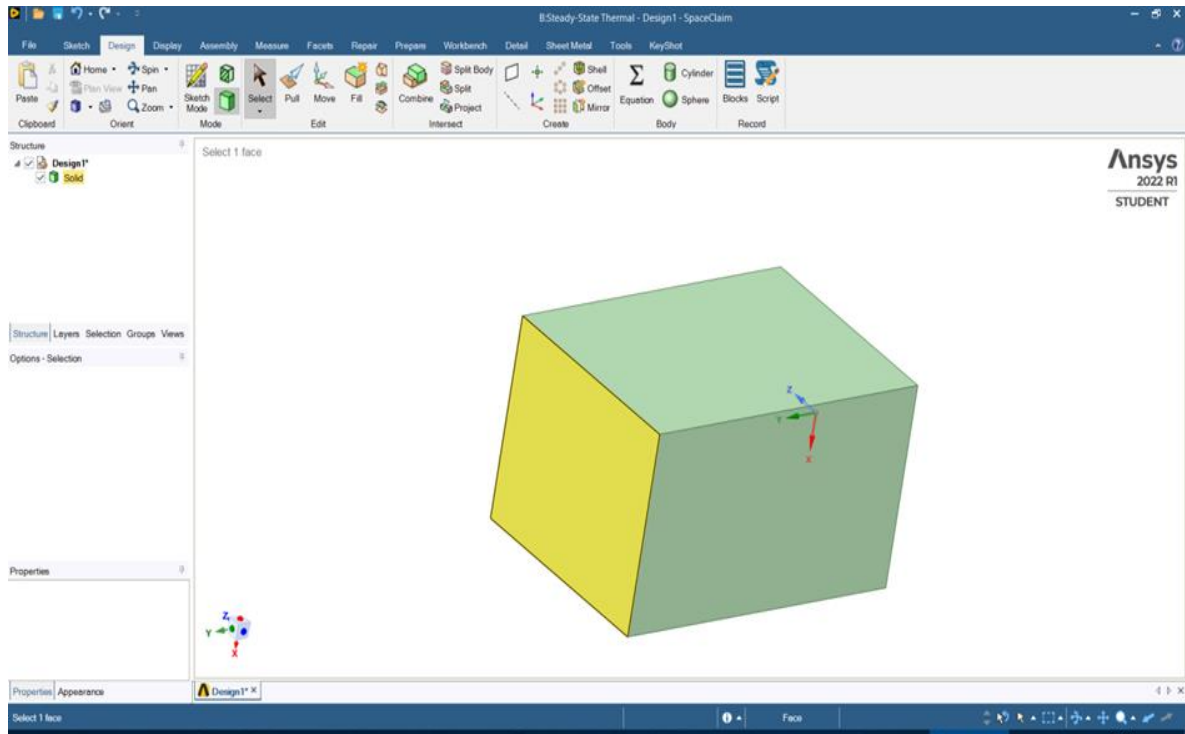


Figure 4.19 – SpaceClaim Direct Modeler

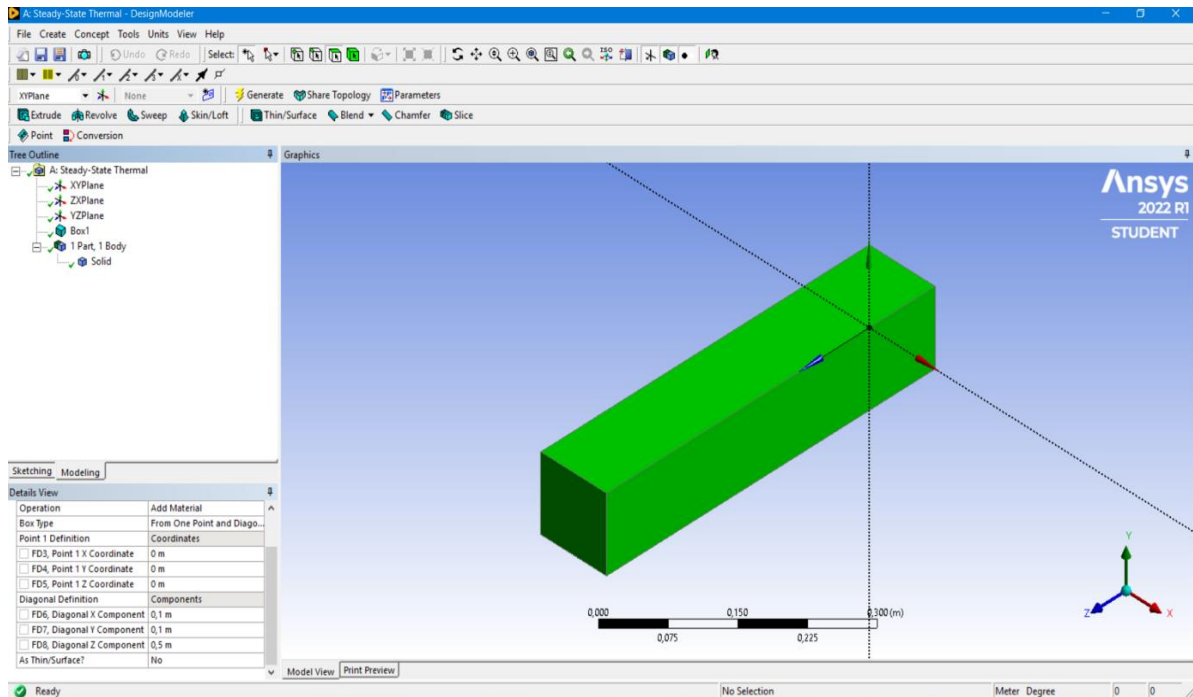


Figure 4.20 – Design Modeler

SpaceClaim Direct Modeler is a multifunctional 3D modeling application that provides an effective solution to common modeling tasks. The program is aimed at creating and editing geometry. Visual and intuitive tools enable the creation of objects for their subsequent multiple use. SpaceClaim is ideal for engineers who do not have time to master the “heavy” tools of the computer-aided design (CAD) system, who need a powerful and simple application for 3D modeling. SpaceClaim provides efficient manipulation of third-party models as well as the creation of original models. Besides solid modeling, SpaceClaim possesses robust tools for analysing 3D scanning outcomes and optimizing models for 3D printing. Using SpaceClaim enables the rapid acquisition of a model of the scanned product for subsequent use in production.

The Design Modeler geometric processor is based on the Parasolid kernel and has geometry parameterization capabilities. Design Modeler is a stand-alone application and is launched via ANSYS WORKBENCH from templates in the **Toolbox** panel. The geometric model is constructed by performing a sequence of operations that transform the geometry. All stages of geometry construction are stored as a tree structure, which allows changing the initial parameters at any time during geometry creation. Design Modeler allows you to create geometric models using graphic primitives, operations with them, and their parametric description.

Building a solid model in Design Modeler is possible using a combination of two options:

- using a set of ready-made primitives and applying Boolean operations to them;
- using a sequential hierarchical construction of model elements, starting with reference points, then lines, splines and then solids. Design Modeler has the means to import geometric models from most popular CAD systems, and also supports reading geometric formats .iges, .sat, .step, etc.

The stages of working with a geometric model have a structural representation in the form of a tree. Each object in the tree has a set of properties that can be supplemented and edited.

The Design Modeler module interface includes the following windows and toolbars.

The main menu includes a set of commands for managing project files, displaying the model in the graphic window, local project settings, selecting units of measurement, as well as a set of commands for constructing 2D/3D geometric models and reference information about the software product. The key commands of the main menu are listed below:

- **File** (commands for working with project files: saving, renaming, importing, exporting, etc.);

- **Create** (tools for generating 3D bodies from sketches, along with operations for their modification, including logical operations, deletion, movement, scaling, etc.);
- **Concept** (tools for conceptual modeling based on sketches: creating points on a plane, constructing lines based on a set of points, creating a plane based on a sketch, etc.);
- **Tools** (advanced functionality for working with 3D models, as well as tools for finalizing the model and conducting automated analysis of geometric defects);
- **Units** (selection of units of measurement);
- **View** (settings for the graphical display of the model);
- **Help** (guide for working with the software product).

Toolbars include a set of tools for quick access to frequently used commands, selection of geometry elements (vertices, edges, faces, 3D bodies) and options for customizing their display in the graphics window.

There are two main operating modes in the Design Modeler module: **Modeling** and **Sketching**, switching between which is carried out using the tabs of the same name.

The **Modeling** mode displays the project tree (Tree Outline), which contains a list of Cartesian planes in the global coordinate system, as well as all operations performed to build the geometry, considering their interrelations and the sequence of adding to the project. Each new operation is a tree object and has a list of properties and parameters displayed in the **Details View** window. The **Sketching** mode is designed to build 2D sketches of models.

The **Graphics** window is the working window of the project, in which the geometric model is displayed.

Below the graphic window there is an information panel, which displays information about the selected units of measurement, the status of the currently running operation, and data about the element of the geometric assembly selected in the graphic window.

The development of a geometric model begins with the creation of initial geometry, which is then transformed into a final geometric model using the functions of the geometric processor. All actions performed during the creation of geometry are displayed in the **Tree Outline** (Figure 4.21). In most cases, the basis of the geometric model is a two-dimensional sketch, which is transformed into a 2D/3D geometric object using special operations.

During the geometry construction phase, one can import a model generated in an external CAD system, correct (change, simplify) this geometry using the **Design Modeler** module and prepare it for the creation of a computational grid, considering the chosen type of numerical analysis.

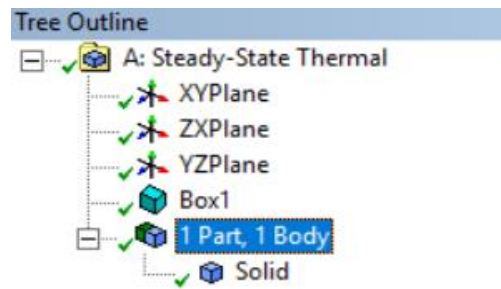


Figure 4.21 – Project tree

To review the model import settings, add the **Geometry** template to the **Project Schematic** window. Most of the settings for importing models from CAD systems are located in the **Basic Geometry Options** and **Advanced Geometry Options** sections of the **Properties** window (Figure 4.22).

The main settings specify the types of objects transferred during import (3D bodies, surfaces, lines), as well as options for importing parameterized geometric models from CAD systems.

The Advanced Geometry Options section contains parameters that determine:

- Analysis Type: 2D or 3D;
- Use Associativity to transfer material properties, boundary conditions and model loads specified in a third-party CAD system;
- Import Coordinate;
- SmartCADUpdate: this option means that if some parts of the assembly have been changed in CAD, only the changed parts are updated in Design Modeler upon re-import.

WORKBENCH supports the following versions of CAD packages: AutoCAD 2012+, Autodesk Inventor 2012, Creo Elements/Direct Modeling 18.0, Creo Parametric (Pro/ENGINEER), NX 8.0, Parasolid 24.0, SolidEdgeST4 (104), SolidWorks 2011, ANSYS SpaceClaim Direct Modeler 2011+, Teamcenter 8.0, 8.1 and 8.3, CATIA.

There are two methods for importing CAD models into the WORKBENCH project: using unidirectional and bidirectional links. Unidirectional links are implemented using the following methods (Figure 4.23):

- main menu commands: File => Import in the ANSYS **WORKBENCH** project window;
- **Import Geometry** commands, called through the context menu of the **Geometry** component in the **Project Schematic** window;
- in the **Design Modeler** application window using the File => **Import External Geometry File** command, which allows importing several files with geometric models simultaneously.

Properties of Schematic C2: Geometry		
	A	B
1	Property	Value
2	General	
3	Component ID	Geometry 2
4	Directory Name	Geom
5	Notes	
6	Notes	
7	Used Licenses	
8	Last Update Used Licenses	
9	Basic Geometry Options	
10	Solid Bodies	☒
11	Surface Bodies	☒
12	Line Bodies	☐
13	Parameters	Independent
14	Parameter Key	ANS;DS
15	Attributes	☐
16	Named Selections	☐
17	Material Properties	☐
18	Advanced Geometry Options	
19	Analysis Type	3D
20	Use Associativity	☒
21	Import Coordinate Systems	☐
22	Import Work Points	☐
23	Reader Mode Saves Updated File	☐
24	Import Using Instances	☒
25	Smart CAD Update	☒
26	Compare Parts On Update	No
27	Enclosure and Symmetry Processing	☒
28	Decompose Disjoint Geometry	☒
29	Clean Geometry On Import	☐
30	Stitch Surfaces On Import	None
31	Mixed Import Resolution	None
32	Import Facet Quality	Source

Figure 4.22 – Model import settings window

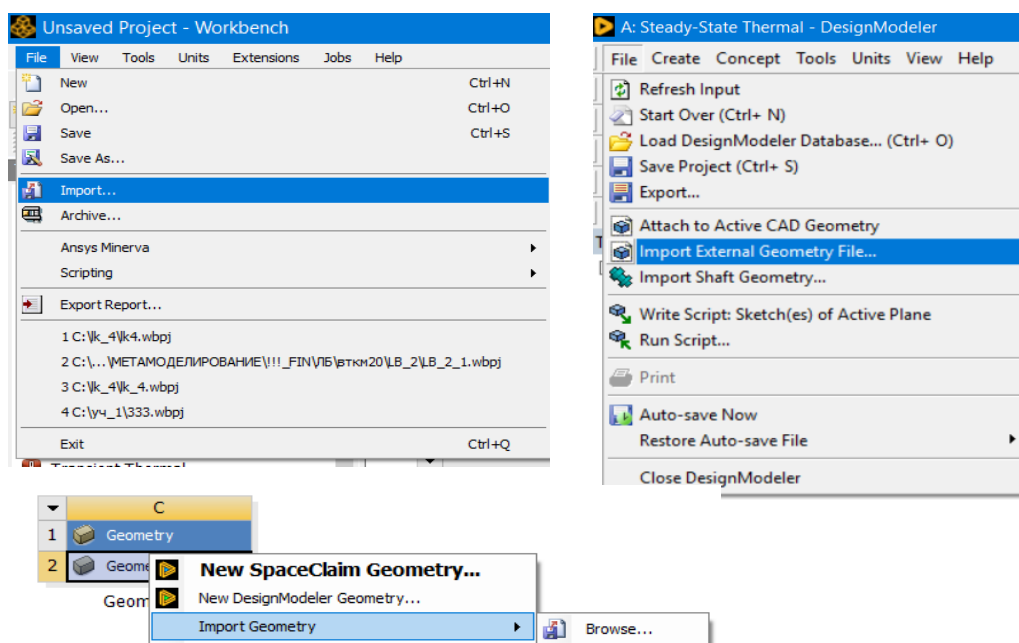


Figure 4.23 – Methods of importing CAD geometry

Bidirectional communication is achieved via calling the ANSYS WORKBENCH application through a CAD system and allows parameters to be exported from CAD to ANSYS WORKBENCH, as well as automatically tracking changes to the model made on one side of the communication (in the CAD system or in ANSYS WORKBENCH).

4.3.4 Finite element model generation

ANSYS WORKBENCH has several tools for constructing finite element meshes. Depending on the type of problem being solved, the following mesh preprocessors can be used:

1) ANSYS Meshing is a universal 2D/3D mesh preprocessor that allows you to reproduce structured and unstructured meshes based on prismatic, hexa- and tetraelements. For 2D areas, the generator enables the construction of meshes using tetrahedral and triangular elements. The advantage of the Meshing mesh preprocessor is its seamless integration into the WORKBENCH environment;

2) ANSYS TurboGrid is designed to automatically build high-quality hexahedral meshes for various types of blade machines (propellers, fan blades, axial pump impellers, etc.);

3) ANSYS ICEM CFD is one of the powerful modern mesh generators that facilitates to expand the functionality of ANSYS Meshing preprocessors. It includes tools for processing complex geometries and assemblies, correcting model defects, and constructing high-quality meshes based on hexa- and tetraelements. The mesh generator includes a set of methods for monitoring the quality of computational cells and is compatible with popular geometric formats. ANSYS ICEM CFD is a preprocessor independent of ANSYS WORKBENCH, which can be called both separately and from under ANSYS WORKBENCH.

Let us examine the fundamentals of operating within the Meshing mesh preprocessor. The Meshing mesh preprocessor is fully integrated into ANSYS WORKBENCH and allows you to create finite element meshes for various types of engineering analysis. Working in Meshing conceptually repeats the process of creating geometric models in Design Modeler, described above. Below we will consider the main features of working in the Meshing preprocessor.

The Meshing mesh preprocessor can be launched through the WORKBENCH environment interface in two ways.

The first method involves using the preprocessor as a separate component, which can then be linked to one or more other modules. To do

this, find the **Mesh** component in the **Component System** panel in the **Toolbox** and double-click the left mouse button on the component or, holding down the left mouse button, “drag” the component from the toolbar to the **Project Schematic** project workspace. Next, double-click the left mouse button on the **Mesh** component added to the project (or call the context menu of the component by right-clicking and selecting Edit) to call the mesh generator (Figure 4.24).

The second method involves the use of a grid pre-processor within ready-made templates designed for calculating problems of hydrodynamics, electromagnetism, mechanics of deformable solids, etc.

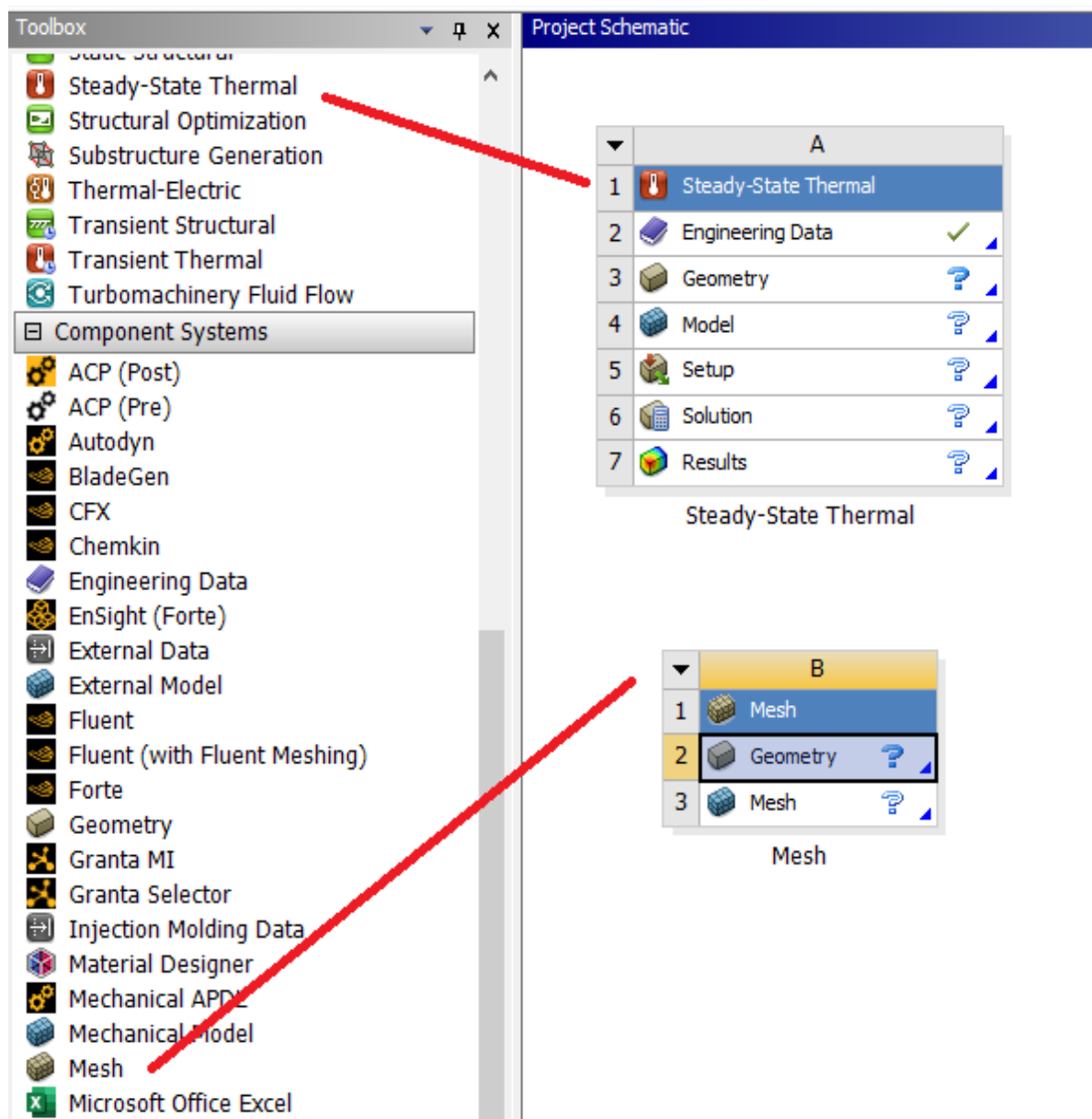


Figure 4.24 – Methods for starting Meshing

As already noted, ready-made templates are located on the Analysis System panel in the **Toolbox**. To add a template to the project window, double-click the left mouse button on the template or, holding down the left mouse button, “drag” the template from the toolbar to the project workspace (Figure 4.24). Next, double-click the left mouse button to select the **Mesh** stage in the added template.

To generate the finite element mesh, we launch Steady-State Thermal Mechanical (Figure 4.25). Steady-State Thermal Mechanical is a universal shell for various solvers and has a convenient graphical interface (compared to the ANSYS Mechanical APDL shell). At the same time, ANSYS Mechanical APDL can act as an auxiliary modeling environment, having more extensive capabilities. It is advisable to use it in cases where the problem cannot be solved using ANSYS WORKBENCH alone.

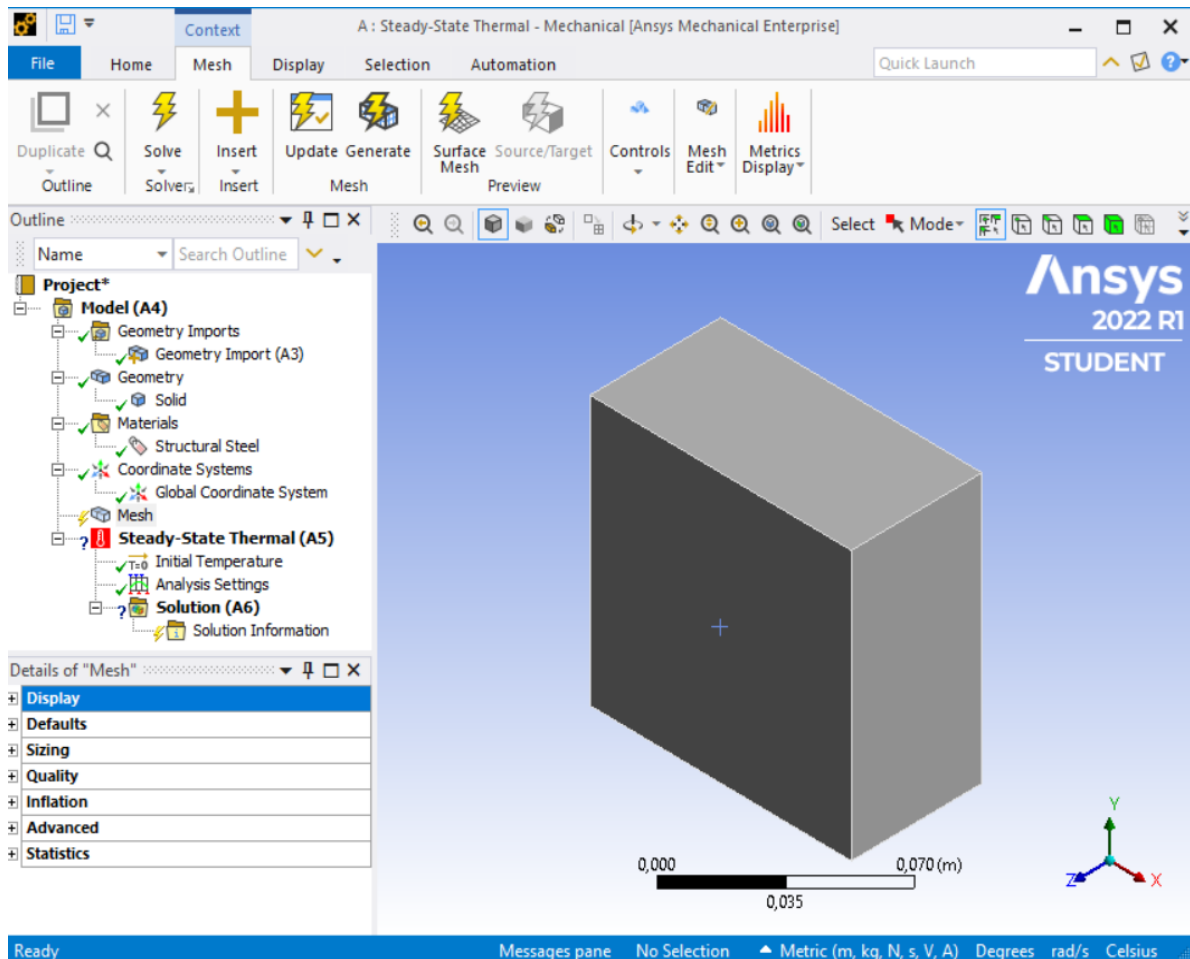


Figure 4.25 – Steady-State Thermal Mechanical application interface

The calculation model created in Steady-State Thermal Mechanical is represented in the APDL macro language and can be used to perform calculations in ANSYS Mechanical APDL. The model is transferred from

ANSYS WORKBENCH Mechanical to ANSYS Mechanical APDL via a text file containing a set of APDL commands (Figure 4.26). It is possible to transfer the geometric model, mesh, and calculation results back from ANSYS Mechanical APDL to ANSYS WORKBENCH Mechanical.

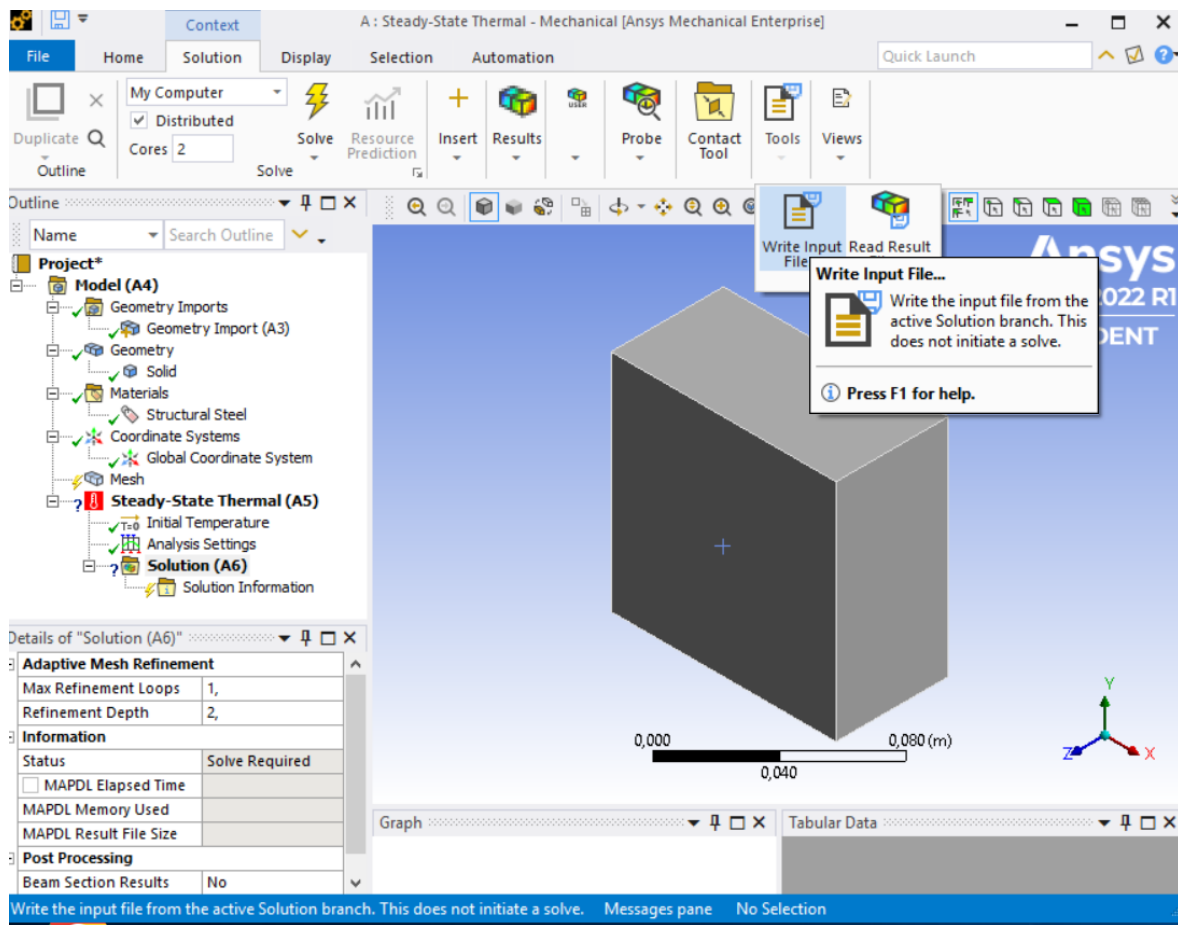


Figure 4.26 – Generating an APDL file to transfer a model from ANSYS WORKBENCH Mechanical to ANSYS Mechanical

The WORKBENCH Mechanical application interface has a structure similar to the Design Modeler application interfaces. Many interface elements in these applications are identical, for example, tools for managing the model view, selecting the mode for selecting geometric objects, etc. However, a number of tools and interface elements depend on the application used. The Steady-State Thermal Mechanical module interface consists of a main menu, a toolbar, a project tree window, a properties window, and a graphics window.

The calculation model in the Steady-State Thermal Mechanical interface is presented as a tree structure with many sections and objects and is located in the **Outline** window of the Steady-State Thermal Mechanical graphical interface (Figure 4.27). The calculation model is created by filling the project tree with objects, and the objects with information.

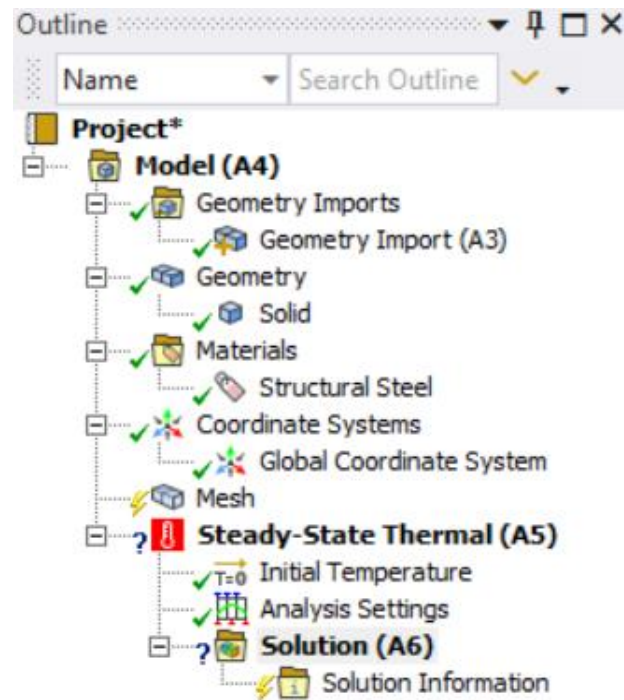


Figure 4.27 – Project tree for the Steady-State Thermal Mechanical module

To set the structure of the mesh and the shape of its cells, you need to use one of the methods integrated into **Meshing**. Access to the methods is provided through the context menu of the **Mesh** component in the **Outline** tree (**Mesh** => **Insert** => **Method**) (Figure 4.28) or through the **Mesh Control** => **Method** button on the **Mesh** toolbar, which appears in the Toolbars section when you select with the left mouse button.

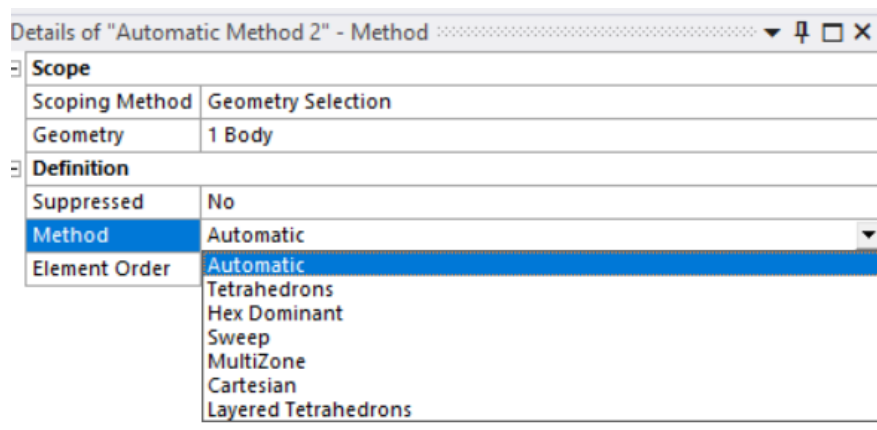


Figure 4.28 – Selecting methods integrated into Meshing

To build a computational mesh in Meshing, you need to set global and/or local settings that will determine the mesh characteristics: mesh structure, shape, size, and deformations of elements. Global settings determine the mesh characteristics for the entire model, while local settings

allow you to set individual properties for individual groups of mesh elements in the model. Local settings in most cases have a higher priority than global ones.

Access to global mesh settings is provided through the **Details of Mesh** window, which opens when you select the **Mesh** component of the **Outline** tree. Figure 4.29 shows the global settings window with drop-down lists of parameters.

Details of "Mesh"	
Display	
Display Style	Use Geometry Setting
Defaults	
Physics Preference	Mechanical
Element Order	Program Controlled
☐ Element Size	Default
Sizing	
Use Adaptive Sizing	Yes
Resolution	Default (2)
Mesh Defeaturing	Yes
☐ Defeature Size	Default
Transition	Fast
Span Angle Center	Coarse
Initial Size Seed	Assembly
Bounding Box Diagonal	0,15 m
Average Surface Area	6,6667e-003 m ²
Minimum Edge Length	5,e-002 m
Quality	
Check Mesh Quality	Yes, Errors
Error Limits	Aggressive Mechanical
☐ Target Element Quality	Default (5,e-002)
Smoothing	Medium
Mesh Metric	None
Inflation	
Use Automatic Inflation	None
Inflation Option	Smooth Transition
☐ Transition Ratio	0,272
☐ Maximum Layers	5
☐ Growth Rate	1,2
Inflation Algorithm	Pre
View Advanced Options	No
Advanced	
Number of CPUs for Parallel Part Meshing	Program Controlled
Straight Sided Elements	No
Rigid Body Behavior	Dimensionally Reduced
Triangle Surface Mesher	Program Controlled

Figure 4.29 – Global mesh settings window in Meshing

Tools for setting local mesh parameters are available through the context menu of the **Mesh** component of the **Outline** tree: **Mesh => Insert** (Figure 4.30). Local settings make it possible to set the properties of the computational mesh for individual parts of the geometric model (3D bodies, surfaces, edges).

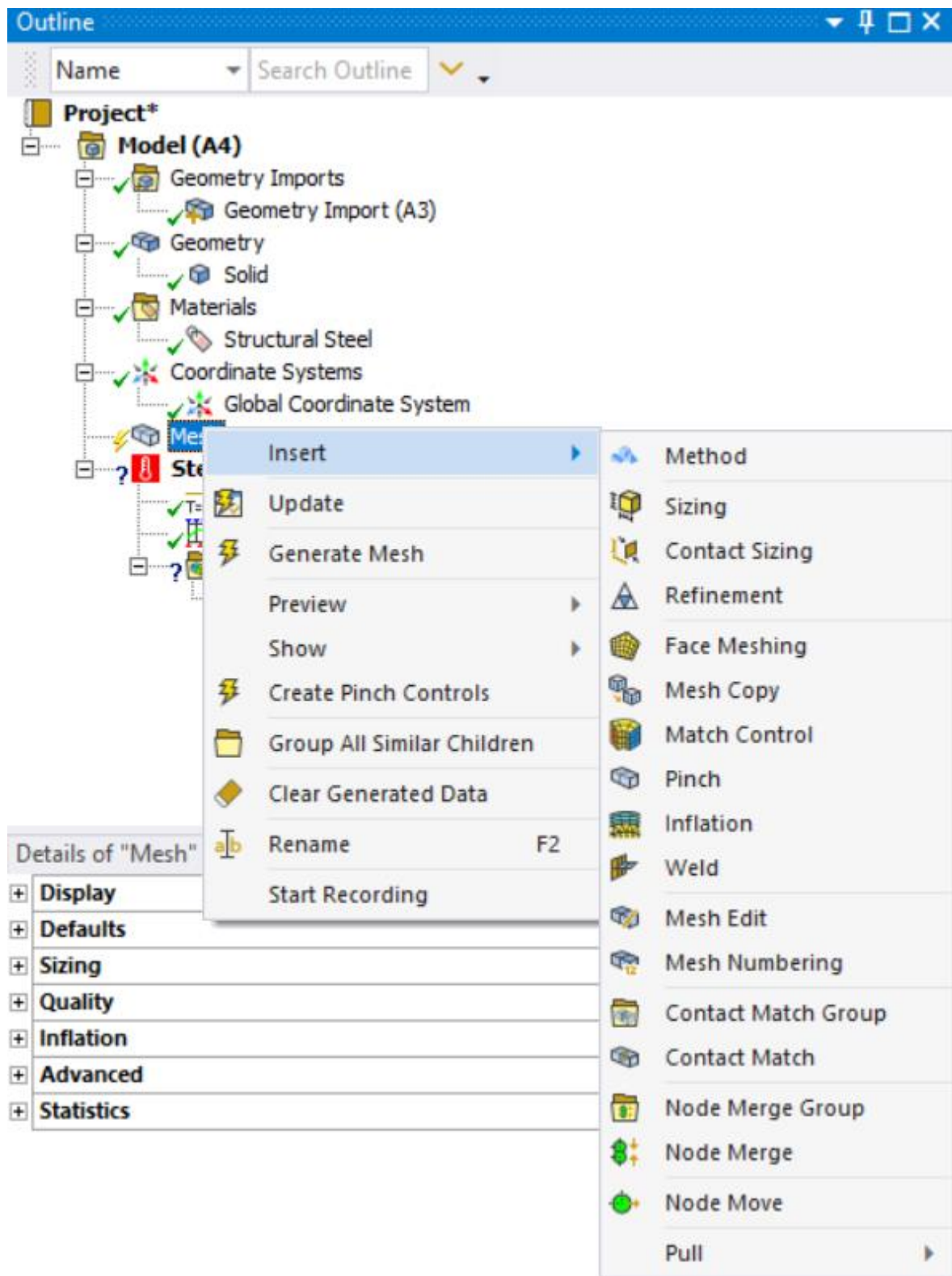


Figure 4.30 – Access to tools for setting local grid parameters

To generate a finite element (FE) mesh, right-click on the project tree item Mesh and then **Generate Mesh**. The resulting FE mesh is shown in Figure 4.31.

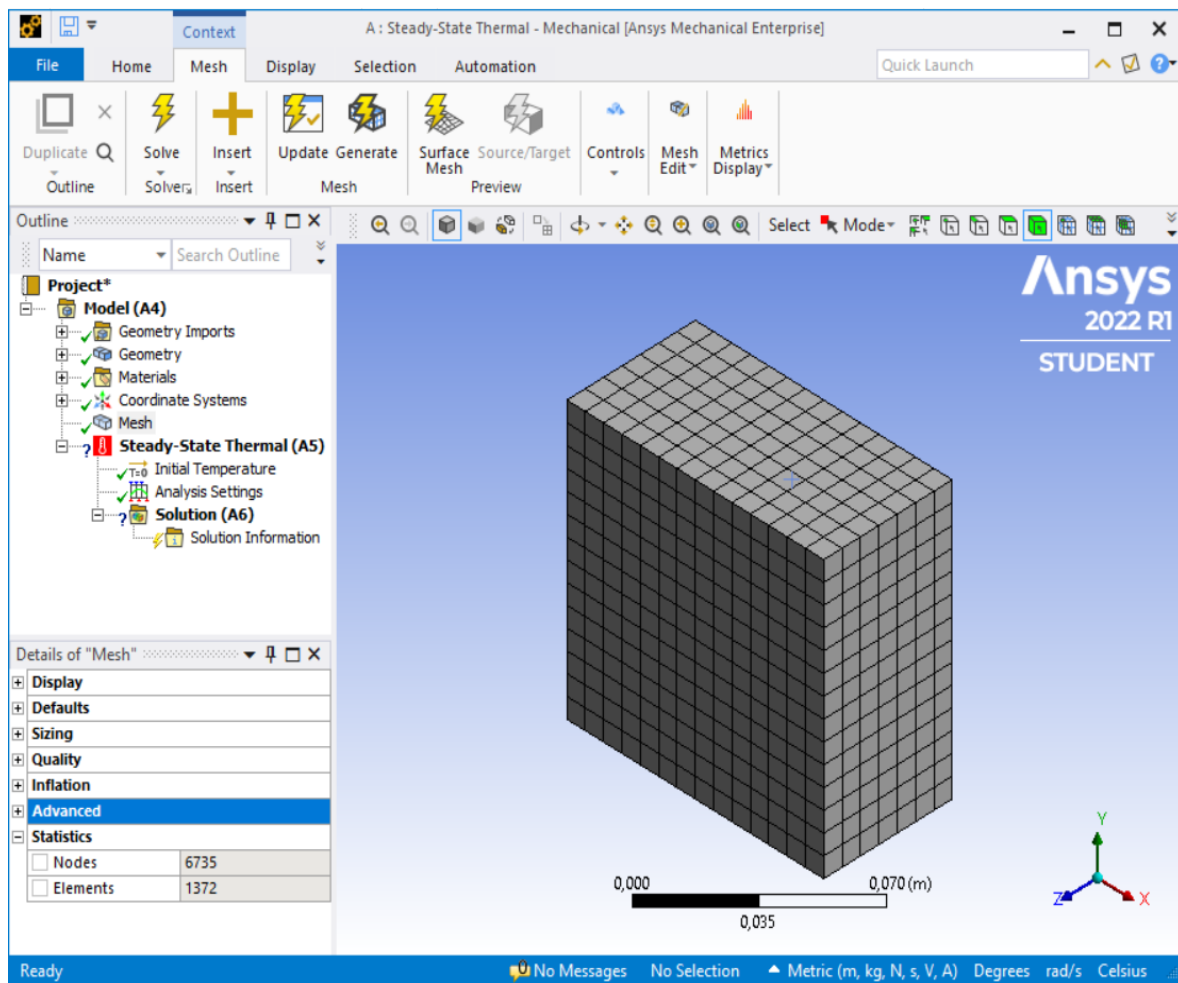


Figure 4.31 – Finite element model of the sample

4.3.5 Setting boundary conditions and loads

The next stage of preparation for the calculation is the definition of external effects (mechanical, thermal, etc.) on the structure. Usually, the external effect is determined at the boundary of the created model. The terms “limitation” and “load” are understood as all the various processes that occur both on the surfaces of a solid and at individual points inside it. “Limitation” in ANSYS WORKBENCH is a fixation, i.e., limitation of displacements and rotations in structural analysis, or determination of temperature when solving heat exchange problems, and “load” is the application of concentrated or distributed forces (structural analysis) or heat flows (heat exchange problems), etc.

When specifying boundary conditions and loads, the project tree section is used, which bears the name of the calculation module used as a solver, e.g., Steady-State Thermal (Figure 4.25). Boundary conditions are added using the section’s context menu (Insert) or the toolbar buttons (Figure 4.32).

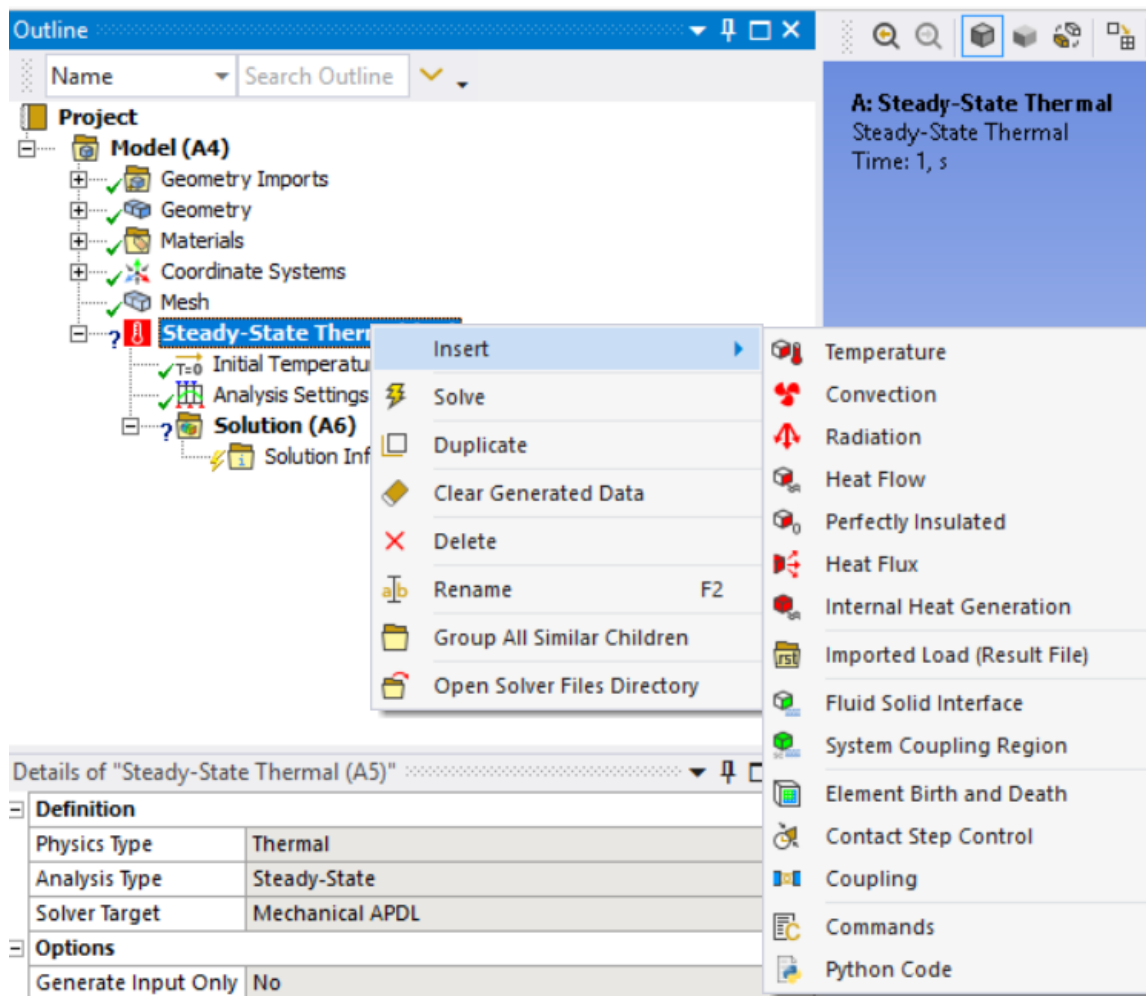


Figure 4.32 – Context menu of the Insert section

When performing thermal analysis, the following boundary conditions can be specified:

- **Temperature;**
- **Convective heat exchange with the environment;**
- **Radiation;**
- **Heat Flow;**
- **Heat Flux;**
- **Internal Heat Generation;**
- **Perfectly Insulated.**

Almost all loads have a **Geometry** parameter, which specifies the application object: point, edge, face, body. To set this parameter, you need to select the required object(s) and, by clicking the mouse on its field in the detail window, confirm your choice by pressing the **Apply** button. The reverse sequence is also possible: first, click the mouse in the **Geometry** parameter field, then select the objects and confirm your choice by pressing the **Apply** button.

For most types of loads, the value can be specified in three ways:

- as a fixed value (**Constant**);
- in tabular form (**Tabular**);
- as a functional dependency (**Function**).

By default, a fixed value is set. The method for specifying the load is selected by clicking the mouse on the arrow in the right part of the numerical value input window and then selecting one of the options (Figure 4.33).

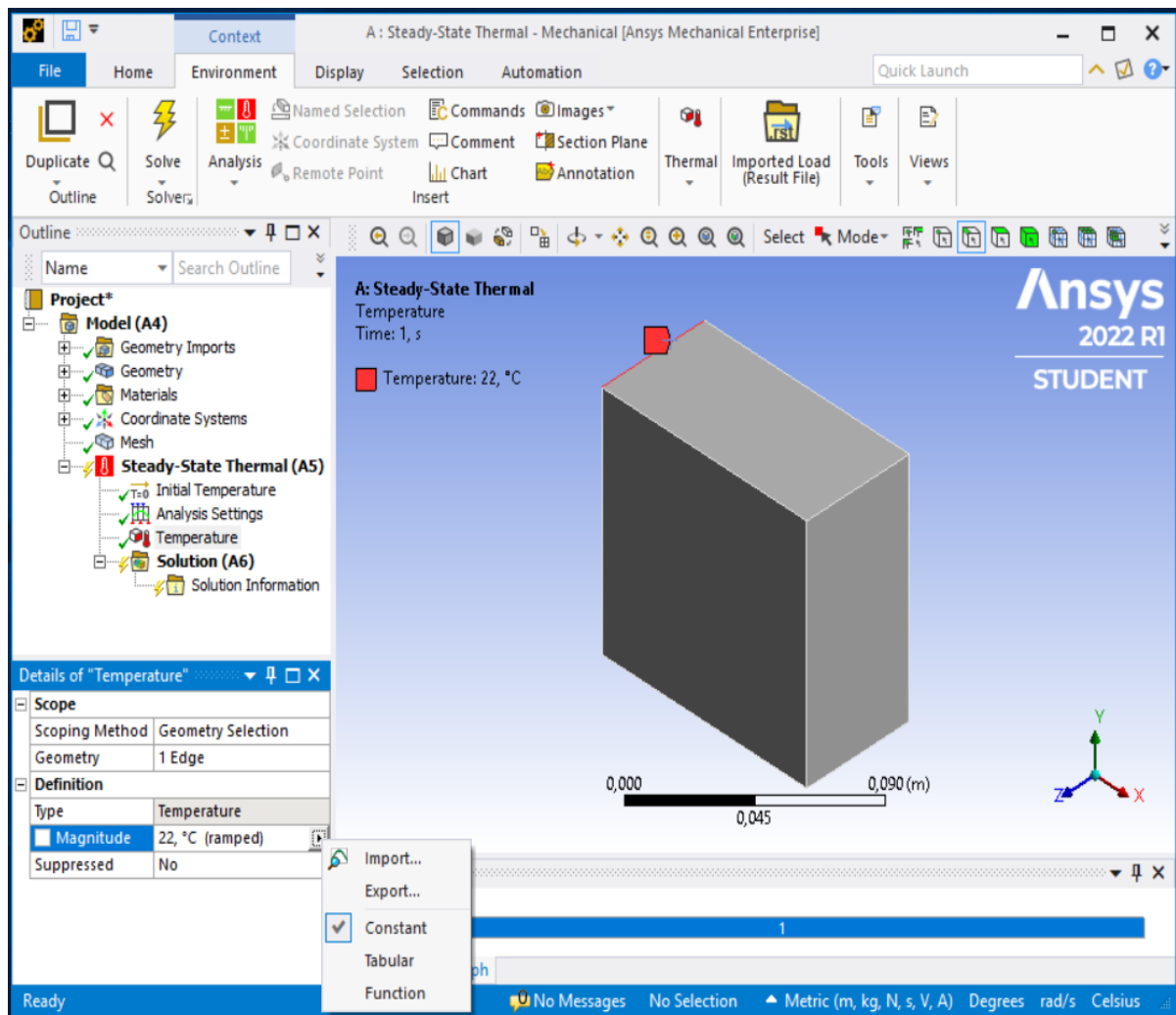


Figure 4.33 – Selecting the load assignment method

4.3.6 Calculation execution and results output

Solver parameters and options are set in the project tree section. They are all divided into groups, the number and composition of which depend on the type of the current engineering analysis. For Steady-State Thermal, the solver settings are shown in Figure 4.34.

The solver is launched in WORKBENCH by pressing the button with the lightning symbol on the toolbar.

Details of "Analysis Settings"	
☐	Step Controls
	Number Of Steps: 1,
	Current Step Number: 1,
	Step End Time: 1, s
	Auto Time Stepping: Program Controlled
☐	Solver Controls
	Solver Type: Program Controlled
	Solver Pivot Checking: Program Controlled
☐	Radiosity Controls
☐	Nonlinear Controls
	Heat Convergence: Program Controlled
	Temperature Conver...: Program Controlled
	Line Search: Program Controlled
☐	Advanced
☐	Output Controls
	Calculate Thermal Flux: Yes
	Contact Data: Yes
	Nodal Forces: No
	Volume and Energy: Yes
	Euler Angles: Yes
	General Miscellaneous...: No
	Contact Miscellaneous...: No
	Store Results At: All Time Points
	Result File Compres...: Program Controlled
☐	Analysis Data Management
☐	Visibility
	[A] Temperature: Display

Figure 4.34 – Solver settings

A window thereafter appears, displaying the overall progress of the calculating procedure.

The buttons in the specified window allow you to stop the solver if necessary (Stop Solution) or interrupt calculations while maintaining the ability to continue calculations (Interrupt Solution).

In ANSYS WORKBENCH, all outcomes are computed within the Solution group. These may consist of distributions of the fields of the examined quantities across the entire model or a portion thereof, or graphs illustrating the dependence of certain parameters on time or coordinates.

Insertion of results can be done in two ways: via the context menu, called by left-clicking on the Solution group in the project tree (Figure 4.35), or by selecting the required element on the toolbar at the top of the working window. This operation can be done either before or after starting the solver.

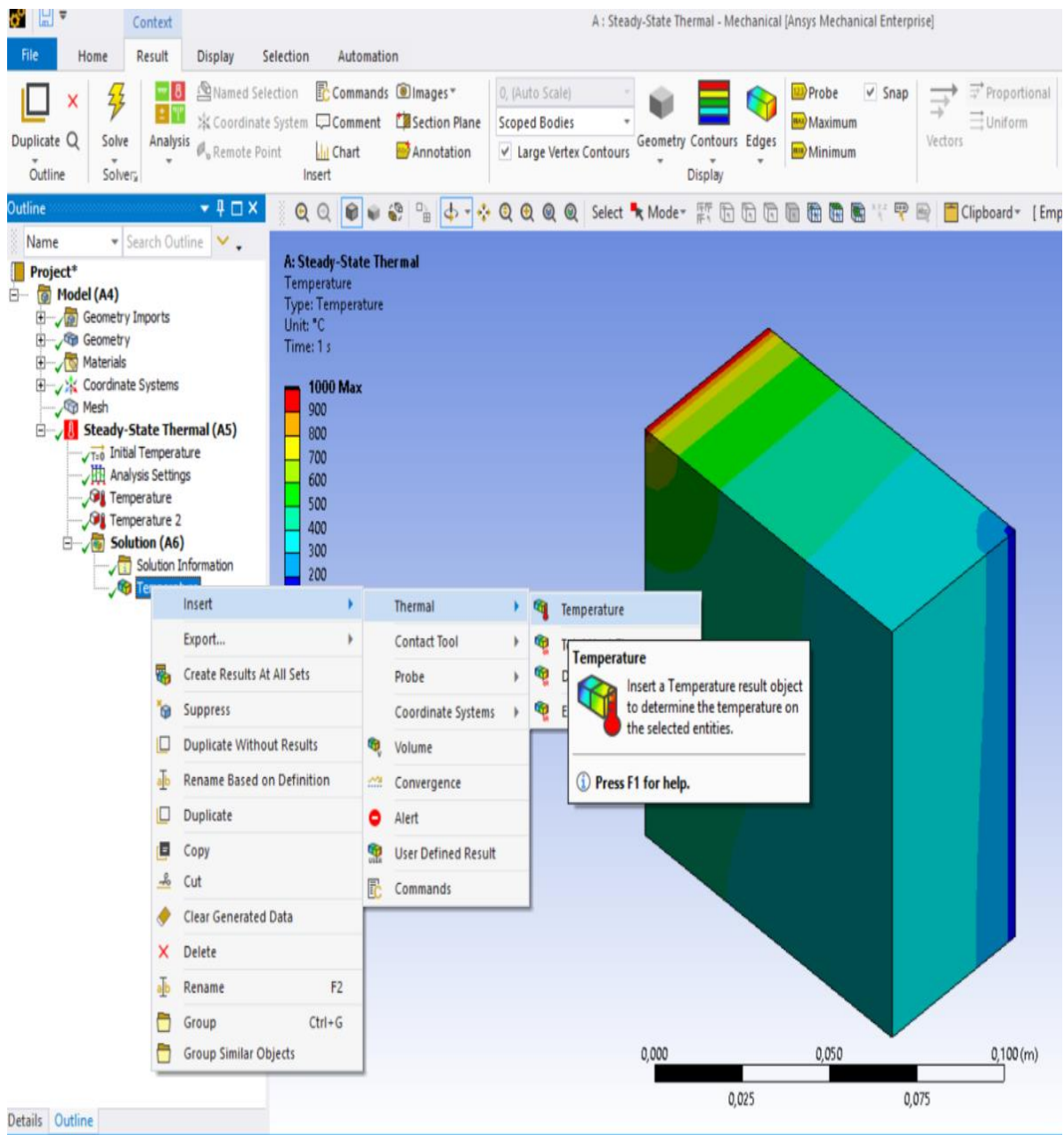


Figure 4.35 – Inserting calculation results

ANSYS WORKBENCH has built-in capabilities for generating a report on the simulation performed. To proceed to generating a report, you need to find the **Report Preview** tab and activate it (Figure 4.36).

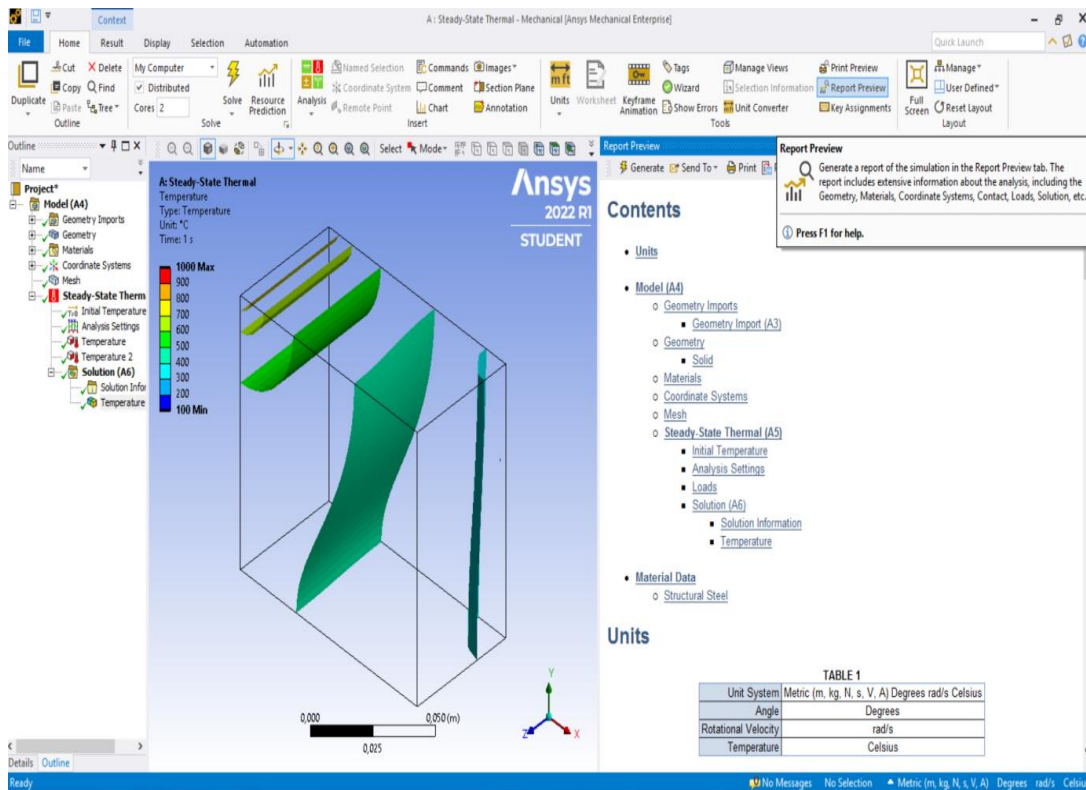


Figure 4.36 – Report generation

Self-Assessment Questions

- 1 Enumerate the main modules and applications used in ANSYS WORKBENCH for project execution.
- 2 Which files are generated during the operation of the system?
- 3 Describe the structure of the Steady-State Thermal static strength analysis project.
- 4 What are the main tasks that can be solved using a LOG file?

5 IMPLEMENTATION EXAMPLES FOR ANSYS ANALYSIS SOFTWARE

Problem 5.1 “Steady-State Thermal Analysis of Plate Heating in 3D Formulation”

5.1.1 Problem statement and assumptions

This example enables familiarization with the interface of the **ANSYS 2019R1** Academic program, and demonstrates the calculation of stationary temperature fields within a cube with specified fixed temperature values on the model faces.

A stationary problem is solved. A rectangular parallelepiped with a side ratio of 1:1:1 maintains a uniform temperature on two distinct faces. A steady temperature distribution in a cube must be achieved. The influence of temperature on material properties and thermal exchange with the environment is not considered. The outcome of resolving the issue is illustrated in Figure 5.1.

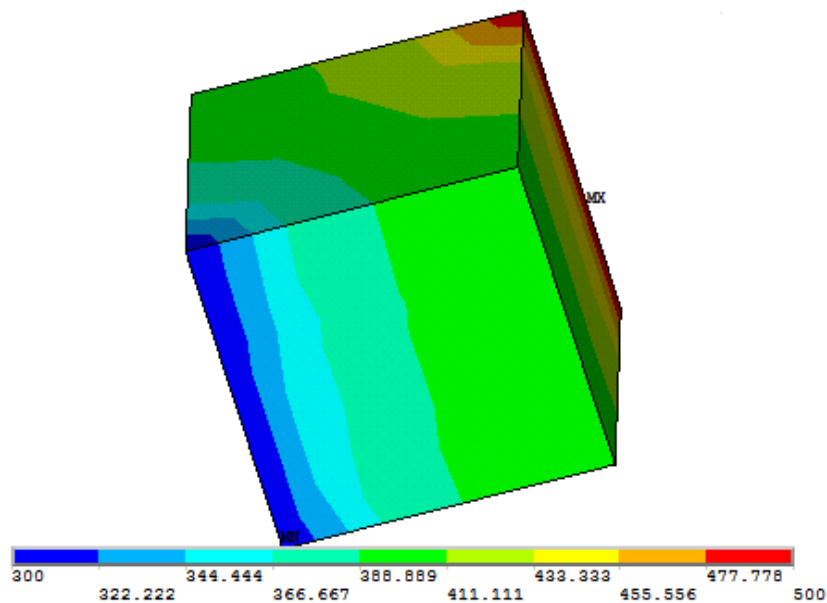


Figure 5.1 – Final temperature distribution in the cube

5.1.2 Implementation of the problem using the program’s graphical interface

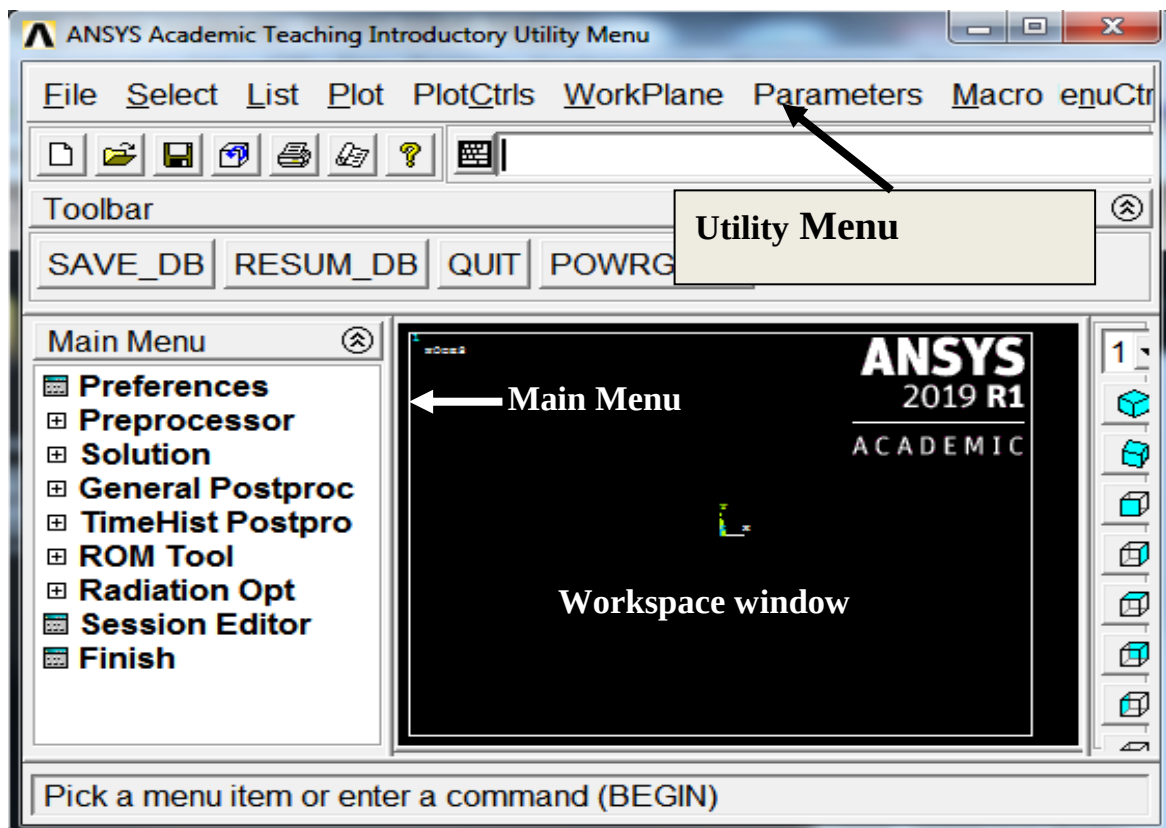
1 The first step in solving the problem is to launch the program. To do this, you must first launch the product launcher. Select **Start**, then **All**

Programs (depending on the operating system version, this item may differ), after that, open the **ANSYS 2019R1** folder and click on **Mechanical APDL Product Launcher 2019R1**. In the future, in order to simplify the perception of the material, a similar sequence of actions required to complete any of the task items will be presented as follows:

Start→All Programs →Ansys 2019R1→Mechanical APDL Product Launcher.

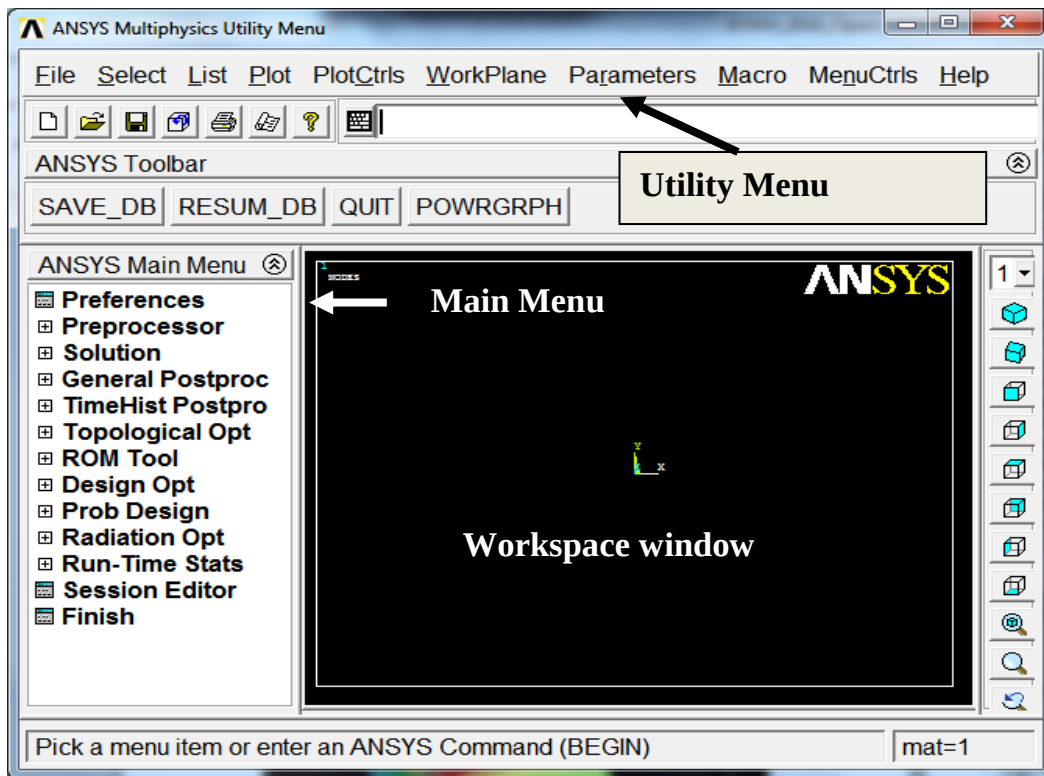
In the window that opens, in the **File Management** tab, you must specify the working folder and the current file name. In the **Working Directory** line, specify the path to the folder in which all data and databases will be saved when performing calculations. In the **Job** name line, enter the default file name for files created by the program during calculations, and run the program using the Run button.

Figure 5.2 (a) shows an example of a typical **ANSYS 2019R1** window, while Figure 5.2 (b) depicts a comparable window from the prior version **ANSYS 12.1**.



a

Figure 5.2 – Typical ANSYS window, Sheet 1



b

Figure 5.2 – Typical ANSYS window, Sheet 2

The **Utility Menu** contains a set of frequently used procedures that are accessible at any point during program operation. The **Main Menu** contains the main functions and stages of program execution, grouped into pop-up (dynamic) menus located on the side, whose visibility is depends on the user's progression through the program.

2 The working folder and the current file name can be changed at any stage of the task. To change the current file name, select **Utility Menu→File→Change Jobname**, enter a new file name and click **OK**. To change the working folder, follow the sequence **Utility Menu→File→Change Directory**, specify a new folder and click **OK**.

3 ANSYS has the ability to display alphanumeric information in the working window at all stages of solving a problem, which is usually used as a task title. To do this, you need to execute the sequence **Utility Menu→File→Change Title**, enter the name of the title that will be displayed on the main screen and click **OK**.

4 By default, ANSYS sets the background to black. To change the background color to white, you need to invert the color: **Utility Menu→PlotCtrls→Style→Colors→Reverse Video**.

5 The task we are considering is heat distribution. In order for the **Main Menu** to display items only for this type of task, you need to enable filters: **Main Menu**→**Preferences**→ **select Thermal**→**OK**.

6 Determine the element type for applying the finite element mesh and solving the heat transfer problem. Use the 8-node **SOLID70** element: **Main Menu**→**Preprocessor**→**Element Type**→ **Add/Edit/Delete**→ **Add**. Select **Thermal Mass**→**Solid**→**Brick 8 node 70** from the list→**click OK**→**close the Close** window.

7 Select temperature units – **Kelvin**: **Main Menu**→**Preprocessor**→**Material Props**→**Temperature Units**→ select **Kelvin or Rankin**→press **OK**.

8 Set the material properties. To solve the heat propagation problem, it is enough to determine the density, specific heat capacity and thermal conductivity of the material: **Main Menu**→**Preprocessor**→**Material Props**→ **Material Models**. In the **Material Models Available** window, select **Thermal**→**Density**→enter any value in the **DENS** line→**click OK**. Select **Specific Heat**→enter any value in the **C** line→**click OK**. Then, **Conductivity**→**Isotropic**→enter any value in the **KXX** line→**click OK**→**Close**.

9 The next step of the solution is to build a solid model. For example, consider a cube with a side of 1 meter: **Main Menu**→**Preprocessor**→**Mdeing**→**Crate**→**Voumes**→**Block**→ **By Dimensions**. In the **Create Block By Dimensions** window that opens, enter **X1=0, X2=1, Y1=0, Y2=1, Z1=0, Z2=1**→**click OK**.

10 Apply the finite element mesh. In this task, we do not ask the program to apply the mesh according to a pre-defined template, but provide a free choice: **Main Menu**→**Preprocessor**→**Meshing**→**Mesh**→**Voumes**→**Free**→**left-click** to select the model→press **OK**.

11 After preprocessing, we proceed to the stage of applying loads and obtaining a solution. We maintain two arbitrary edges of the cube at a constant temperature: **Main Menu**→**Solution**→**Define Loads**→**Apply**→**Thermal**→**Temperature**→**On Lines**→**click** any edge of the cube→**click Apply**. In the opened **Apply TEMP on Lines** window, in the **Load TEMP** value line, enter the temperature value for the first edge. In the **DOF's to be constrained** line, select **Temp**→**click Apply**. **Click on any line**→**click Apply**. In the opened **Apply TEMP on Lines** window, in the **Value Load Temp** line, enter the temperature value for the second edge. In the **DOF's to be constrained** line, select **Temp**→**click OK**.

12 This problem requires the determination of a steady-state temperature distribution. This must be considered while selecting the type

of analysis. The chosen analysis type instructs the program on the solution equations to employ for resolving this issue. We shall designate the analysis type as steady-state: **Main Menu**→**Solution**→**Analysis Type**→**New Analysis**→select **Steady-State**→press **OK**.

13 Launch the problem for solution: **Main Menu**→**Solution**→**Solve**→ **CurrentLS**→press **OK**. Wait for completion (The message **Solution is done!** will appear).

14 The last stage of solving the problem is post-processing of the results. We display the results of the solution graphically as a spatial distribution of temperature fields: **Main Menu**→General Postproc→Plot Results→Contour Plot→Nodal solution. In the **Contour Nodal Solution Data** window, select **Nodal Solution**→**DOF Solution**→**Nodal Temperature**→click **OK**. The temperature distribution with a color legend should appear on the screen.

15 Alphanumeric and graphic information displayed on the **Working Window** can be customized to suit your needs: **Utility Menu**→**PlotCtrls**→**Window Controls**→**Window Options**. Then, in the **Window Options** window, select the necessary items:

- change the temperature legend display type – in the **Display of Legend** line, select **Legend ON** → click **Apply**;
- disable the legend header display – in the **Legend** header line, uncheck **the box** → click **Apply**;
- disable the display of some numerical model parameters above the legend – in the **View** portion of legend line, uncheck **the box** → click **Apply**;
- disable the display of the working window frame – in the **Window frame line**, uncheck **the box** → click **Apply**;
- disable the display of the title on the screen – in the **Title line**, uncheck **the box** → click **Apply**;
- change the display of the ANSYS logo – in the **ANSYS logo display line**, select **Text in legend** → click **Apply**;
- disable the display of the date and time – in the **DATE/TIME display line**, select **No Date or Time** → click **Apply**;
- disable the display of the coordinate system on the screen – in the **Location of triad line**, select **Not shown** → click **OK**.

16 The calculation results can be saved as a database file with the .db extension, which can be loaded in subsequent work sessions to obtain the calculation data in graphical or tabular form: **Utility Menu**→**File**→**Save As**→select **the folder** in which the database is saved and **specify the file name in Latin letters**→click **OK**.

17 The ANSYS working window can be saved as an image file with the .bmp extension using the **Utility Menu: Utility Menu→PlotCtrls→Capture Image**. In the window that appears, select **File→Save as→** select a folder and specify a file name→**Save**.

Problem 5.2 “Transient Analysis of Ingot Solidification in 3D Formulation”

5.2.1 Problem statement and assumptions

This example facilitates familiarity with approaches for handling three-dimensional non-stationary thermal problems, considering the temperature dependency of material parameters alongside a heat transfer coefficient.

The ingot, the cross-section of which is shown in Figure 5.3, is 50 cm long and, after being cast into a sand mold 4 cm thick, cools together with it in the air.

The problem of non-stationary thermal conductivity with conditions of the third kind at the sand-air boundary is solved. The cooling process is monitored for 10 minutes. The problem considers the dependence of the thermal properties of steel on the temperature of the material, as well as the presence of heat transfer from the surface of the sand. Due to the symmetry of the cross section, only half of the region is taken into account.

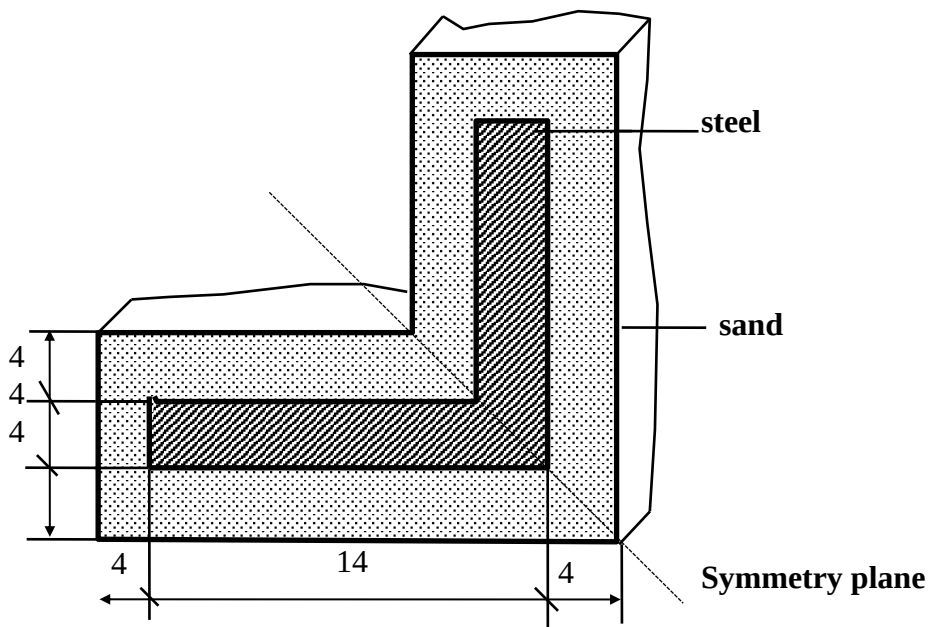


Figure 5.3 – Cross section of the model

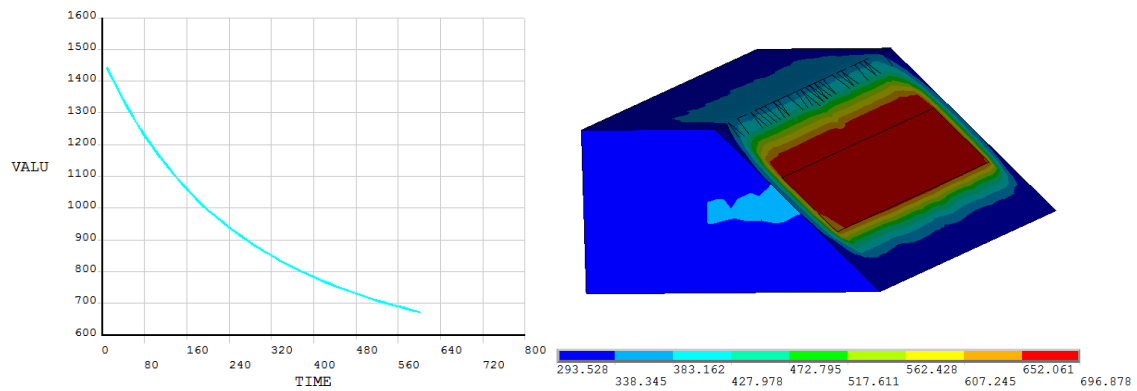


Figure 5.4 – The result of the problem solution in graphical form

5.2.2 Implementation of the problem using the program's graphical interface

1 Load the product launcher: **Start**→**All programs**→**ANSYS 2019R1**→**Mechanical APDL Product Launcher 2019R1**. In the window that opens, in the **File Management** tab, specify the working folder and the current file name. In the **Working Directory** line, specify the path to the folder in which all data and databases will be saved when performing calculations. In the **Job** name line, enter the default file name for files created by the program during calculations, and run the program using the Run button.

2 Set the task title: **Utility Menu**→**File**→**Change Title**. Enter the title name that will be displayed on the main screen (for example, Ingot 3D), and click **OK**.

3 The problem under consideration is heat distribution. To ensure that only items related to this sort of calculation appear in the **Main Menu**, you must activate filters **Main Menu**→**Preferences**→select **Thermal**→**OK**.

4 Select the element type for applying the finite element mesh and solving the heat transfer problem. We use the 8-node six-sided **SOLID70** element: **Main Menu**→**Preprocessor**→**Element Type**→**Add/Edit/Delete**→**Add**. Select **Thermal Mass**→**Solid**→**Brick 8 node 70 from the list**→click **OK**→close the **Close** window.

5 The properties of the materials must be set. For sand, thermal conductivity is 0.35 W/(m·K); density is 1600 kg/m³; specific heat capacity is 850 J/(kg·K).

Main Menu→**Preprocessor**→**Material Props**→**Material Models**. In the **Material Models Available** window, select **Thermal**→**Density**→enter 1600 in the **DENS** field→click **OK**. Then, select **Specific Heat**→enter 850 in the **C** field→click **OK**. After that, **Conductivity**→**Isotropic**→enter 0.35 in the **KXX** field→click **OK**→**Close**.

6 If the tasks use a composite model of different materials, the properties of each must be specified as an individual material model with a unique number. When you first enter properties, **ANSYS** creates a material model with number 1 by default. To specify the properties of steel, you must create a material model with a new number and define properties for it.

Main Menu→Preprocessor→Material Props→Material Models. In the **Define Material Models Behavior** window, select **Material→New Model→** enter material number 2 in the **Define Material ID field→**click **OK**.

7 Thermal conductivity and specific heat capacity of the ingot material (steel) depend on temperature and are specified in the form of Table 5.1.

Table 5.1 – Temperature dependence of steel’s thermal conductivity and specific heat capacity

Temperature, K	273	780	1300	1480
Thermal conductivity	15	18	31	34
Specific heat capacity	450	550	620	640

To do this in **ANSYS**, you need to perform the following steps:

Main Menu→Preprocessor→Material Props→Material Models.

In the **Material Models Defined** window, select **Material Model Number 2**. Then, in the **Material Models Available** window, select **Thermal→Conductivity→Isotropic**. In the **Conductivity for Material Number 2** window, click **Add Temperature** three times. Fill in Table 2.2. Click **OK**.

Table 5.2 – Table of thermal conductivity dependence on temperature in ANSYS

Temperatures	T1	T1	T3	T4
	273	780	1300	1480
KXX	15	18	31	34

Without closing the **Define Material Model Behavior** window, select **Thermal→Specific Heat**. In the **Specific Heat for Material Number 2** window, click **Add Temperature** three times. Fill in Table 5.3. Click **OK**.

Table 5.3 – Temperature-Dependent Specific Heat Capacity in ANSYS

Temperature	T1	T1	T3	T4
	273	780	1300	1480
C	450	550	620	640

8 The material properties specified in the table can be displayed as a graph. To do this, select **Main Menu→Preprocessor→Material Props→Material Models**. In the **Material Models Defined** window, select **Material Model Number 2**. Then, in the **Material Models Available** window, select **Thermal→Conductivity→Isotropic**. In the **Conductivity for Material Number 2** window, click **Graph**.

Similarly, plot a graph for specific heat capacity.

9 The next step in solving the problem is to build a solid model. Let us consider one of the possible options for building a complex composite model. To build the contour of the sand mold, it is necessary to build the vertices of a rectangular trapezoid, which is one of the ends of the mold. **Main Menu→Preprocessor→Modeling→Create→Keypoints→In Active CS**. In the **Keypoint** number line, enter the point number 1, and in the **Location** in active CS line, enter the coordinates of the point X, Y, Z, respectively 0, 0, 0. Click **Apply**. Similarly, enter three more points (table 5.4). After entering the coordinates of the last point, click **OK**.

Table 5.4 – Coordinates of the vertices of a rectangular trapezoid

Point number	X, m	Y, m	Z, m
2	0.22	0	0
3	0.1	0.12	0
4	0	0.12	0

10 To make the procedure for further construction of the model more convenient, we will make several adjustments:

- enable display of point, surface and volume numbers on the screen: **Utility Menu→PlotCtrls→Numbering**. Check the boxes **Keypoint** numbers, **Area** numbers, **Volume** numbers. Click **OK**;

- move the origin symbol to the upper left corner of the graphic window so that it does not overlap the model. **Utility Menu→PlotCtrls→Window Controls→Window Options**. In the **Location of triad** line, select **At top left** and click **OK**.

11 Create the end surface of the sand mold by points: **Main Menu→Preprocessor→Modeling→Create→Areas→Arbitrary→Through KPs**. Mark the points from the first to the fourth in sequence with the mouse pointer. Click **OK**.

12 Extrude the constructed surface in the direction of the Z axis by 0.58 m (the length of the ingot is 0.5 m + the thickness of the sand walls is 0.04 m on each side): **Main Menu→Preprocessor→Modeling→Operate→**

Extrude→**Are-as**→**By XYZ Offset**→left-click on the surface A1→press **OK**. In the Offsets for extrusion line, enter the value 0.58 in the DZ field→press **OK**.

13 Save the intermediate database file: **Utility Menu**→**File**→**Save as**→select the folder in which the database is saved and specify the file name in Latin letters→click **OK**.

14 Initially, we create the ingot model as a rectangular parallelepiped: **Main Menu**→**Preprocessor**→**Modeling**→**Create**→**Volumes**→**Block**→**By Dimensions**. In the **Create Block by Dimensions** window, enter the coordinates of the extreme points along the X, Y, Z axes (table 5.5). Click **OK**.

Table 5.5 – Table with coordinates of extreme points of a parallelepiped in ANSYS

X-coordinate	0.04	0.22
Y-coordinate	0.04	0.08
Z-coordinate	0.04	0.54

15 The part protruding beyond the plane of symmetry must be removed. To do this, perform the procedure “Overlap” of areas (overlap). This procedure divides the existing areas into parts at their intersections: **Main Menu**→**Preprocessor**→**Modeling**→**Operate**→**Booleans**→**Overlap**→**Volumes**. Click **Pick All**.

Delete the extra area: **Main Menu**→**Preprocessor**→**Modeling**→**Delete**→**Volume and Below**. Mark the **V4** volume with the mouse and click **OK**.

16 Apply the finite element mesh. In this task, we will divide the structure into elements with a given element side size. To do this, you need to define the default element side size along the edges of the model’s component parts in ANSYS: **Main Menu**→**Preprocessor**→**Meshing**→**Size Cntrls**→**Manual Size**→**Global**→**Size**. In the **Size** line, mark the default element size value of 0.015, click **OK**.

We divide the area occupied by sand into elements: **Main Menu**→**Preprocessor**→**Meshing**→**Mesh**→**Volumes**→**Free**→mark the outer part of the model (sand) **V5**→press **OK**.

When applying a finite element mesh, ANSYS will, by default, assign the properties of material No. 1 (in this problem, sand) to the elements of the meshed area.

17 In order for the internal area to be assigned the properties of material No. 2 (steel), it is necessary to indicate to the program that all further actions on applying the mesh will be performed for material No. 2:

Main Menu→Preprocessor→Meshing→Mesh Attributes→Default Attribs. In the window that opens, in the **Material Number** line, select material No. 2 and click **OK**.

Display volume areas on the screen. **Utility Menu→Plot→Volumes.** Apply a finite element mesh to the area occupied by steel: **Main Menu→Preprocessor→Meshing→Mesh→Volumes→Free→**mark the interior of the model (steel) **V3→**click **OK**.

18 Despite the fact that the model in this problem consists of materials with different properties, graphically in ANSYS the finite element mesh is displayed in one color by default. For comfortable work, you can display areas with different thermophysical properties in different colors: **Utility Menu→PlotCtrls→Numbering→in the Elem/Attrib** numbering line, select **Material** numbers, and in the **Numbering** shown with line, select **Colors only→**click **OK**.

19 Save the intermediate database file: **Utility Menu→File→Save as→**select the folder in which the database is saved and specify the file name in Latin letters→click **OK**.

20 We proceed to the next stage of solving the problem – the stage of applying loads and obtaining a solution. In this problem, it is necessary to obtain a solution on the temperature distribution during cooling of the ingot in dynamics (non-stationary analysis). The type must be set as transient: **Main Menu→Solution→Analysis Type→New Analysis→select Transient →**press **OK→**select **Full →**press **OK**.

21 Determine the initial temperature of the ingot. The initial temperature of the ingot is specified in the nodes belonging to the ingot, for which they are preliminarily selected with the **Select** command.

It is necessary to select the elements belonging to steel (material No. 2): **Utility Menu→Select→Entities.** In the first line, select **Elements**, in the second – **By Attributes**. Make sure that the **Material** num field is checked, in the **Min,Max,Inc** line, enter the material number, i.e., the number 2→click **Apply**. Select the nodes that belong to the elements just selected. In the first line, select **Nodes**, in the second – **Attached to**. Make sure that the **Elements** field is checked→click **OK**. Display the selected nodes on the screen: **Utility Menu→Plot→Nodes**. Set the initial temperature of the steel in kelvins for the selected nodes: **Main Menu→Solution→Define Loads→Apply→Initial Condit'n→Define**. In the window that appears, click **Pick All** (in all active selected nodes), In the **DOF** to be specified line, select **Temp**, in the Initial value of DOF line, enter 1480 → click **OK**.

22 Set the initial temperature of the sand mold. First, select the nodes belonging to sand (material No. 1) using the invert operation:

Utility Menu→Select→Entities. In the first line, select **Nodes**→click the **Invert** button→click **Cancel**. Refresh the working window to display the inverted nodes: **Utility Menu→Plot→Replot**.

Set the initial sand temperature in Kelvin for the selected nodes: **Main Menu→Solution→Define Loads→Apply→Initial Condit'n→Define**. In the window that appears, click Pick All (in all active selected nodes). In the **DOF** to be specified line, select **Temp**, in the Initial value of **DOF** line, enter 293 and click **OK**.

23 Set the heat exchange conditions at the sand – air boundaries. To do this, activate all model nodes **Utility Menu→Select→ Everything**. Display volume areas on the screen **Utility Menu→Plot→ Volumes**. Next **Main Menu→Solution→Define Loads→Apply→ Thermal→ Convection→On Areas**. In the Apply **CONV** on Areas window, in the line, enter the area numbers: **1,2,3,5,6** (all external surfaces of the model, except for surfaces belonging to the plane of symmetry)→click **OK**. In the **Film** coefficient line, enter **0.4** (heat transfer coefficient $W/(m^2 \cdot K)$), in the Bulk temperature line, enter **293** (air temperature in kelvins)→click **OK**.

24 Since the problem is non-stationary, it is necessary to set the time parameters of the solution.

Call the time parameters settings window: **Main Menu→Solution→Load Step Opts→Time/Frequency→Time-Time Step**. In the **Time and time step options** window that opens, do the following:

- in the **Time at end of load step** line, enter 600 (the cooling process observation time in seconds);
- in the **Time step size** line, enter 10 (the recommended step size for solving the problem);
- in the **Stepped or ramped b.c.** line, select **Stepped** (the boundary conditions are constant over time);
- in the **Automatic time stepping** line, select **ON** (an automatic variable time step for solving the problem is set);
- in the **Minimum step size** line, enter 5 (the minimum time step for solving the problem);
- in the **Maximum step size** line, enter 15 (the maximum time step for solving the problem);
- click **OK**.

25 To enable the program to retain the results for each stage of the solution, which can then be shown as a graph or table, it is essential to execute: **Main Menu→Solution→Load Step Opts→Output Ctrl's→DB/Results File**. In the **File** write frequency field, select **Every substep**→click **OK**.

26 Save the intermediate database file: **Utility Menu**→**File**→**Save as**→select the folder in which the database is saved and specify the file name in Latin letters→click **OK**.

27 Launch the task for solution: **Main Menu**→**Solution**→**Solve**→**CurrentLS**→click **OK**. Wait for completion (the message **Solution is done!** will appear).

28 The final stage of solving the problem is post-processing of the results. We will graphically display the dependence of temperature on time in a given node.

Enable node numbering on the screen: **Utility Menu**→**PlotCtrls** →**Numbering**. In the **Node numbers** line, mark **ON**, in the **Numbering shown with** line, select **Colors&Numbers**→click **OK**. Display elements in the working window: **Utility Menu**→**Plot**→**Elements**.

As can be seen, when the finite element mesh is dense enough, it can be very difficult to visually determine the number of the required node. Therefore, we will use another method.

Select the node with the center coordinates ($x=0.16$ m, $y=0.06$ m, $z=0.29$ m), approximately corresponding to the center of the plane of symmetry. To do this, assign the variable `cnt_pt` the number of the node with these coordinates: **Utility Menu**→**Parameters**→**Scalar Parameters**. In the Selection line, type `cnt_pt=node(0.16,0.06,0.29)`→click **Accept**→click **Close**.

29 Let us present a graph illustrating the temporal dependency of temperature at this node: **Main Menu**→**TimeHist Postpro**→**Define Variables**→in the window that opens, click **Add**→select **Nodal DOF result**→click **OK** (add a new variable). In the **Define Nodal Data** window, enter `cnt_pt` in the empty line→click **OK**. Check that the **Node number** field contains the node number→click **OK**→**Close**. Next, **Main Menu**→**TimeHist Postpro**→**Graph Variables**→enter parameter 2 in the line 1st variable to graph→Click **OK**.

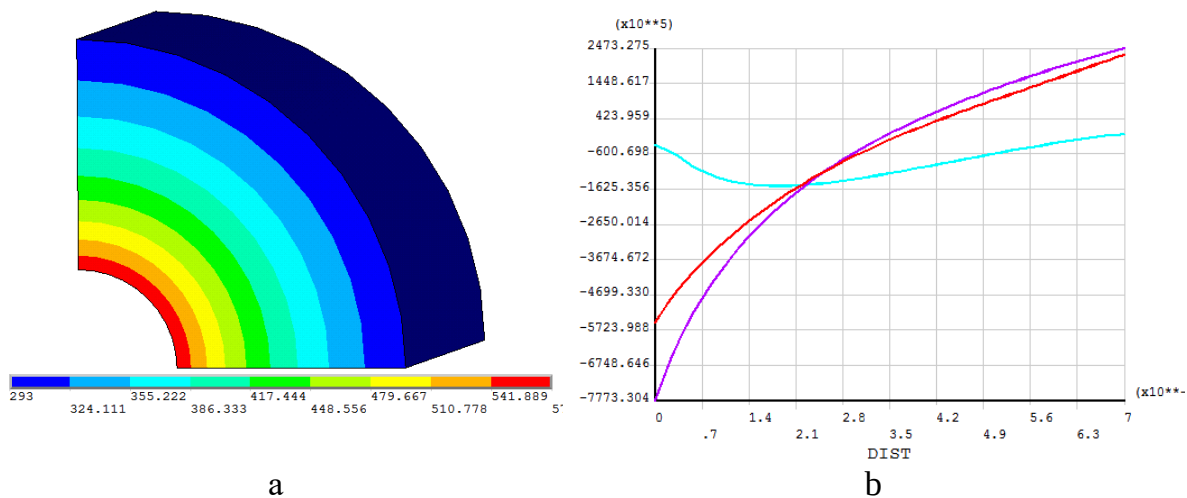
Problem 5.3 “Thermal-Stress Analysis of Thick-Walled Hollow Cylinder in 3D Formulation”

5.3.1 Problem statement and assumptions

This example enables familiarity with the methodology for solving three-dimensional stationary problems related to the computation of thermoelastic stresses induced by non-uniform heating of a sample. At the first stage, the temperature fields are computed, after which the derived

nodal temperature values are transferred to a new model utilizing load interpolation techniques, and the thermoelastic stresses are determined inside a quasi-static problem framework.

A 0.2 m long cylinder with internal and external radii of 0.03 and 0.1 m, respectively, is selected as a model. The external surface of the cylinder is maintained at a temperature of 20 °C, and the internal surface at a temperature of 300 °C. Due to symmetry, only one quarter of the cylinder is modeled. The eight-node thermal element SOLID 70 and the strength element SOLID 185 are used to solve the problem. The example is intended to study the methodology for solving problems of this type (Figure 5.5).



a – distribution of temperature fields in the cylinder;
b – Stress component distribution along designated path in cylinder

Figure 5.5 – Result of solving the problem in graphical form

5.3.2 Implementation of the problem using the program's graphical interface

1 Load the product launcher: **Start→All programs→ANSYS 2019R1→Mechanical APDL Product Launcher 2019R1**. In the window that opens, in the **File Management** tab, select the working folder and the current file name. In the **Working Directory** line, specify the path to the folder in which all data and databases will be saved when performing calculations. In the **Job** name line, enter the default file name therm for files created by the program during calculations, and run the program using the **Run** button.

2 Create a geometric model.

A cylinder is a symmetrical figure relative to the axis. In such cases, you can use not a full-size model for the calculation, but only a part of it,

specifying the symmetry conditions at the boundaries of the model section. For this analysis, we will create a quarter-cylinder geometry by navigating to **Main Menu → Preprocessor → Modeling → Create → Volumes → Cylinder → Partial Cylinder**. In the **Partial Cylinder** window, enter 0.2 for the **Depth** line (cylinder length), specify 0.03 and 0.1 for **Rad-1** and **Rad-2** lines (inner and outer radii respectively), set **Theta-1** and **Theta-2** to 0 and 90 degrees (angular sector bounds), and click **OK** to generate the geometry.

3 Save the database for subsequent use in the strength problem.

The sizes of the finite element mesh used in the thermal and strength problems are different, but the geometric sizes of the models are the same. Therefore, at this stage of solving the problem, you can save the geometric model separately as a database file (for example, **struct.db**) and import it at the stage of constructing the strength model. To save the database, do the following: **Utility Menu → File → Save as → in the Save Database to** line, enter the name **struct.db** → click **OK**.

4 Set the task title: **Utility Menu → File → Change Title**. Enter the title name that will be displayed on the main screen (for example, Thermal problem in concentric cylinder), and click **OK**.

5 Select the type of solid element used in the thermal calculation. We use the eight-node six-sided element **SOLID70**. **Main Menu → Preprocessor → Element Type → Add/Edit/Delete → Add**. Select **Thermal Mass → Solid → Brick 8 node 70** from the list → click **OK** → close the window.

6 Set the properties of the material required for the stationary thermal calculation: **Main Menu → Preprocessor → Material Props → Material Models**. We will use steel as the material. In the **Material Models Available** window, select **Thermal → Conductivity → Iso-tropic** → enter 40 in the **KXX** field → click **OK** → **Close**.

7 The next stage of solving the problem is applying a finite element mesh. In this example, we will analyze the option with preliminary division of the model contour into a given number of elements.

For convenience, the model should be presented as contour lines: **Utility Menu → Plot → Lines**. Display the contour line numbers on the screen: **Utility Menu → PlotCtrls → Numbering** → in the **Plot Numbering Controls** window, check the box next to **LINE Line numbers** → click **OK**.

8 Set the parameters of the grid division. In this example, it is suggested to set the number of divisions on the lines: **Main Menu → Preprocessor → Meshing → Size Cntrls → ManualSize → Lines → Picked Lines**. Select lines **L2, L4, L5, L7** (circle arcs) or enter line numbers **2,4,5,7** in the input line → press **OK**.

In the **Element Sizes on Picked Lines** window that appears, enter the number of elements for dividing lines 10 in the **NDIV** No. of element divisions line and click **Apply**. Repeat the operation for lines **L1, L3, L6, L8** (number of elements for dividing lines 14) and for lines **L9, L10, L11, L12** (number of elements for dividing lines 15). Click **OK**.

9 We perform the planned mesh division of the created model: **Main Menu**→**Preprocessor**→**Meshing**→**Mesh**→**Volumes**→ **Mapped**→ 4 to 6 sided→select the model with the cursor→press **OK**.

10 Set the boundary conditions for the heat transfer problem – the temperature on the outer and inner surfaces of the cylinder: **Main Menu**→**Solution**→**Define Loads**→**Apply**→**Thermal**→**Temperature**→**On Areas**→select the inner surface of the cylinder with the cursor→click **OK**. In the window that appears Apply **TEMP** on Areas in the Lab2 **DOFs** to be constrained column, select **TEMP**, in the **VALUE Load TEMP** value column, enter the temperature value 300+273→click **Apply**. Select the outer surface→click **OK**. Similarly, set the temperature of the outer surface of the cylinder 20+273→click **OK**.

11 Two flat surfaces of the cylinder, limited by contour lines L3, **L11, L6, L9** and **L1, L10, L8, L12**, are located inside the real model and do not exchange heat with the environment. Therefore, it is necessary to indicate that heat exchange does not occur on these surfaces (adiabatic condition).

First, it is necessary to select the nodes belonging to these surfaces: **Utility Menu**→**Select**→**Entities**. In the **Select Entities** window, select Nodes in the first line, select By Location in the second line; in the menu that appears, select X coordinates, and in the **Min,Max** column, enter 0. Put a dot next to **From Full (select from all available)** and click **Apply**.

Select **Y coordinates** and in the **Min,Max** column enter 0 → put a dot next to **Also Select (in addition to the selected ones)** → click **OK**.

12 Set the adiabatic condition on the selected surfaces: **Main Menu**→**Solution**→**Define Loads**→**Apply**→**Thermal**→**Heat Flow**→ **On Nodes**→**Pick All**→in the **Apply Heat on Nodes** window, in the **VALUE Load HEAT** value column, enter 0→click **OK**.

13 Activate all elements of the model before launching the solution: **Utility Menu**→**Select**→**Everything**.

14 Launch the task for solution: **Main Menu**→**Solution**→**Solve**→**CurrentLS**→press **OK**. Wait for completion (The message **Solution is done!** will appear).

Save the obtained data as a database file with a default name: **Utility Menu**→**File**→**Save as Jobname.db**

If the default name therm was given at the first stage of solving the problem, then the file will be saved under the name therm.db.

15 Display the solution results as a spatial distribution of temperature fields: **Main Menu**→**General Post-proc**→**Plot Results**→**Contour Plot**→**Nodal Solution**. In the **Contour Nodal Solution Data** window, select **Nodal solution**→**DOF Solution**→**Nodal Temperature**→click **OK**. The temperature distribution with a color legend should appear on the screen.

16 Next, we proceed to solving the strength problem. It is necessary to reset the current task and start a new one: **Utility Menu**→**File**→**Clear & Start New**→select **Do not read file**→press **OK**.

17 Set a new default file name struct: **Utility Menu**→**File**→**Change Jobname**→enter a new file name struct and click **OK**.

18 Restore the saved database containing the model template: **Utility Menu**→**File**→**Resume from...**→in the **Resume Database** window, select the struct.db file from the list of files in your folder→click **OK**.

19 Set a new task title: **Utility Menu**→**File**→**Change Title**→enter the name of the title that will be displayed on the main screen (for example, **Structural problem in concentric cylinder**)→click **OK**.

20 Select the type of solid element used in the strength calculation. We use the eight-node six-sided element **SOLID185**: **Main Menu**→**Preprocessor**→**Element Type**→**Add/Edit/Delete**→**Add**. Select **Structural Mass**→**Solid**→**Brick 8 node 185** from the list→click **OK**→close the **Close** window.

21 Set the steel properties for strength analysis: **Main Menu**→**Preprocessor**→**Material Props**→**Material Models**. In the **Material Models Available** window, select **Structural**→**Linear**→**Elastic**→**Isotropic**→enter the **Young's modulus value** of $2e^{11}$ (i.e., $2 \cdot 10^{11}$ Pa) in the **EX** field→ enter the **Poisson's ratio value** of 0.3 in the **PRXY** field→Click **OK**.

In the **Material Models Available** window, select **Structural**→**Thermal Expansion**→**Secant Coefficient**→**Isotropic**→enter the value of the coefficient of thermal expansion $1.22e^{-5}$ (i.e., $1.22 \cdot 10^{-5} \text{ K}^{-1}$) in the **ALPX** field→ enter the value of the initial material temperature 0 in the **Reference temperature** field→click **OK**→close the window.

22 Let us apply the finite element mesh. To do this, repeat steps 7–9 of this example for the following line splitting parameters:

- lines **L2, L4, L5, L7** – number of elements to be split 20;
- lines **L1, L3, L6, L8** – number of elements to be split 28;
- lines **L9, L10, L11, L12** – number of elements to be split 30.

23 The next step is interpolation of temperatures to the new model from the solution of the heat transfer problem. First, it is necessary to write down the coordinates of all nodes of the strength model: **Main Menu**→**Preprocessor**→**Modeling**→**Create**→**Nodes**→**Write Node File**→in the

Write Nodes to File window, in the **Name of file** line, enter **struct_1.node**→click **OK**. As a result, the **struct_1.node** file will be created with the coordinates of all nodes.

24 Save the strength problem database with the default file name (in this example, **struct.db**): **Utility Menu**→**File**→**Save as Jobname.db**.

25 Read the heat transfer task database: **Utility Menu**→**File**→**Resume from...**→in the **Resume Database** window, select the **therm.db** file from the list of files in your folder→click **OK**.

26 Perform the data identification operation for recovery from the results file: **Main Menu**→**General Post-proc**→**Data & File Opts...**→in the **Data to be read** window, select **All items**→in the Results file to be read field, specify **therm.rth** as the source file for the results→click **OK**.

27 Interpolate the load: **Main Menu**→**General Post-proc**→**Submodeling**→**Interp Body Forc**→in the **File containing nodes** window, enter the name of the previously created node file **struct_1.node** or select it using the **Browse** button→press **OK**.

After completing this step, a **struct.bfin** file containing the interpolated load will appear in the working directory.

28 We set the temperatures in the nodes of the strength model from the results of solving the heat transfer problem. It is necessary to go to the **Solution** section. To do this, it is necessary to perform the following actions: **Main Menu**→**Solution**. This operation is necessary, since the commands in the **APDL** programming language contained in the **struct.bfin** file that is subsequently read are executed only when the **Solution** section is active.

Load the database with the saved model for the strength problem: **Utility Menu**→**File**→**Resume from...**→in the **Resume Data-base** window, select the **struct.db** file from the list of files in your folder→click **OK**.

Read data from the file of interpolated load on model nodes from the **struct.db** file: **Utility Menu**→**File**→**Read Input from...**→in the **Read File** window, in the **Read input from** section, select the **struct.bfin** file→click **OK**.

As a result, the nodal temperature values obtained in the thermal analysis will be interpolated onto the mesh of the strength problem model.

29 Two flat surfaces of the cylinder, limited by contour lines **L3, L11, L6, L9** and **L1, L10, L8, L12**, are located inside the real model. The program must be informed that the flat rectangular surfaces are not the boundaries of the model, but are located inside it.

For ease of working with the model, the model should be presented as a wireframe (contour lines): **Utility Menu**→**Plot**→**Lines**. **At the end**

boundaries, set the symmetry condition: **Main Menu**→**Solution**→**Define Loads**→ **Application**→**Structural**→**Displacement**→**Symmetry B.C.**→**On Areas**→ select with the cursor the flat rectangular surfaces limited by the contour lines **L3, L11, L6, L9** and **L1, L10, L8, L12**, and click **OK**.

30 Before starting the solution, you need to set the solver parameters: **Main Menu**→**Solution**→**Analysis Type**→**Sol'n Controls**→in the **Basic** section, in the **Analysis Options** line, select **Large Displacement Static**→click **OK**. In the **Sol'n Options** section, in the **Equation Solver** block, select the **Pre-Condition CG (Preconditioned Conjugate Gradient (PCG) Solver)** solver type and move the slider to the far left position.

When selecting the standard solver, errors may occur when running the problem. You can also use other types of solvers. For example, good results were obtained using the **Jacobi Conjugate Gradient (JCG) Solver**. When solving thermal problems, you can select it using the following sequence of actions: **Main Menu**→**Solution**→**Analysis Type**→**Analysis Options**→in the **EQSLV Equation solver** line, select **Jacobi Conj Gradnt**→click **OK**. When solving thermoelasticity problems, this item is unavailable in the **Main Menu** window. However, you can use the **APDL** programming language to enter the following command in the command line: **EQSLV,JCG**.

31 Activate all elements of the model before launching the solution: **Utility Menu**→**Select**→**Everything**.

32 Launch the task for solution: **Main Menu**→**Solution**→**Solve**→**CurrentLS**→press **OK**. Wait for completion (the message **Solution is done!** will appear).

33 Read the results of the solution of the last step of the solution: **Main Menu**→**General Postproc**→**Read Results**→**Last Set**.

In earlier versions of **ANSYS**, solution results may be available for display immediately after the calculations are completed.

34 Save the received data as a database file with the default name: **Utility Menu**→**File**→**Save as Jobname.db**.

35 The results of the calculation of thermoelastic stresses are presented in the form of a graph of stress values along a given path.

To accomplish this, you should switch to the cylindrical coordinate system to view the results: **Main Menu**→**General Postproc**→**Options for Outp**→to view the results in the cylindrical coordinate system in the **Options for Output** window in the **Results coord system** line, select **Global cylindric**→click **OK**.

Next, you need to visualize the nodes belonging to the section in the middle of the model: **Utility Menu**→**Plot**→**Nodes**. After that, activate the necessary model nodes: **Utility Menu**→**Select**→**Entities**→in the **Select Entities** window, in the first line, select **Nodes**→in the second line, select **By Location**→in the menu that appears, select Z coordinates, and in the **Min,Max** column, enter the range of values 0.095,0.1→put a dot opposite **From Full (select from all available)** →click **OK**.

36 Let us define the trajectory along which we will plot the voltage dependence graph: **Main Menu**→**General Postproc**→**Path Operations**→**Define Path**→**By Nodes**.

Use the cursor to sequentially select several nodes lying on any radial line, starting with the node lying on the inner circle and ending with the node lying on the outer circle → click **OK** → in the **By Nodes** window that appears, in the **Define Path Name** line, enter the name of the specified path radial → click **OK** → close the window with information about the selected path.

37 Select the stress components of interest to plot a graph in a cylindrical coordinate system: **Main Menu**→**General Postproc**→**Path Operations**→**Map onto Path**→in the **Map Result Items onto Path** window, in the **Item to be mapped** section, select **Stress**→**X-Direction SX**→click **Apply**→**Y-Direction SY**→click **Apply**→**Z-Direction SZ**→click **OK**:

- SX is the radial component;
- SZ is the axial component along Z-axis;
- SY is the tangential component.

38 Display the graphs on the screen: **Main Menu**→**General Post-proc**→**Plot Results**→**Plot Path Item**→**On Graph**→in the **Plot of Path Items on Graph** window, in the **Path items to be graphed** section, select the values of interest **SX, SY, SZ**→click **OK**.

Problem 5.4 “Transient Thermal Analysis with Tabular Boundary Conditions in 3D Formulation”

5.4.1 Problem statement and assumptions

This example demonstrates the solution methodology for three-dimensional transient thermal problems involving convective heat transfer at material interfaces, with temperature-dependent heat transfer coefficients (Figure 5.6).

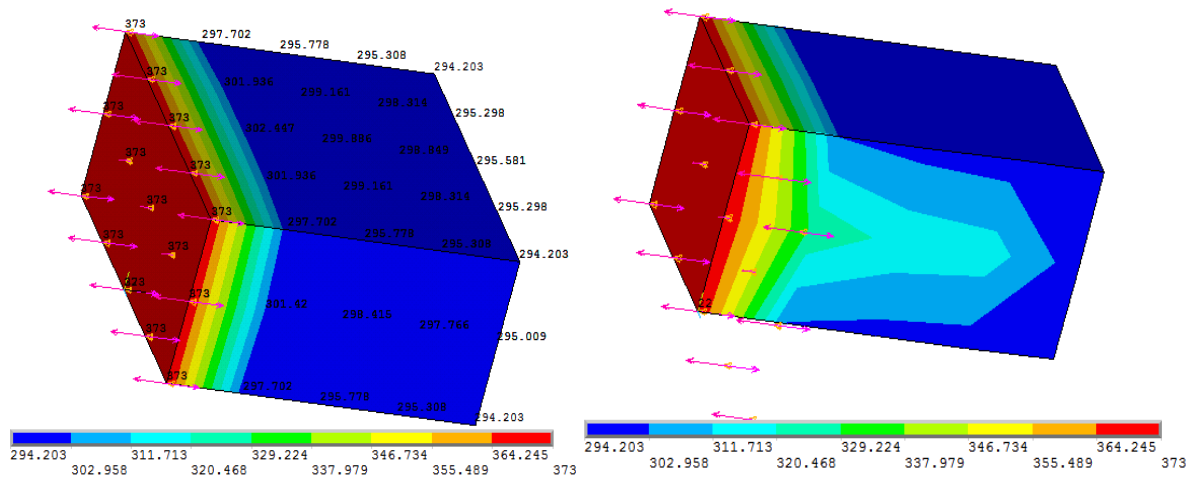


Figure 5.6 – Temperature distribution in the plate as a result of solving the problem

A rectangular parallelepiped with a side ratio of 2:1:2 has a constant temperature on one side, while convective heat exchange occurs on the others. In this case, the convective heat transfer coefficient (film coefficient) is a function of temperature, and this parametric dependence is specified in a table (table name – cnvtab).

5.4.2 Implementation of the problem using the program's graphical interface

1 Load the product launcher: **Start**→**All programs**→**ANSYS 2019 R1**→**Mechanical APDL Product Launcher 2019 R1**. In the window that opens, in the **File Management** tab, specify the working folder and the current file name. In the **Working Directory** line, specify the path to the folder in which all data and databases will be saved when performing calculations. In the **Job** name line, enter the default file name (for example, **Tab**) for files created by the program during calculations, and run the program using the **Run** button.

2 In ANSYS, multiple methods exist to ascertain the relationship between material properties and boundary conditions with temperature. One of them was addressed in task 2. Let us explore an alternative approach. In this instance, we will construct a table to input the values of the heat transfer coefficient from the material's surface at various temperatures, which will be utilized during the determination of loads and boundary conditions.

Perform the following sequence: **Utility Menu**→**Parameters**→**Array Parameters**→**Define/Edit**→in the **Array Parameters** window that

opens, click **Add**→in the **Add New Array Parameter** window, in the **Parameter name** line, enter the table name **cnvtab**→in the **Parameter type** line, specify the **Table data** type→in the **No. Of rows,cols,planes** line, enter the number of **rows, columns, sheets** equal to **3,1,1** respectively→in the **Row Variable** line, enter **TEMP** (in this case, another column will be added to the table, in which the entered values will have the dimension of temperature)→click **OK**.

Fill the table with data. In the **Array Parameters** window that remains on the screen, click the **Edit** button or, if the window is closed, do the following: **Utility Menu**→**Parameters**→**Array Parameters**→**Define/Edit**→**Edit**. Enter the following values in the first column instead of the highlighted numbers: **20+273, 50+273** and **100+273** (to do this, click the gray buttons with numbers). Enter the heat transfer coefficient values **10, 20, 30** in the white input fields of the second column in lines **1, 2, 3**, respectively. Note that you do not need to enter anything in the field above the second column, since this is a two-dimensional table. Select **File**→**Apply\Quit**→in the **Array Parameters** window, click **Close**.

3 Select the type of solid element used in the thermal calculation. In this example, we will use the eight-node six-sided element **SOLID70**: **Main Menu**→**Preprocessor**→**Element Type**→**Add/Edit/Delete**→**Add**. Select **Thermal Mass**→**Solid**→ **Brick 8 node 70** from the list→click **OK**→close the window.

4 Set the material properties: density, specific heat capacity and thermal conductivity of the material. For this, follow the sequence: **Main Menu**→**Preprocessor**→**Material Props**→ **Material Models**. In the **Material Models Available** window, select→ **Thermal**→**Density**→enter **10** in the **DENS** line→click **OK**. Next, select **Specific Heat**→enter **100** in the **C** line→click **OK**. Then, **Conductivity**→**Isotropic**→enter **1** in the **KXX** line→click **OK**→**Close**.

5 Create a geometric model of a parallelepiped using the coordinates of its sides: **Main Menu**→**Preprocessor**→**Modeling**→**Create**→**Volumes**→ **Block**→**By Dimensions**→in the **Create Block by Dimensions** window, enter the coordinates of the parallelepiped's boundaries (Table 5.6). Click **OK**.

Table 5.6 – Coordinates of the extreme points of a parallelepiped in ANSYS

X1,X2 X-coordinates	0	2
Y1,Y2 Y-coordinates	0	1
Z1,Z2 Z-coordinates	0	2

6 Apply the finite element mesh. This task will consider the option of dividing the model using the **MeshTool** utility.

First, you need to set the default element size when breaking: **Main Menu**→**Preprocessor**→**Meshing**→**Size Cntrls**→**SmartSize**→**Adv Opts**→ in the **Advanced SmartSize Setting-tings** window, in the **Global element size** line, enter the value **0.25**→click **OK**.

Then call the **MeshTool** utility: **Main Menu**→**Preprocessor**→**Meshing**→ **MeshTool**. In the **MeshTool** window that opens, in the **Mesh** section, select **Volumes**, set the partition type **Shape: Hex and Mapped**→click **MESH**. In the **Mesh Volumes** selection menu that appears,click **Pick All**.

7 We proceed to the stage of applying loads.

To set a constant temperature on one side of the model, you need to perform the following sequence: **Main Menu**→**Solution**→**Define Loads**→**Apply**→**Thermal**→**Temperature**→**On Areas**→ select a vertical surface with coordinates **X=0** in the graphic window→click **OK**→in the **Apply TEMP on Areas** window, in the **Load TEMP value** line, enter the value **100+273**, and in the **DOFs to be constrained** line, choose **TEMP**→click **OK**.

The heat transfer coefficient values for considering heat exchange with the environment on the other sides of the model must be sourced from the cnvtab table: **Main Menu**→**Solution**→**Define Loads**→**Apply**→**Thermal**→ **Convection**→**On Areas**→select with the mouse the five remaining boundaries of the model that exchange heat with the environment→click **OK**→in the **Apply CONV on areas** window, in the **Apply Film Coef on areas** line, select **Existing table**→in the **VAL2I Bulk temperature** line, enter the value of the constant ambient temperature **20+273** and delete everything in the **VALI Film coefficient** field→click **OK**. Then, the **Apply CONV on areas** window will appear, where you must select the **CNVTAB table** for the “Existing table”. Click **OK**.

Set the initial temperature of the material: **Main Menu**→**Solution**→**Define Loads**→**Apply**→**Thermal**→ **Tempera-ture**→**Uniform Temp**→ in the window that appears, in the **Uniform temperature** line, enter the value **50+273**→click **OK**.

8 When preparing a problem for solution, it is convenient to have the applied loads and specified boundary conditions displayed graphically on the screen.

This can be achieved by the following actions:

- display the surfaces limiting the model on the working field: **Utility Menu**→**Plot**→**Areas**;

- display convective heat transfer as arrows: **Utility Menu**→**PlotCtrls**→**Symbols**→in the **Symbols** window, in the **Surface Load Symbols** line, select **Convect FilmCoef**, and in the **Show pres and convect** as line, select **Arrows** (i.e., display as arrows)→click **OK**;

- display the name of the table from which the heat transfer coefficient values were taken above the arrows in the graphics window: **Utility Menu**→ **PlotCtrls**→**Numbering**→in the **Plot Numbering Controls** window, in the **Table Names** line, check the on box→click **OK**.

9 This problem requires the determination of the solution for the temperature distribution in dynamics (non-stationary analysis). Let's set the analysis type - transient: **Main Menu**→**Solution**→**Analysis Type**→**New Analysis**→select **Transient**→press **OK**→select **Full**→press **OK**.

10 Set the time parameters of the solution for the non-stationary problem: **Main Menu**→**Solution**→**Load Step Opts**→**Time/Frequency**→**Time-Time Step**. In the **Time and Time Step Options** window that opens, do the following:

- in the **Time at end of load step** line, enter **60** (process observation time in seconds);

- in the **Time step size** line, enter **1** (the recommended step size for solving the problem);

- in the **Stepped or ramped b.c.** line, select the **Stepped** option (the boundary conditions are constant over time);

- click **OK**.

11 To enable the graphical or tabular display of solution results for any point in time, the program is required to save the results upon the completion of each solution step. To do this, navigate to: **Main Menu**→**Solution**→**Load step opts**→**Output Ctrls**→**DB/Results File**. In the **File write frequency** field, select **Every substep** and click **OK**.

12 Activate all elements of the model before launching the solution: **Utility Menu**→**Select**→**Everything**.

13 Save the intermediate database file: **Utility Menu**→ **File**→**Save as**→select the folder in which the database is saved and specify the file name in Latin letters→click **OK**.

14 Launch the task for solution: **Main Menu**→**Solution**→**Solve**→**CurrentLS**→press **OK**. Wait for the calculation to complete until the message "**Solution is done!**" appears.

15 Display the calculation results on the screen.

To display the temperature distribution, it is essential to first load the results from the pertinent phase, as the program has preserved the outcomes of each step during the solution process. By default, upon completion of the calculations, the results for the final step of the solution are loaded.

Main Menu→General Postproc→Read Results→By Time/Freq→ in the **Read Results by Time of Frequency** window, in the **Value of time of freq** line, enter value **50** (i.e., load the solution results for the fiftieth second)→click **OK**.

16 Display the solution results as a spatial distribution of temperature fields: **Main Menu→General Postproc→Plot Results→Contour Plot→Nodal solu**. In the **Contour Nodal Solution Data** window, select **Nodal Solution→DOF Solution→Nodal Temperature**→click **OK**.

The temperature distribution with a color legend should appear on the screen.

17 Display the numerical values of temperatures and loads on the model in the graphic window. For this, perform the following steps: **Utility Menu→PlotCtrls→Numbering**→in the **Plot Numbering Controls** window, in the **Table Names** line, mark **Off**, and in the **Numeric contour values** line, mark **On**→click **OK**.

18 Let us show convective heat transfer on the model in the form of arrows indicating the numerical values of the heat transfer coefficient for a given solution step: **Utility Menu→PlotCtrls→Symbols**→in the **Symbols** window, in the **Surface Load Symbols** line, select **Convect FilmCoef**, and in the **Show pres and convect as** line, select **Arrows** (i.e. **display as arrows**)→click **OK**.

In earlier versions of ANSYS, the temperature distribution and heat transfer coefficient are displayed simultaneously on the model. In ANSYS 2016 R1 Academic, to display the heat transfer coefficient distribution on the model surface, display the **Utility Menu→Plot→Elements** elements, and then repeat the steps described in this section above.

19 To display the distribution of temperature fields inside the model, it is necessary to set the local coordinate system to a given point and orient the Z-axis in the direction required to hide part of the model.

Utility Menu→WorkPlane→Local Coordinate Systems→Create Local CS→At Specified Loc +→left-click anywhere→click **OK**. In the **Create Local CS at Specified Location** window, in the **Ref number of new coord sys** line, enter the local coordinate system number **22**→in the **Origin of coord system** line, enter the **X, Y, Z** coordinates of the origin of the local coordinate system relative to the global system **0,0,1**→click **OK**.

Next, you need to tie the work plane to the local coordinate system: **Utility Menu→WorkPlane→Align WP with→Active Coord Sys**.

It remains to hide part of the model in the direction of the Z-axis from the origin of the local coordinate system: **Utility Menu→PlotCtrls→Style→Hidden Line Options**→in the **Hidden-Line Options** window, in the **Type of Plot** line, select **Capped Hidden**→in the **Cutting plane is** line, select **Working plane**→click **OK**.

Problem 5.5 “Steady-state thermal analysis of temperature field at pipe intersection”

5.5.1 Problem statement and assumptions

This example provides an opportunity to understand the processes involved in constructing a complex geometric model, implementing a finite element mesh, and addressing stationary heat transfer issues while considering the temperature-dependent thermophysical properties of the material.

A small pipe is positioned within the cylindrical tank, extending in the radial direction and maintaining an equal distance from both ends of the tank. The interior of the tank is subjected to a washing process utilizing liquid maintained at a temperature of 232 °C. A stationary flow of liquid at a temperature of 38 °C is present in the small pipe. Both flows are separated from one another by a thin pipe.

The heat transfer coefficient inside the tank is constant and equal to 1420 W/(m² K). The heat transfer coefficient in the small pipe depends on the temperature of the metal, and this dependence, as well as the dependence of the thermal properties of the pipe and tank material, are given below in Table 5.7.

Table 5.7 – Properties of materials

Temperature	294	366	422	478	533	K
Density	7 889	7 889	7 889	7 889	7 889	kg/m ³
Thermal conductivity	14.45	15.4	16.18	16.96	17.71	J/(s·m·°C)
Specific heat capacity	0.473	0.490	0.498	0.511	0.523	J/(kg·°C)
Heat transfer coefficient	2 419	2 300	1 999	1 562	1 255	W/(m ² ·K)

Since the solution to the problem uses thermal-physical properties of materials that depend on temperature, substeps and the choice of a time step will be required (Figure 5.7).

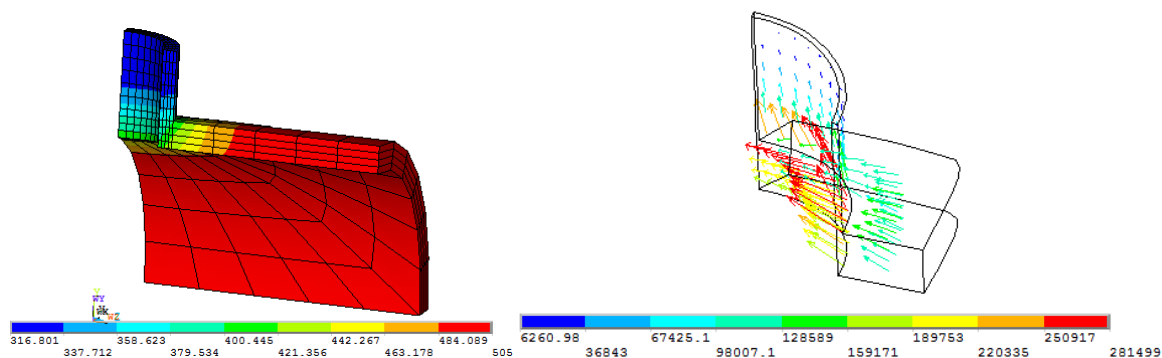


Figure 5.7 – Temperature field and vector field of heat flux densities at the intersection of two pipes

5.5.2 Implementation of the problem using the program's graphical interface

1 Load the product launcher: **Start** → **All Programs** → **ANSYS 2019 R1** → **Mechanical APDL Product Launcher 2019 R1**. In the window that opens, in the **File Management** tab, specify the working folder and the current file name. In the **Working Directory** line, specify the path to the folder in which all data and databases will be saved when performing calculations. In the **Job** name line, enter the default file name (for example, **Pipe**) for files created by the program during calculations, and run the program using the **Run** button.

2 Set the problem title: **Utility Menu**→**File**→**Change Title**. Enter the title name that will be displayed on the main screen (for example, **Steady-state thermal analysis of pipe junction**), and click **OK**.

3 Determine the element type for applying the finite element mesh: **Main Menu**→**Preprocessor**→**Element Type**→**Add/Edit/Delete**→ **Add**. In the window that opens, select **Thermal Mass**→**Solid**→**Brick 20 node 90** from the list→click **OK**→close the window.

4 Set the properties of the pipe and tank materials.

Set the material density independent of temperature: **Main Menu**→**Preprocessor**→**Material Props**→**Material Models**. In the **Material Models Available** window, select **Thermal**→**Density**→enter the value **7889** in the **DENS** field→click **OK**.

Specify the thermal conductivity coefficient and specific heat capacity as temperature dependent: **Main Menu**→**Preprocessor**→**Material Props**→**Material Models**. In the **Material Models Available** window, select **Thermal**→**Conductivity**→**Isotropic**. In the **Conductivity for Material Number 1** window, click **Add temperature** 4 times. Fill in Table 5.8 and click **OK**.

Table 5.8 – Thermal Conductivity Dependence on Temperature in ANSYS

Temperature	T1	T1	T3	T4	T5
	294	366	422	478	533
KXX	14.45	15.4	16.18	16.96	17.71

Without closing the **Define Material Model Behavior** window, select **Thermal**→ **Specific Heat (specific heat capacity)**. In the **Specific Heat for Material Number 1** window, click **Add temperature** 4 times. Fill in Table 5.9 and click **OK**.

Table 5.9 – Dependence of specific heat capacity on temperature in ANSYS

Temperature	T1	T1	T3	T4	T5
	294	366	422	478	533
C	0.473	0.490	0.498	0.511	0.523

5 We define the dependence of the heat transfer coefficient of a small pipe as a property of material 2.

First, you need to create material 2: **Main Menu**→**Preprocessor**→**Material Props**→**Material Models**. In the **Define Material Model Behavior** window, select **Material**→**New Model**→enter **material number 2**→click **OK**.

Next, without closing the **Define Material Model Behavior** window, select **Thermal**→**Convection or Film Coef**. In the **Convection or Film Coefficient for Material Number 2** window, click **Add temperature 4** times. Fill in Table 5.10 and click **OK**.

Table 5.10 – Temperature-dependent heat transfer coefficient in ANSYS

Temperature	T1	T1	T3	T4	T5
	294	366	422	478	533
HF	2 419	2 300	1 999	1 562	1 255

Close the material properties window

6 When constructing a geometric model, you can use pre-defined scalar parameters to specify dimensions: **Utility Menu**→**Parameters**→**Scalar Parameters**. In the **Scalar Parameters** window, enter the parameters in the **Selection** line. After entering each value, click “**Accept**”:

- **RI1=0.03302** (inner radius of cylindrical tank);
- **RO1=0.0381** (outer radius of cylindrical tank);
- **Z1=0.0508** (length of cylindrical tank);
- **RI2=0.01016** (inner radius of pipe);
- **RO2=0.0127** (outer radius of pipe);
- **Z2=0.0508** (length of pipe).

Finally, click **Close**.

Create a cylindrical tank. In the case of radial intersection of pipes, several planes of symmetry can be identified passing along their axes. Using this in heat transfer problems for analysis, it is possible to use not a full-size model, but its part, cut out by planes of symmetry.

To create a quarter cylinder, perform the following sequence of actions: **Main Menu**→**Preprocessor**→**Modeling**→**Create**→**Volumes**→**Cylinder**→**By Dimensions**. In the **Create Cylinder by Dimensions** window, enter the following scalar parameters in the lines:

- in the **Outer radius** line – **RO1**;
- in the **Optional inner radius** line – **RI1**;
- in the **Z-coordinates** line (the Z-coordinates of the start, defining the position and length of the tank) – **0** and **Z1** respectively;
- in the **Starting angle (degrees)** line (the starting angle of the cylinder sector) – **0**;
- in the **Ending angle (degrees)** line (the ending angle of the cylinder sector) – **90**.

Click **OK**.

7 Similarly, a model of a small pipe can be constructed by initially orienting the Z axis of the working plane, which corresponds to the axis of the cylinder being formed, in the required direction.

Expand the Z-axis: **Utility Menu**→**WorkPlane**→**Offset WP by Increments**→in the **Offset WP** window, in the **XY, YZ, ZX Angles** line, set the values to **0, -90** (rotate the work plane relative to the X axis by -90°)→click **OK**.

Then create a quarter cylinder: **Main Menu**→**Preprocessor**→**Modeling**→**Create**→**Volumes**→**Cylinder**→**By Dimensions**. In the **Create Cylinder by Dimensions** window, enter the following scalar parameters for the small pipe in the lines:

- in the **Outer radius** line – **RO2**;
- in the **Optional inner radius** line – **RI2**;
- in the **Z-coordinates** line (Z-coordinates of the beginning determining the position and length of the pipe) – **0** and **Z2** respectively;
- in the **Starting angle (degrees)** line (the starting angle of the cylinder sector) – **-90**;
- in the **Ending angle (degrees)** line (the ending angle of the cylinder sector) – **0**.

Click **OK**. Return the work plane to its original state: **Utility Menu**→**WorkPlane**→**Align WP with**→**Global Cartesian**.

8 In the constructed model, two cylindrical sectors are created, resulting in the presence of two volumes at their points of intersection. To combine these volumes and to be able to delete unnecessary parts of the volumes, we will use the Overlap procedure, which will separate and combine the elements of the model at the places of their intersection: **Main Menu**→**Preprocessor**→**Modeling**→**Operate**→**Booleans**→**Overlap**→**Volumes**→

in the **Overlap Volumes** menu that opens, click the **Pick All** button. As a result, five non-intersecting volumes will be formed from two intersecting volumes.

9 To facilitate future interactions with the model, activate the display of volume and line numbers on the screen: **Utility Menu**→**PlotCtrls**→**Numbering**→in the **Plot Numbering Controls** window, activate the **Volume numbers On** and **Line numbers On** options→press **OK**.

10 There are several ways to scale and rotate a model in ANSYS. You can change the scale by holding down Ctrl on the keyboard and rotating the mouse wheel. You can rotate the model by holding down Ctrl and the right mouse button on the keyboard simultaneously, moving the mouse around the screen. You can move the model around the screen in a similar way, but by holding down the left mouse button. As an alternative, you can use the toolbar to the right of the working window to scale and rotate the model.

This problem will be approached using an alternative method. We will establish the coordinates of the observation point for the model: **Utility Menu**→**PlotCtrls**→**View Settings**→**Viewing Direction**. In the opened dialog panel **Viewing Direction**, set the coordinates of the view point in the **Coords** of view point line to **-0.07, -0.02, 0.025**. In the **Coord axis orientation** line, select the **Y-axis up** direction and click **OK**.

11 Remove unnecessary internal parts of the volumes. First, there should be no element of the cylindrical tank inside the small pipe. Second, since the distribution of temperature fields at the intersection of the pipe and the tank is considered, the part of the pipe located in the tank can be removed.

To execute this operation, navigate: **Main Menu**→**Preprocessor**→**Modeling**→**Delete**→**Volume and Below**.

Select volumes **No. 3** and **No. 4** with the mouse, or, in the **Delete Volume and Below** menu that opens, write **3,4** in the input window and click **OK**.

12 When repeatedly accessing a group of model elements during the process of solving a problem, the group of elements can be saved as a component with a unique name and loaded when necessary. Let us create the AREMOTE component, which contains the most distant areas of the tank from the intersection point.

First, you need to activate the model elements: **Utility Menu**→**Select**→**Entities**. In the **Select Entities** dialog box that appears, execute the following actions:

- select **Areas** in the first line. In the second line, select **By Location**. Select **Z Coordinates** and in the **Min,Max** field, enter the value **Z1**. Click **Apply**;
- select **Y Coordinates** and **Also Select**. Set the **Min,Max** field to 0 and click **OK**.

Next, create a component **Utility Menu→Select→Comp/Assembly → Create Component**. In the **Create Component** window that opens, in the **Component name** line, enter the name of the component **AREMOTE**, and in the **Component is made of** line, select **Areas**. Click **OK**.

13 When generating a finite element mesh with hexahedral elements, the target volume must meet two topological requirements: be bounded by no more than six surfaces and have each surface contour limited to 4 edges. For complex geometries, this necessitates preprocessing steps including surface merging and edge consolidation.

First, we combine the surfaces: **Main Menu→Preprocessor→Meshing→Mesh→Volumes→Mapped→Concatenate→Areas**. In the **Concatenate Areas** window that opens, click the **Pick All** button to combine the pre-selected surfaces at the ends of the tank that are furthest from the connection point with the pipe.

We connect the lines in a similar way:

- display the model as a wireframe **Utility Menu→Plot→ Lines**;
- combine lines **Main Menu→Preprocessor→Meshing→Mesh→ Volumes→Mapped→ Concatenate→Lines**. Select lines #12 and #7 with the mouse or, in the **Concatenate Lines** window that opens, type **12.7** in the input window and click **Apply**. Select lines #10 and #5 or enter their numbers in the input window. Click **OK**.

14 The next stage of solving the problem is the application of a finite element mesh with preliminary division of part of the model contour into a given number of elements.

Activate all elements of the model: **Utility Menu→Select→Everything**. Then, set the mesh density along the specified lines: **Main Menu→Preprocessor→Meshing→Size Cntrls→ Manual Size→Lines→Picked Lines**. Select lines #6 and #20 with the mouse or, in the **Element Size on Picked Lines** window that opens, type **6.20** in the input window. Click **OK**. In the **Element Size on Picked Lines** window that opens, in the **No. of element divisions** field, enter the **number of elements** as **4** and click **OK**.

Similarly, divide line **No. 40** into **6** elements.

15 Set the default element size when splitting and apply the finite element mesh: **Main Menu→Preprocessor→Meshing→Size Cntrls→ Manual Size→Global→Size**. In the **Global Element Sizes** window that opens, in the **Element edge length** line, enter the number **0.01016** and click **OK**.

Next, **Main Menu→Preprocessor→Meshing→Mesher Opts**. In the **Mesher Options** window that opens, activate the **Mapped** parameter in the **Mesher Type** section and click **OK**. In the **Set Element Shape** window that appears, in the **2D Shape key** line, select **Quad** in the drop-down menu and click **OK**.

Apply the finite element mesh: **Main Menu→Preprocessor→Meshing→Mesh→Volumes→Mapped→4 to 6 sided→**in the **Mesh Volumes** window that opens, click **Pick All**.

16 Disable line and volume numbering on the screen: **Utility Menu→PlotCtrls→Numbering**. In the **Plot Numbering Controls** window, deactivate the **Volume numbers Off** and **Line numbers Off** options. Click **OK**.

17 The next step is to set the loads on the model and boundary conditions. We shall categorise the problem as stationary Steady-State: **Main Menu→Solution→Analysis Type→New Analysis→**select **Steady-State→**Click **OK**.

18 Set the value of the initial temperature of the model: **Main Menu→Solution→Define Loads→Apply→Thermal→Temperature→Uniform Temp→**in the window that appears, in the Uniform temperature line, enter the value **232+273→**click **OK**.

19 We define the heat exchange between the walls of the tank and the liquid flowing through it in the form of convective boundary conditions for nodes on the inner surface of the tank.

First, it is required to change the coordinate system to cylindrical: **Utility Menu→WorkPlane→Change Active CS to→Global Cylindrical**. After that, select the nodes belonging to the inner surface of the tank: **Utility Menu→Select→Entities**. In the opened **Select Entities** window, in the first line, select **Nodes**, in the second **By Location**, below select **X Coordinates** and **From Full**. Enter the scalar parameter of the radius of the inner surface of the tank **RI1** in the **Min,Max** field and click **OK**. Next, set the heat transfer coefficient on the inner surface of the tank: **Main Menu→Solution→Define Loads→Apply→Thermal→Convection→On Nodes**. In the opened **Apply CONV on Nodes** window, click **Pick All**. In the **Apply CONV on Nodes** window, in the **Film coefficient** line, enter the heat transfer coefficient value of **1420**, and in the **Bulk Temperature** line, determine the temperature of the tank contents equal to **232+273**. Click **OK**.

20 Set a constant temperature on the **AREMOTE** component: **Utility Menu→Select→Comp/Assembly→Select Comp/Assembly→**in the **Select Component or Assembly** window, make sure that by component name is selected and click **OK**; in the next window, select the **AREMOTE** component and click **OK**.

Activate the nodes belonging to the component: **Utility Menu→Select→Entities**. In the **Select Entities** window, select **Nodes** in the first line and **Attached To** in the second. Below, select **Areas**, **all** and make sure **From Full** is active. Click **OK**.

Set the temperature in nodes: **Main Menu→Solution→Define Loads→Apply→Thermal→Temperature→On Nodes**. In the window that opens

Apply TEMP on Nodes, click **Pick All**, and in the next window, in the **Load TEMP value** line, enter **232+273**, set **TEMP** in the **DOFs to be constrained** field. Click **OK**.

21 Convective boundary conditions are applied to nodes on the pipe's inner surface to account for heat transfer between the pipe walls and the internal fluid flow.

First, you need to rotate the working plane by **90** degrees around the X axis: **Utility Menu**→**WorkPlane**→**Offset WP by In-crements**→in the **Offset WP** window, in the **XY, YZ, ZX Angles** line, set the values to **0, -90** (rotate the working plane relative to the X axis by -90°)→click **OK**.

After that, create a local coordinate system based on the location of the current work plane: **Utility Menu**→**WorkPlane**→**Local Coordinate Systems**→**Create Local CS**→**At WP Origin**. In the **Create Local CS at WP Origin** window, in the **Ref number of new coord sys** line, enter the local coordinate system number **11**, in the **Type of coordinate system** line, select **Cylindri-cal 1** and click **OK**.

Activate the nodes belonging to the inner surface of the small pipe: **Utility Menu**→**Select**→**Entities**. In the **Select Entities** window, select **Nodes** in the first line and **By Location** in the second. Below, select X Coordinates and in the **Min,Max** field, enter the values of the radius of the inner surface of the small pipe as a scalar parameter **RI2**. Click **OK**.

The heat transfer coefficient must be set: **Main Menu**→**Solution**→**Define Loads**→**Apply**→**Thermal**→**Convection**→**On Nodes**. In the opened **Apply CONV on Nodes** window, click **Pick All**. In the **Apply CONV on nodes** window, in the **Film coefficient** line, enter the heat transfer coefficient value of **-2** (the values will be taken from the material properties table 2), and in the **Bulk Temperature** line, determine the temperature of the tank contents equal to **38+273**. Click **OK**.

22 Activate all elements of the model: **Utility Menu**→**Select**→**Everything**.

23 Activate all elements of the model: **Utility Menu**→**Select**→**Everything**.

24 Let us display convective heat transfer in the form of arrows: **Utility Menu**→**PlotCtrls**→**Symbols**→in the Symbols window, in the **Surface Load Symbols** line, select **Convect FilmCoef**, and in the **Show pres and convect as** line, select **Arrows** (i.e., display as arrows)→click **OK**.

25 Let us restore the working plane and coordinate system to their original form.

First, **Utility Menu**→**WorkPlane**→**Change Active CS to**→**Global Cartesian**. Then, **Utility Menu**→**WorkPlane**→**Align WP with**→**Global Cartesian**.

26 The type of analysis selected in this problem is stationary, but the material properties depend on temperature. Therefore, to take this dependence into account when solving, it is necessary to set the number of substeps for problem-solving with automatic step selection based on loading time: **Main Menu**→**Solution**→**Load Step Opts**→**Time/Frequency**→**Time and Substeps**. In the **Time and Substep Options** window that opens, set the number of substeps in the **Number of substeps** line to 50. Enable **Automatic time stepping** by checking the **ON** box. Click **OK**.

27 Save the intermediate database file: **Utility Menu**→**File**→**Save as**→select the folder in which the database is saved and specify the file name in Latin letters→click **OK**.

28 Launch the task for solution: **Main Menu**→**Solution**→**Solve**→**CurrentLS**→press **OK**. Wait for the calculation to complete until the message “**Solution is done!**” appears.

29 Let us display the results of the solution in the form of a spatial distribution of temperature fields.

To facilitate visualisation, you can enable the finite element mesh on the model by navigating to: **Utility Menu**→**PlotCtrls**→**Style**→**Edge Options**. In the **Edge Options** window, set the **Edge only/ALL** parameter in the **Element outlines for non-contour/contour plots** line and click **OK**.

Next, display the temperature field on the model: **Main Menu**→**General Postproc**→**Plot Results**→**Contour Plot**→**Nodal Solution**. In the **Contour Nodal Solution Data** window, select **Nodal Solution**→**DOF Solution**→**Nodal Temperature**→click **OK**.

30 We shall construct a vector field of heat flow densities at the intersection of the pipe with the reservoir.

First, it is required to activate the previously created cylindrical local coordinate system number 11: **Utility Menu**→**WorkPlane**→**Change Active CS to**→**Specified Coord Sys**. In the **Change Active CS to Specified CS** window, enter 11 in the **Coordinate system number** line. Click **OK**.

Next, select the elements in the area where the pipe and tank intersect: **Utility Menu**→**Select**→**Entities**. In the **Select Entities** window, select **Nodes** in the first line and **By Location** in the second. Below, select **X Coordinates** and in the **Min,Max** field, enter the values of the radius of the outer surface of the small pipe as a scalar parameter **RO2**. Click **Apply**. Select **Elements** in the first line and **Attached to** in the second. Below, select **Nodes** and click **Apply**. Select **Nodes** in the first line and **Attached to** in the second. Select **Elements** and click **OK**.

Then, you should read the results of the last step of the solution: **Main Menu→General Postproc→Read Results→Last Set**. In earlier versions of **ANSYS**, the solution results may be available for display immediately after the calculations are completed.

After that, display the vector field of heat flux densities at the intersection of the pipe with the tank: **Main Menu→General Postproc→Plot Results→Vector Plot→Predefined**. In the **Vector Plot of Predefined Vectors** window, in the **Vector item to be plotted** section, select **Flux & gradient** in the left list of the panel and **Thermal flux TF** in the right list. Click **OK**.

6 IMPLEMENTATION EXAMPLES FOR ANSYS WORKBENCH ENVIRONMENT

Problem 6.1 Fundamentals of ANSYS Workbench: static strength analysis

6.1.1 Problem statement

This example allows you to get acquainted with the capabilities of the Ansys Workbench package, consider the main stages of modeling in this package and perform static strength analysis.

A silicon wafer with dimensions of $1 \times 0.1 \times 0.02$ mm is considered. It is assumed that a force of 1 N acts on the wafer.

6.1.2 Implementation of the problem using the Ansys Workbench graphical interface

1 The solution to the problem begins with launching Ansys Workbench, which is executed from the main menu **Start** → **Programs** → **Ansys 2020R1** → **Workbench 2020R1**. After loading, the main program window will be displayed, which consists of several windows, as shown in Figure 6.1.

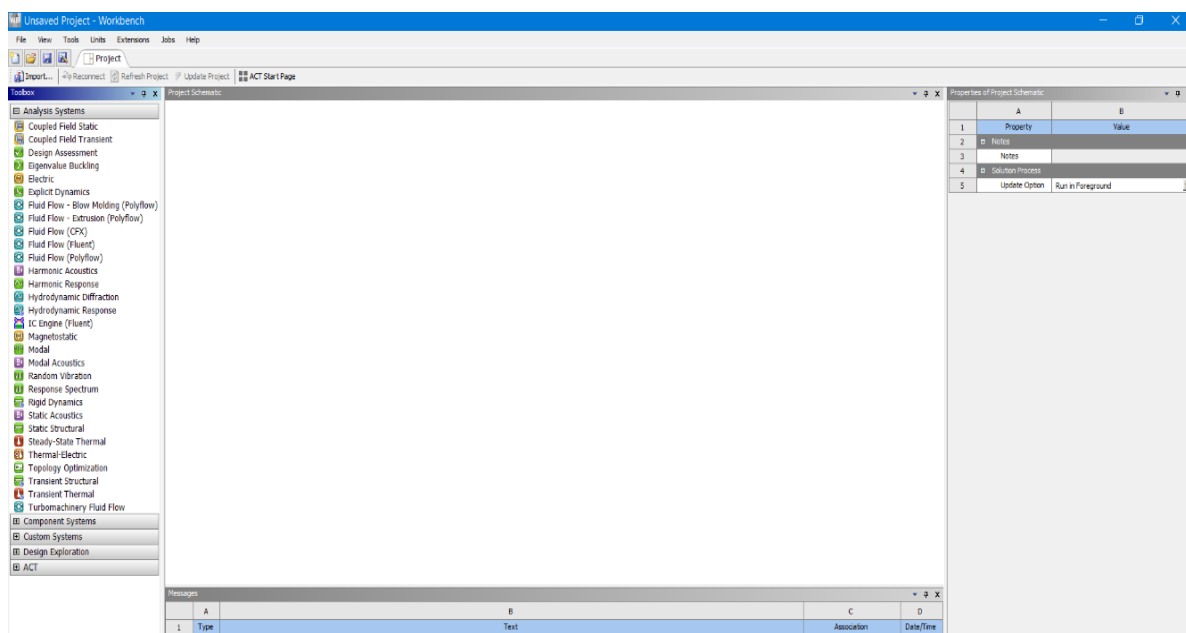


Figure 6.1 – Workbench main window

The Project Schematic window is the main window of the project, located in the center of the main Workbench window. It contains the structural components of the project and the links between them. This approach enables the visualization of project components and the management of connections among its discrete blocks.

The Toolbox window is a project tool window located to the left of the Project Schematic window. It contains several sections displayed as drop-down lists (Figure 6.2)

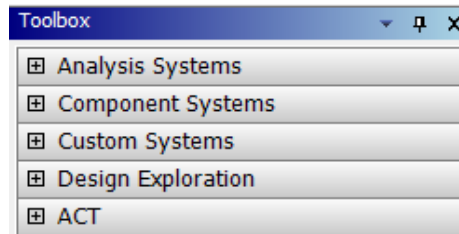


Figure 6.2 – Project tools window

Sections of the tool window:

– Analysis Systems contains ready-made templates for various types of numerical analysis (Figure 6.3);

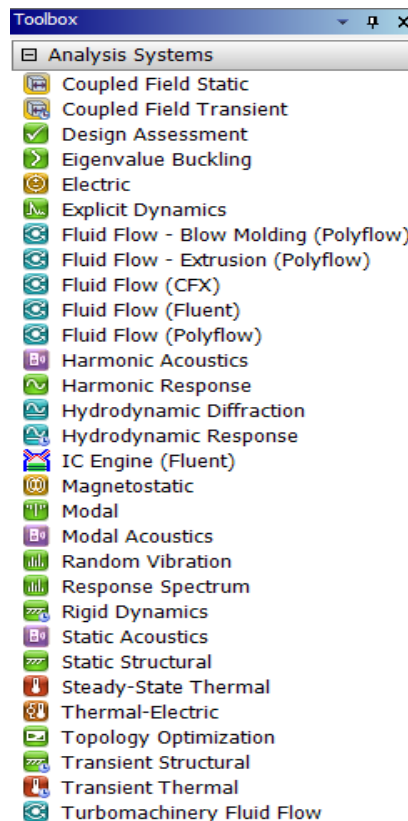


Figure 6.3 – Analysis Systems section

- Component Systems contains individual components of engineering analysis, such as a geometric model, finite element mesh, etc.;
- Custom Systems contains ready-made bundles of templates for solving interdisciplinary problems;
- Design Exploration contains tools for solving optimization problems.

2 In the Toolbox section, hold down the left mouse button on Static Structural and drag it to the green free rectangle of the Project Schematic window (until it turns red), release the left mouse button. As a result, a project template will appear in the Project Schematic window (Figure 6.4). The Engineering Data line is marked with a green check mark (the material is already set by default – structural steel). The remaining lines of the project Geometry, Model, Setup, Solution, Results contain a blue question mark.

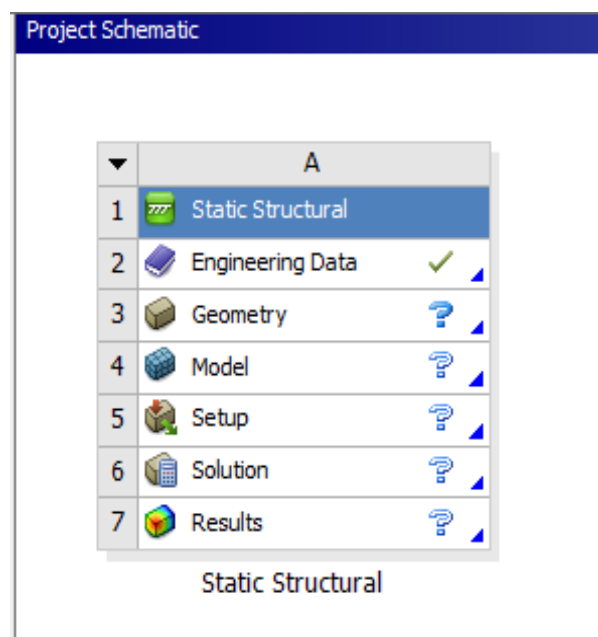


Figure 6.4 – Result of selecting the type of analysis

Thus, after adding a project template (analytical block), it is necessary to define all the data in the order of the analytical block cells (Figure 6.4). Each cell is an object, the properties and methods of which are accessed by right-clicking after selecting the appropriate item from the context menu. The list of properties and methods depends on the cell type. The cell containing all the necessary data is marked with a green check mark.

In this case, all messages caused by incorrect selection of parameters in the program settings, incorrect selection of conditions in the task formulation, etc., are displayed in the Messages window. It is desirable that there should be no message in the form of a warning (Warning), and in the case of an error (Error), the solution of the task is stopped.

Figure 6.4 demonstrates that the project template (analytical block) added to the Workbench project includes six sections corresponding to the modeling stages:

1) Engineering Data – materials and mathematical models (double-clicking the left mouse button on this section calls up the Engineering Data interface, which allows for the description of the physical and mechanical properties of the materials of the calculated structure, as well as the input parameters of the mathematical model);

2) Geometry – geometric model (double-clicking the left mouse button on this section calls the geometry editor selected in the Workbench settings, for example, Design Modeler);

3) Model – finite element model (double-clicking the left mouse button on this and subsequent sections of the Static Structural template calls up the Workbench Mechanical application, which provides, among other things, the definition of materials, the creation of coordinate systems, and the description of contact interactions between individual bodies and groups of bodies);

4) Setup – setting boundary conditions, as well as choosing a solution method and describing the solution procedure;

5) Solution – a section of the solution that shows the state of the modeling, and if the problem is solved, the results of the calculation performed, which can be further processed and presented in graphical form or used in other calculations as initial data;

6) Results – the results processing section that shows the status of the results.

Many calculation modules, such as Transient Structural, Transient Thermal, Electric, Thermal-Electric and others, have a similar structure.

3 To set the material properties, right-click on Engineering Data and then click Edit. In the group of windows that appears, go to the Outlook of Schematic A2: Engineering Data window and enter the name of the new material Silicon (Figure 6.5).

	A	B	C	D	E
1	Contents of Engineering Data			Source	Description
2	Material				
3	Silicon				
4	Structural Steel				Fatigue Data at zero mean stress comes from 1998 ASME BPV Code, Section 8, Div 2, Table 5-110.1
*	Click here to add a new material				

Figure 6.5 – Setting up a new material

Next, we will set the properties of the new material. To do this, open the Physical Properties and Linear Elastic lists. We will assume that the material has isotropic properties. We will set the following properties: density is 2,330 kg/m³, Young's modulus is 1.9·10⁵ MPa, Poisson's ratio is 0.42 (Figure 6.6).

Properties of Outline Row 4: Silicon					
	A	B	C	D	E
1	Property	Value	Unit		
2	Material Field Variables	Table			
3	Density	2330	kg m ⁻³		
4	Isotropic Elasticity				
5	Derive from	Young's ...			
6	Young's Modulus	1,9E+05	MPa		
7	Poisson's Ratio	0,42			
8	Bulk Modulus	3,9583E+11	Pa		
9	Shear Modulus	6,6901E+10	Pa		

Figure 6.6 – Assigning isotropic properties to silicon

4 To create the sample geometry, right-click on the Geometry item in the project template and select New Design Modeler Geometry (Figure 6.7). This will launch Design Modeler – the Ansys geometric editor (Figure 6.8).

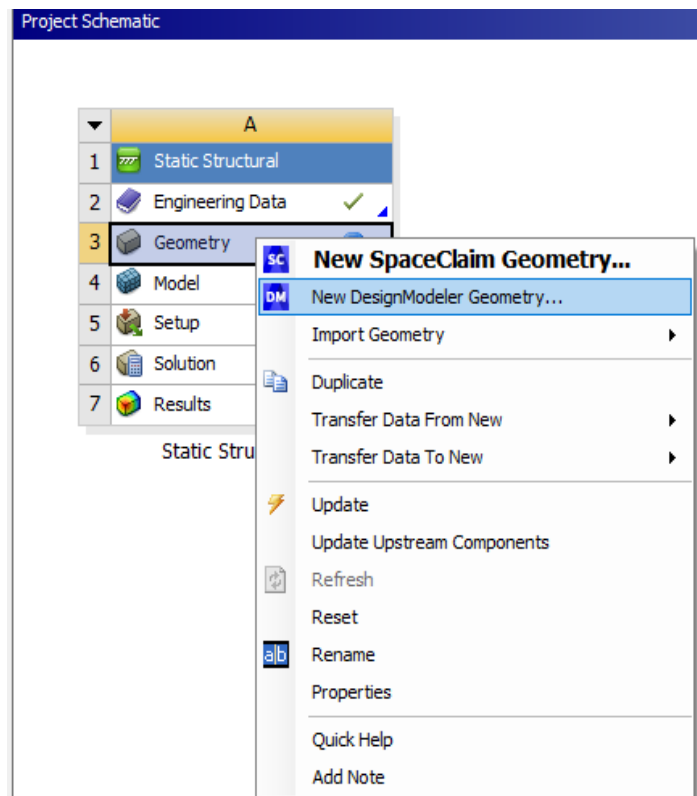


Figure 6.7 – Launching the Design Modeler application

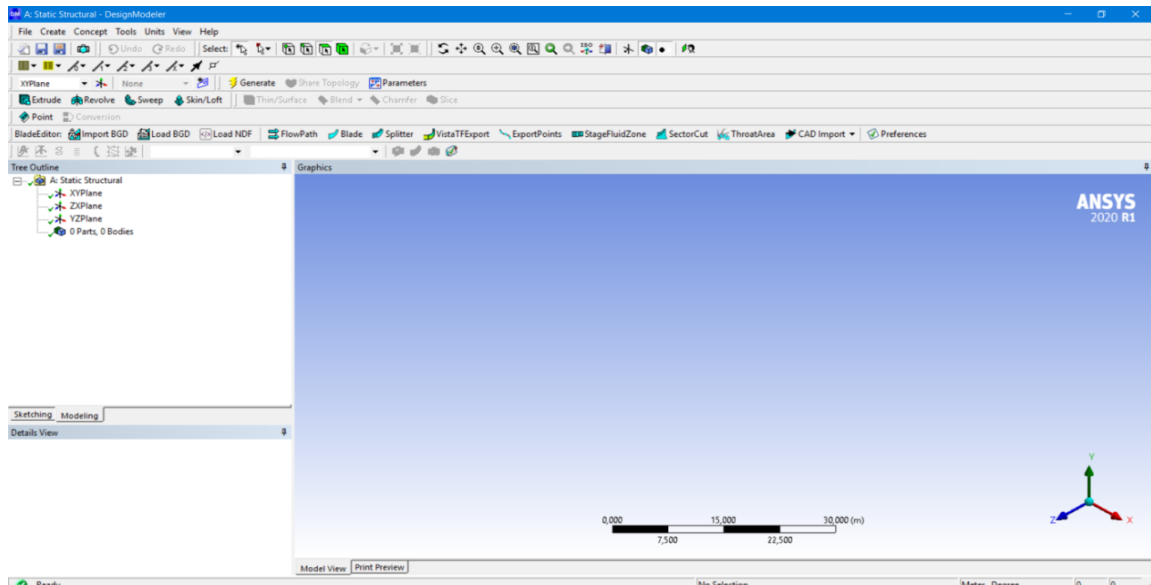


Figure 6.8 – Design Modeler Window

Design Modeler can work in two modes: 2D Sketching and 3D Modeling. In Sketching mode, you can use five toolboxes to create 2D sketches:

- Draw Toolbox – drawing lines, rectangles and polylines;
- Modify Toolbox – modification, trimming, cutting and insertion;
- Dimensions Toolbox – defining dimensions;
- Constraints Toolbox – imposing constraints on mutual arrangement;
- Settings Toolbox – displaying a grid on a plane.

Geometric models in Design Modeler can be built both on the basis of sketches and using ready-made 3D primitives, a set of which is located in the Create menu (Figure 6.9).

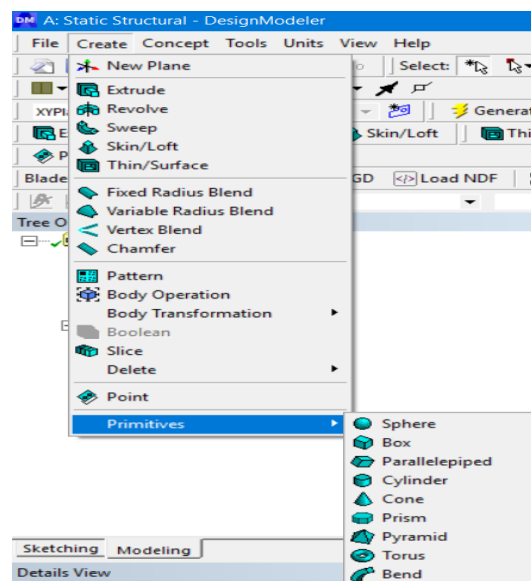


Figure 6.9 – Tools for creating 3D models from primitives

The Primitives command has a drop-down list that allows the user to create simple 3D geometric shapes: sphere, quadrilateral, parallelepiped, cone, pyramid, torus, etc.

The figure shows a view of a plate created using the Primitives tool. At the first stage, a Box object is added to the project tree; then, in the properties window of this object (Details View), the coordinates of the plate must be specified; at the last stage, the object is generated using the Generate command (Figure 6.10).

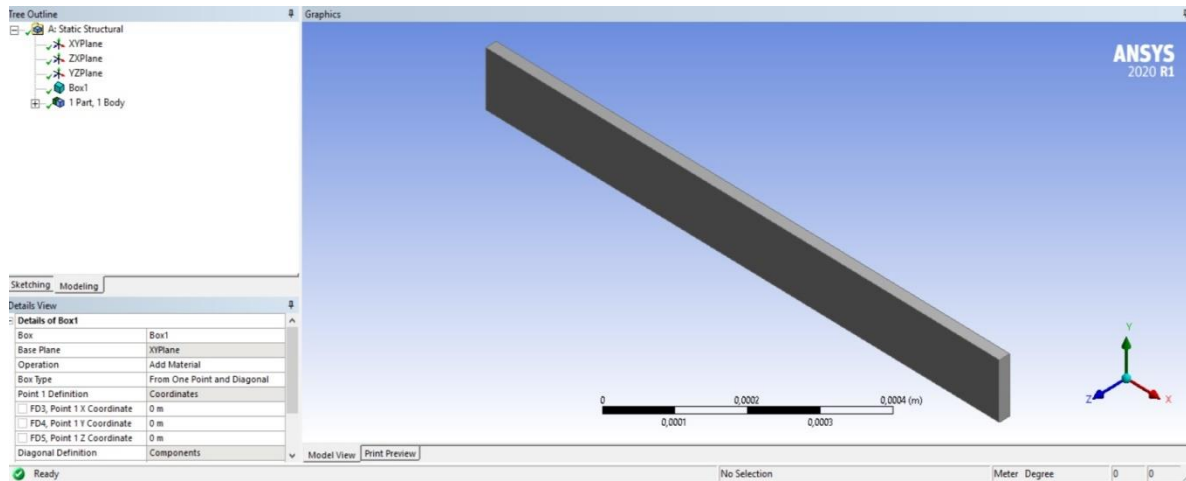


Figure 6.10 – Constructed sample visualization

5 To generate a finite element mesh, right-click on the Model item in the project template and then Edit. Workbench Mechanical will be launched (Figure 6.11).

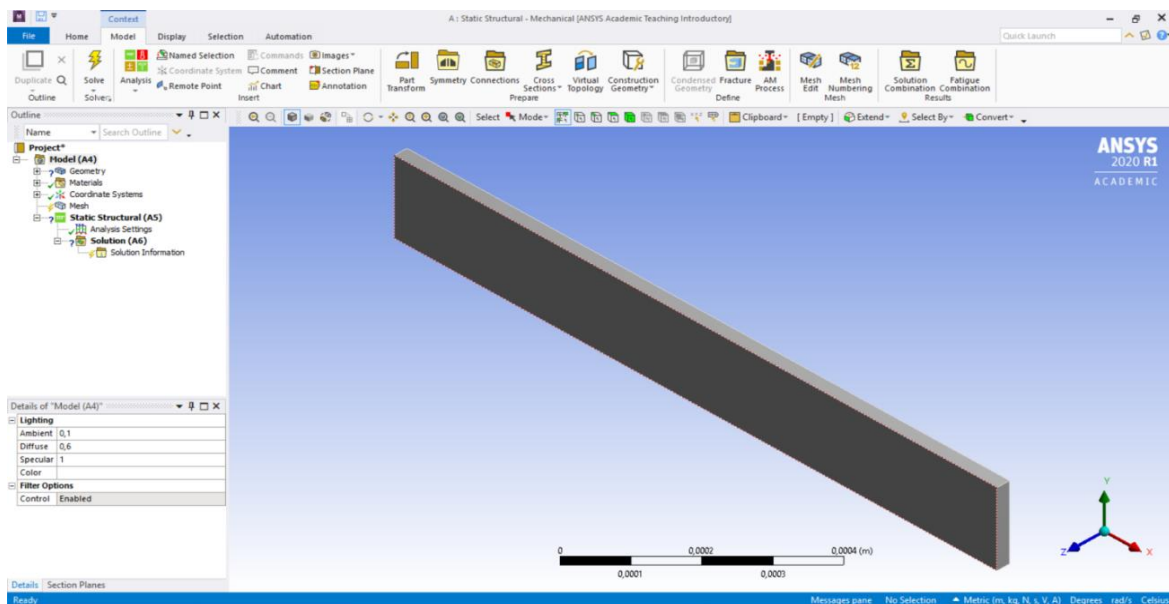


Figure 6.11 – Workbench Mechanical application interface

Workbench Mechanical is a universal shell for various solvers and has a convenient graphical interface, compared to the Mechanical APDL shell. At the same time, Mechanical APDL is an auxiliary modeling environment, it has broader capabilities and is used in cases where the problem cannot be solved by means of Workbench Mechanical.

The calculation model created in Workbench Mechanical is represented in the APDL macro language and can be used to perform calculations in Mechanical APDL. The model is transferred from Workbench Mechanical to Mechanical APDL via a text file containing a set of APDL commands. It is also possible to transfer the geometric model, mesh, and calculation results back from Mechanical APDL to Workbench Mechanical.

The interface of the Workbench Mechanical application has a structure similar to the interfaces of the Design Modeler applications. Many interface elements in these applications are identical, for example, tools for controlling the model view, selecting the mode for selecting geometric objects, etc. However, a number of tools and interface elements depend on the application used. The interface of the Static Structural module consists of the main menu, the toolbar, the project tree window, the properties window, and the graphics window.

The calculation model in the Workbench Mechanical interface is presented as a tree structure with many sections and objects and is located in the Outline window of the Workbench Mechanical graphical interface (Figure 6.12). The calculation model is created by filling the project tree with objects, and the objects with information.

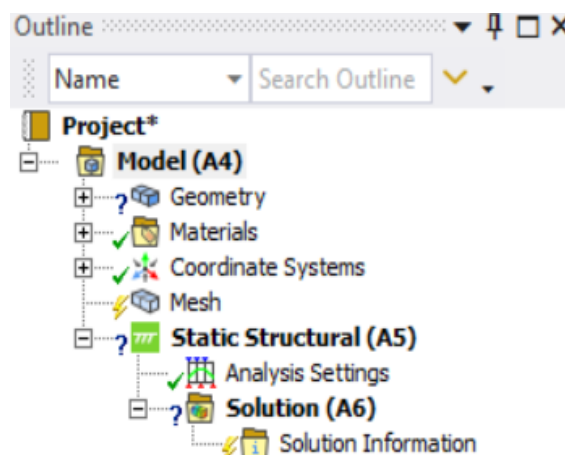


Figure 6.12 – Project tree for the Static Structural module

The tools for managing mesh construction are available in the Mesh section. To generate a finite element mesh, right-click the Mesh project tree item and then Generate Mesh. The resulting FE mesh is shown in Figure 6.13.

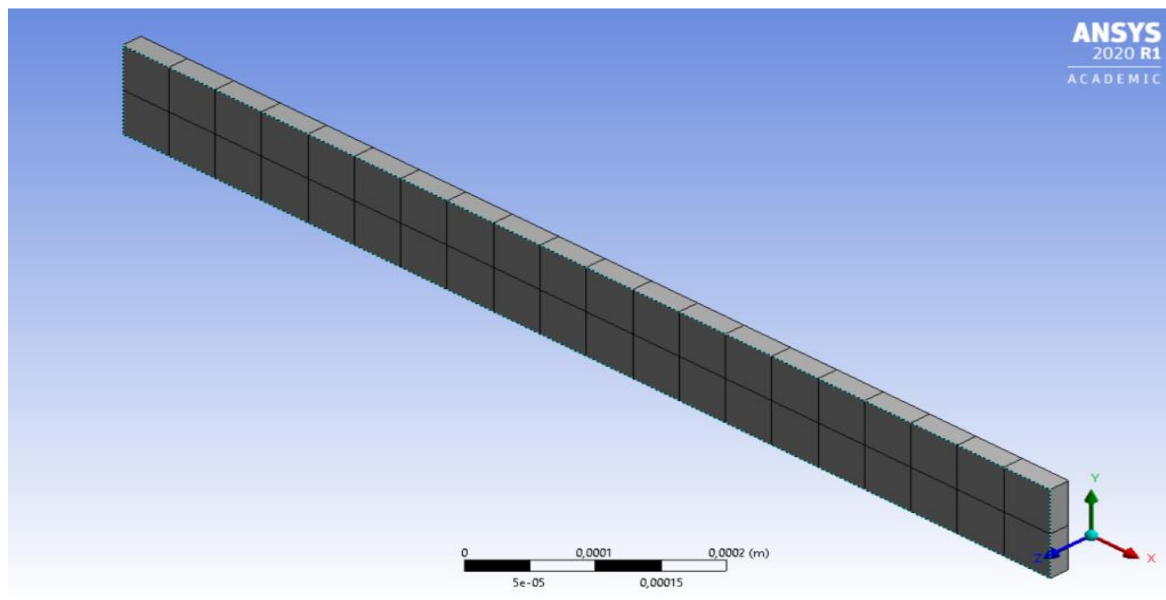


Figure 6.13 – Finite element model of the sample

6 We set the boundary conditions. Select the required edge, right-click the Analysis Settings project tree item and Insert – Fixed Supports.

Then, you need to apply a load of 1 N. Similarly, right-click on the Analysis Settings project tree item and Insert – Force and set the value to 1 N (Figure 6.14).

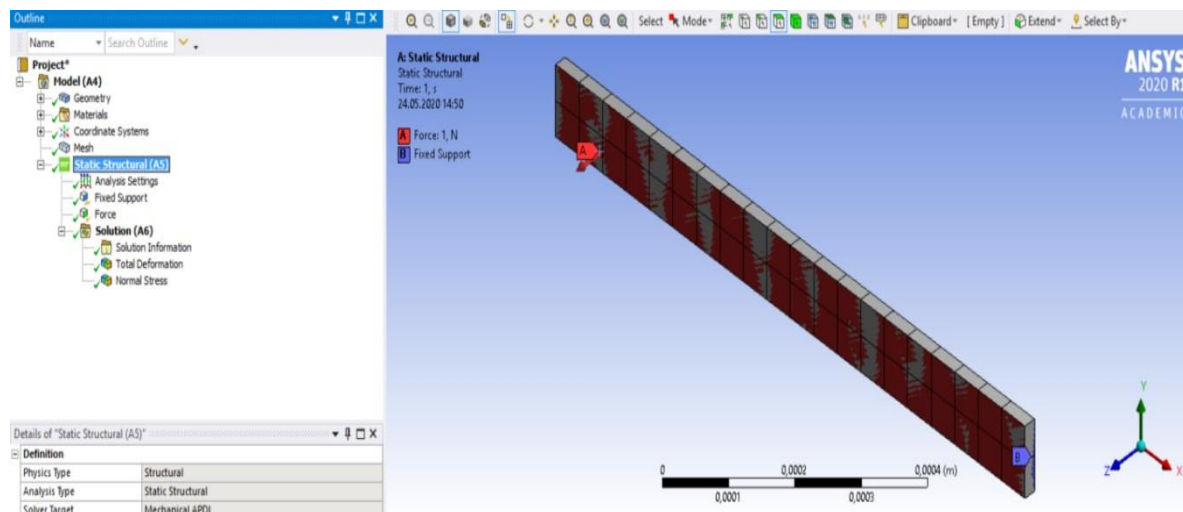


Figure 6.14 – Image of applied loads

To start the solution, you need to click the Solve button.

7 We visualize the obtained solution by two parameters Total Deformation (Figure 6.15) and Normal Stress (Figure 6.16). To do this, select the corresponding items in the Solution menu, and then click Evaluate All Results.

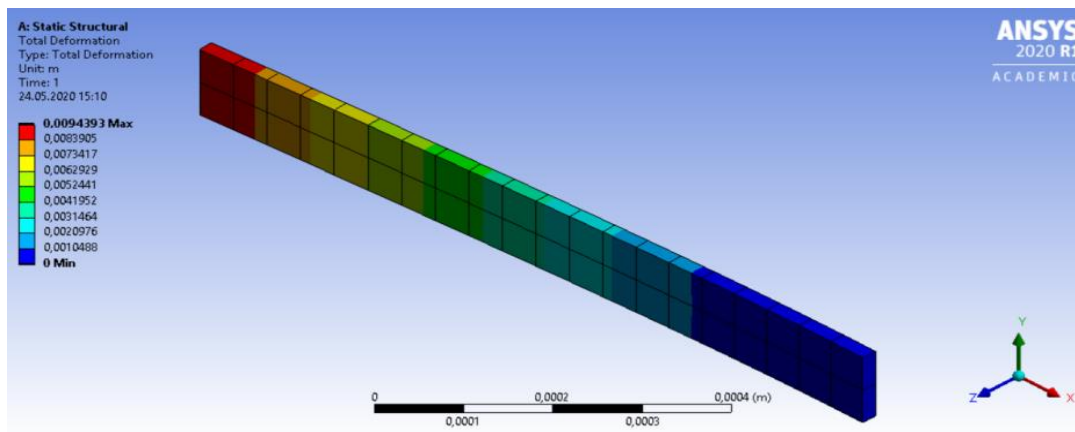


Figure 6.15 – Solution results. Sample deformation

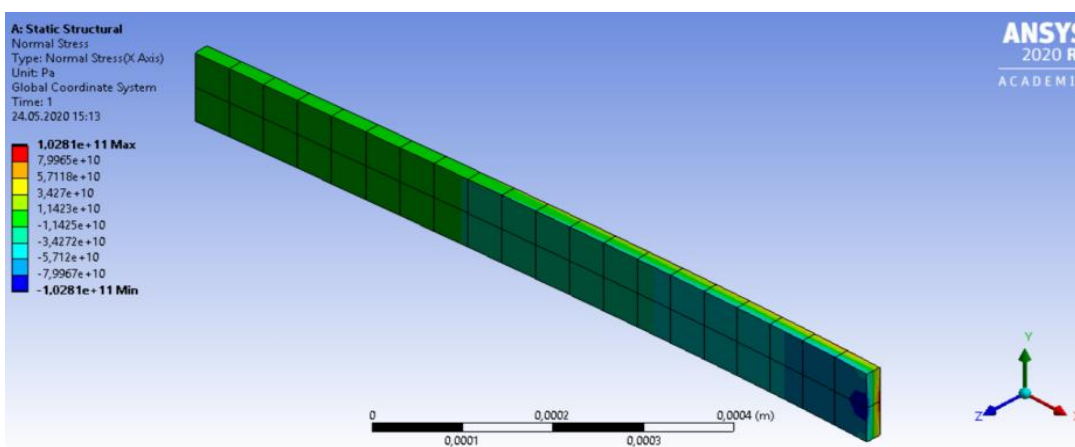


Figure 6.16 – Solution results. Stresses in the sample

8 An important stage of modeling is the selection of materials and definition of their properties. For these purposes, there is a separate materials management module in Workbench, which is connected to the analysis block and represented by the Engineering Data element (see point 3). You can change the material or its properties at any stage of modeling. There are three types of data sources used in the workspace of the materials management module: the current set of materials, the material library and favorites. General Materials is a library of general-use materials that can be used in various types of analysis.

Materials from the library are added to the current set of materials by clicking the button with a yellow plus in the Add column of the Outline Panel. Let us add anisotropic properties of silicon from the General Materials library to the current set of materials in this way (Figure 6.17).

Next, in the Outline window of the Workbench Mechanical graphical interface, select the Model – Geometry item from the tree structure of the calculation model. In the Details of «Solid» window, in Material – Assignment, select Silicon Anisotropic (Figure 6.18).

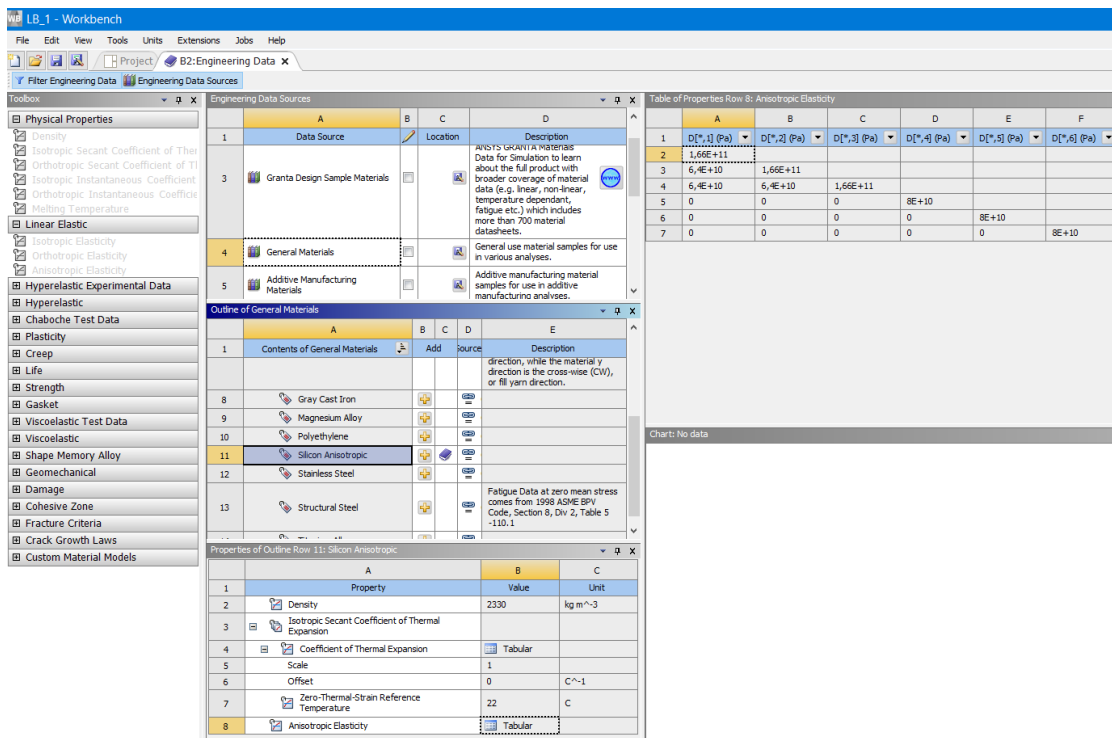


Figure 6.17 – Setting the anisotropic properties of silicon

9 To restart the solution, in the main Workbench window, you need to select Refresh Project and Update Project in sequence.

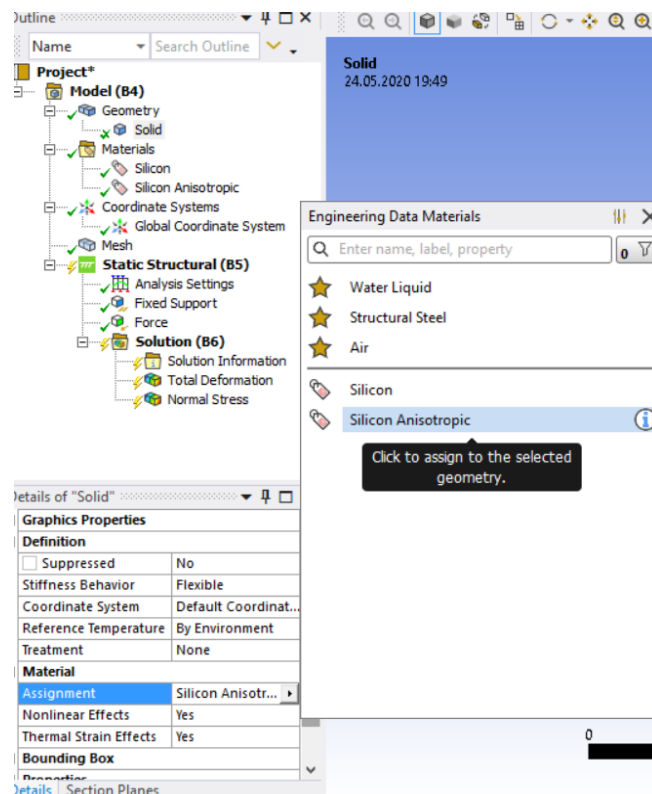


Figure 6.18 – Connecting the anisotropic properties of silicon

The obtained solutions for the previously selected two parameters Total Deformation and Normal Stress are presented in Figures 6.19 and 6.20, respectively.

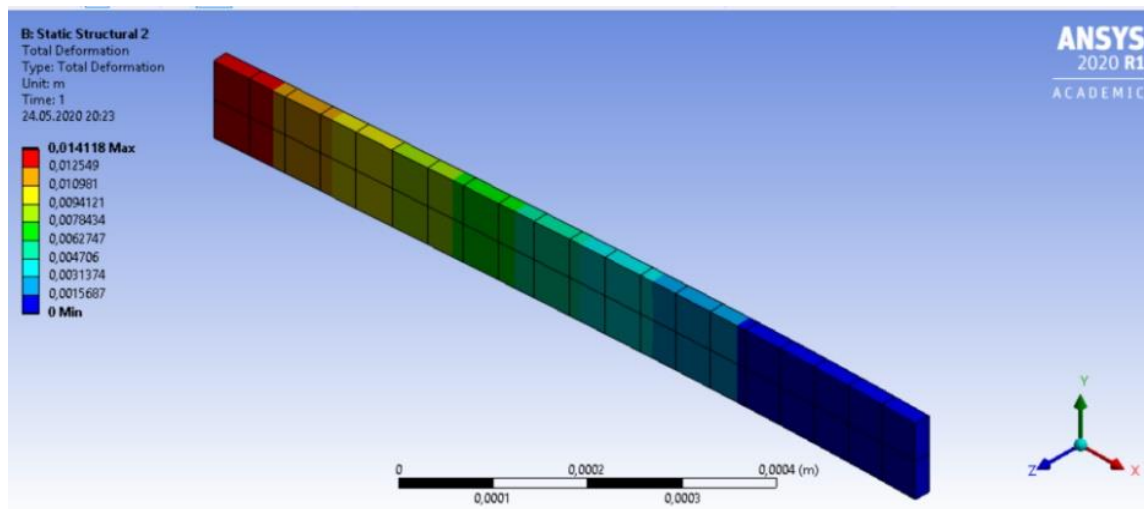


Figure 6.19 – Solution results. Deformation in an anisotropic sample

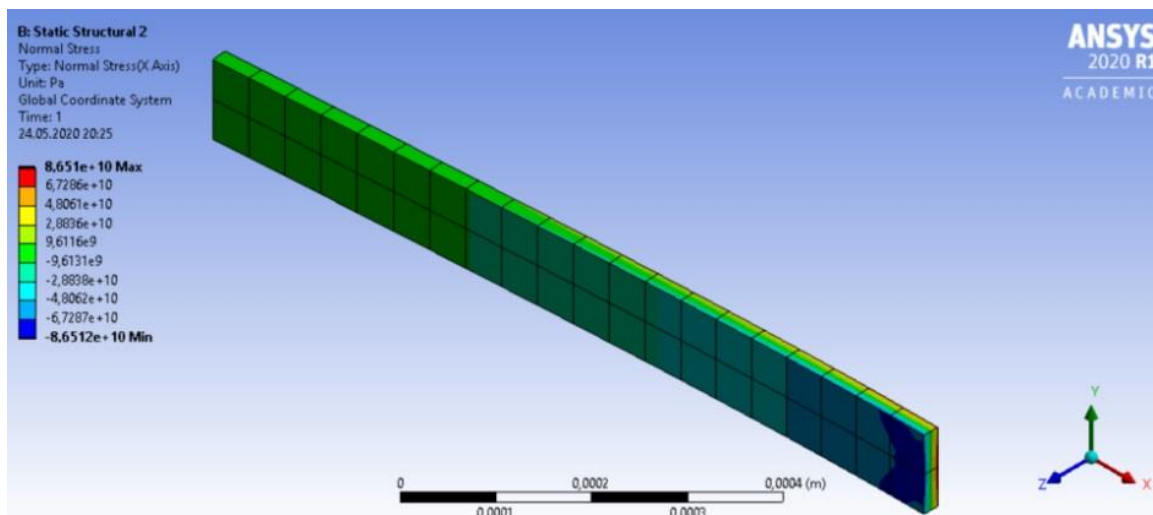


Figure 6.20 – Solution results. Stresses in an anisotropic sample

10 The finite element mesh is generated on the geometric model and serves as the basis for compiling and solving the system of equations in matrix form.

There are two ways to generate a grid:

1 Automatic generation of a grid with default settings when running the solution (see point 5).

2 Generating a grid with user-defined settings. In this case, the user independently determines the features of the grid being created and sets the values of the necessary parameters.

Several ways exist to control the grid density. When constructing a grid model, it is necessary to search for the optimal grid density, regarding the resources of the computing system (memory, time, etc.) and the accuracy of the calculations. A grid with a large number of nodes allows you to find a more accurate solution, but increases the calculation time and memory volume.

To increase the mesh density in the Outline window of the Workbench Mechanical graphical interface, select the Model – Mesh item from the tree structure of the calculation model. Then, in the Details of «Mesh» window, increase Sizing –Resolution to 7 (Figure 6.21).

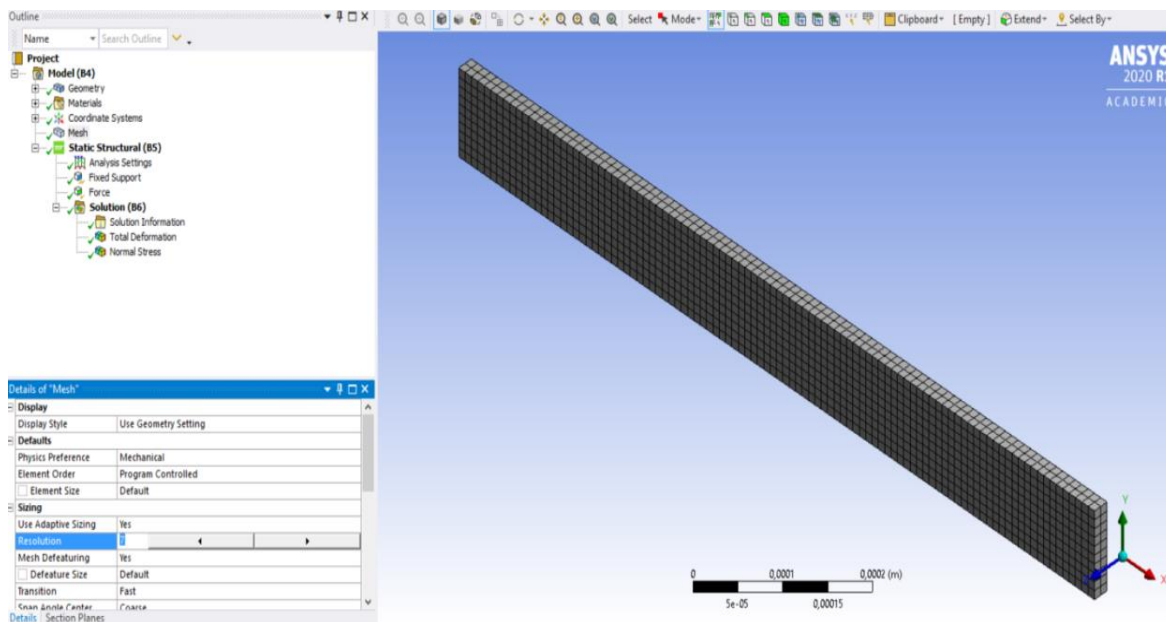


Figure 6.21 – Modified finite element model of the sample

To restart the solution, in the main Workbench window, you need to select Refresh Project and Update Project in sequence.

As a result, the number of elements and nodes in the model increased from 40 and 393 to 2,000 and 11,885, respectively. At the same time, the calculated values of deformations for the model considering the anisotropic properties of the material increased from 14.12 mm to 14.22 mm, and the calculated values of stresses – from 86.5 GPa to 87.6 GPa, respectively.

11 The dimensions of the plate created earlier using the Primitives tool (see point 4) can be changed. For this, select the Box object in the project tree, then change the coordinates of the plate in the properties window of this object (Details View), for example, increase the thickness of the plate to 0.05 mm (Figure 6.22).

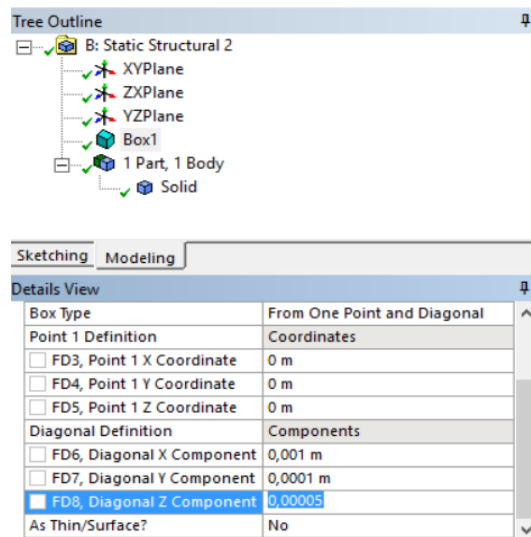


Figure 6.22 – Changing the sample geometry

12 The boundary conditions specified in point 6 can also be changed. Select the Force project tree item. Next, in the Tabular Data window, change the force value, for example, reduce it to 0.1 N (Figure 6.23).

Tabular Data			
	Steps	Time [s]	☒ Force [N]
1	1	0.	0.1
2	1	1.	0.1
*			

Figure 6.23 – Changing boundary conditions

To restart the solution, in the main Workbench window, you need to sequentially select the Refresh Project and Update Project items. The resulting solution for the previously chosen two parameters Total Deformation and Normal Stress are shown in Figures 6.24 and 6.25, respectively.

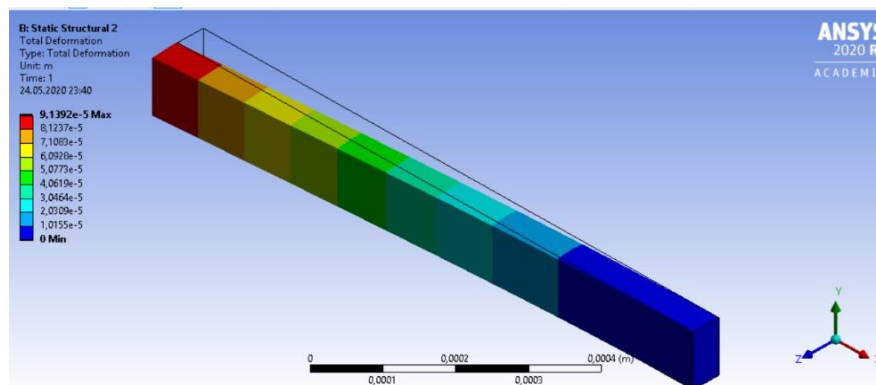


Figure 6.24 – Solution results. Deformation in the sample

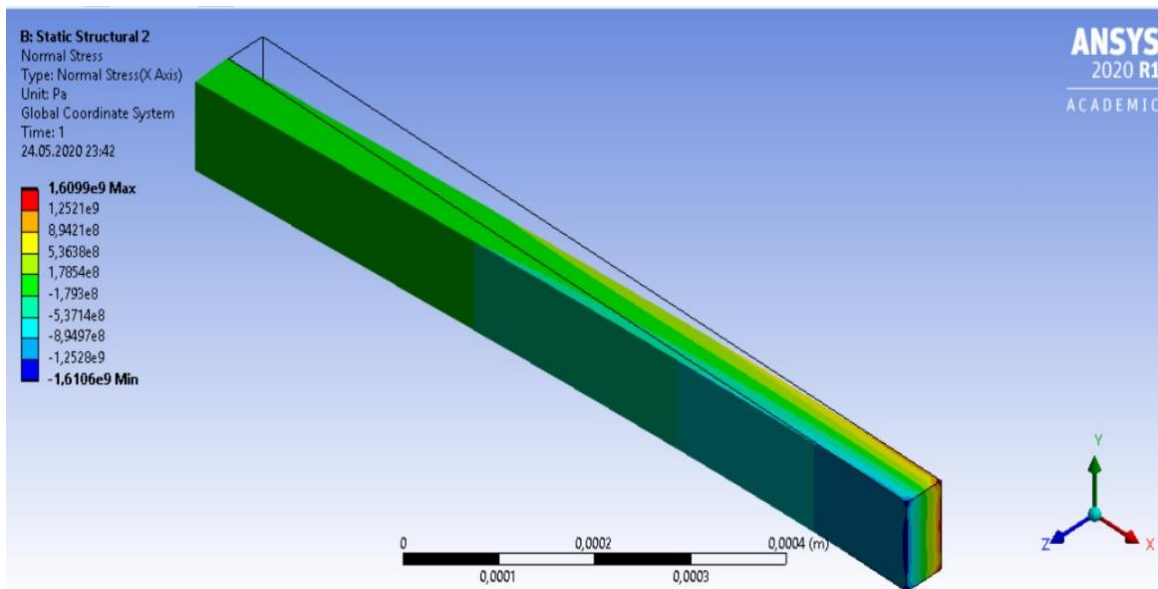


Figure 6.25 – Solution results. Stresses in the sample

Problem 6.2 Calculation of thermoelastic stresses in ANSYS WORKBENCH

6.2.1 Problem statement

This example demonstrates the methods for solving coupled thermal and strength problems in Ansys Workbench. The thermoelastic behavior of a heat-generating cylinder is simulated.

The fuel rod of a nuclear reactor is assembled from cylindrical uranium dioxide pellets. The sizes of the fuel pellets may vary. In the modeling, we will assume that the pellets of the type under consideration have the shape of a cylinder with a radius of 4 mm with an internal hole of radius 1 mm. The height of the cylinder will be considered equal to 10 mm. Uranium dioxide has the following physical properties: density $\rho = 10\,800\text{ kg/m}^3$; Young's modulus $E = 1.75 \cdot 10^{11}\text{ Pa}$; Poisson's ratio $\nu = 0.316$; thermal conductivity $\lambda = 3.487\text{ W/(m}\cdot\text{K)}$; specific heat capacity $C = 310\text{ J/(kg}\cdot\text{K)}$; coefficient of thermal expansion $\alpha = 10^{-5}\text{ K}^{-1}$.

The nuclear fuel is enclosed in a protective shell. The power of the internal heat sources in the fuel elements reaches 109 W/m^3 . In the most common reactors, cooling is carried out by water under a pressure of 15 MPa. The coolant (water) temperature is approximately $300\text{ }^\circ\text{C}$. The heat transfer coefficient is approximately $30,000\text{ W / (m}^2\text{ }^\circ\text{C)}$. The cylindrical symmetry of the fuel pellet and the external load in the fuel element determines the axisymmetric formulation of the numerical modeling problem.

6.2.2 Implementation of the problem using Ansys Workbench graphical interface

Launch Ansys Workbench from the main menu Start → Programs → Ansys 2020R1 → Workbench 2020R1.

In the Toolbox section, hold down the corresponding menu items and double-click the left mouse button on Thermal-Stress from Custom Systems. As a result, a template of the linked static thermal-stress analysis will appear in the Project Schematic window (Figure 6.26). Ansys Workbench supports two types of links established between engineering analysis blocks: a simple link for transferring data (Transfer Data) or a shared link (Share). Both types of links are shown in Figure 6.26. In this problem, the material properties (Engineering Data), the geometric model (Geometry), and the finite element mesh (Model) are used together, so the link of these elements is of the Share type. The results of the thermal problem solution must be transferred to the input of the strength problem solution, so it is necessary to set a shared link (Share) of the Solution element in the Steady-State Thermal analysis and the Setup element in the Static Structural Transfer Data analysis.

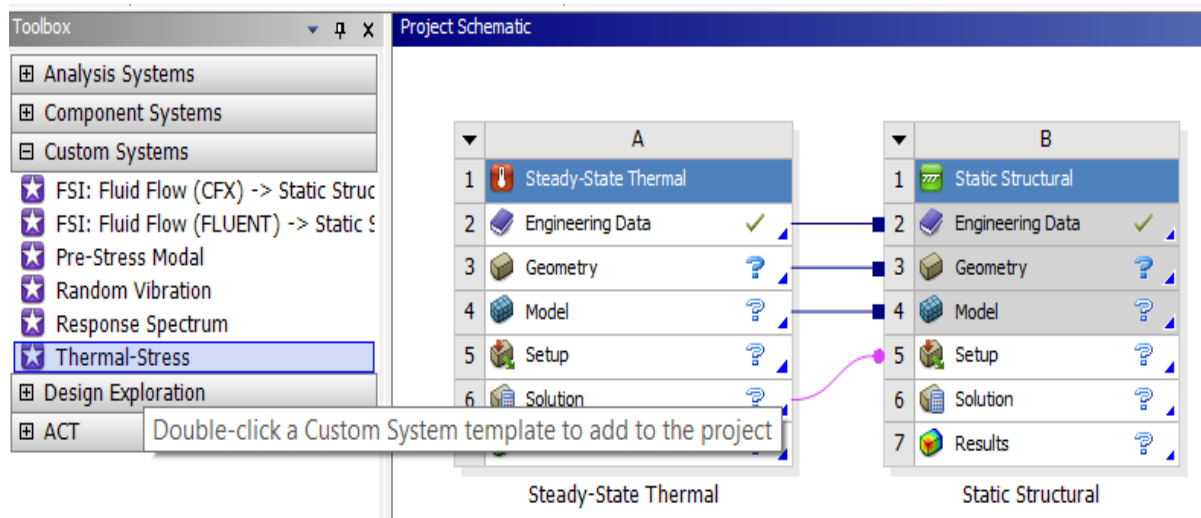


Figure 6.26 – Analysis type selection result

To set the material properties, right-click on Engineering Data and then click Edit. In the group of windows that appears, go to the Outlook of Schematic A2, B2: Engineering Data window, enter the name of the new material UO_2 (uranium dioxide) and set the required properties for the new material (Figure 6.27).

Properties of Outline Row 3: UO2				
	A	B	C	D E
1	Property	Value	Unit	
2	Material Field Variables	Table		
3	Density	10800	kg m ⁻³	
4	Isotropic Secant Coefficient of Thermal Expansion			
5	Coefficient of Thermal Expansion	1E-05	C ⁻¹	
6	Isotropic Elasticity			
7	Derive from	Young's Modulus and Poisson's Ratio		
8	Young's Modulus	1,75E+11	Pa	
9	Poisson's Ratio	0,316		
10	Bulk Modulus	1,5851E+11	Pa	
11	Shear Modulus	6,6489E+10	Pa	
12	Isotropic Thermal Conductivity	3,487	W m ⁻¹ C ⁻¹	
13	Specific Heat, C _p	310	J kg ⁻¹ C ⁻¹	

Figure 6.27 – Properties of uranium dioxide

To create the geometry of the sample, right-click on the Geometry item in the project template and select New DesignModeler Geometry (see item 4 of task 1). The creation of the geometric model begins with the selection of the coordinate plane for the primary constructions (Figure 6.28).

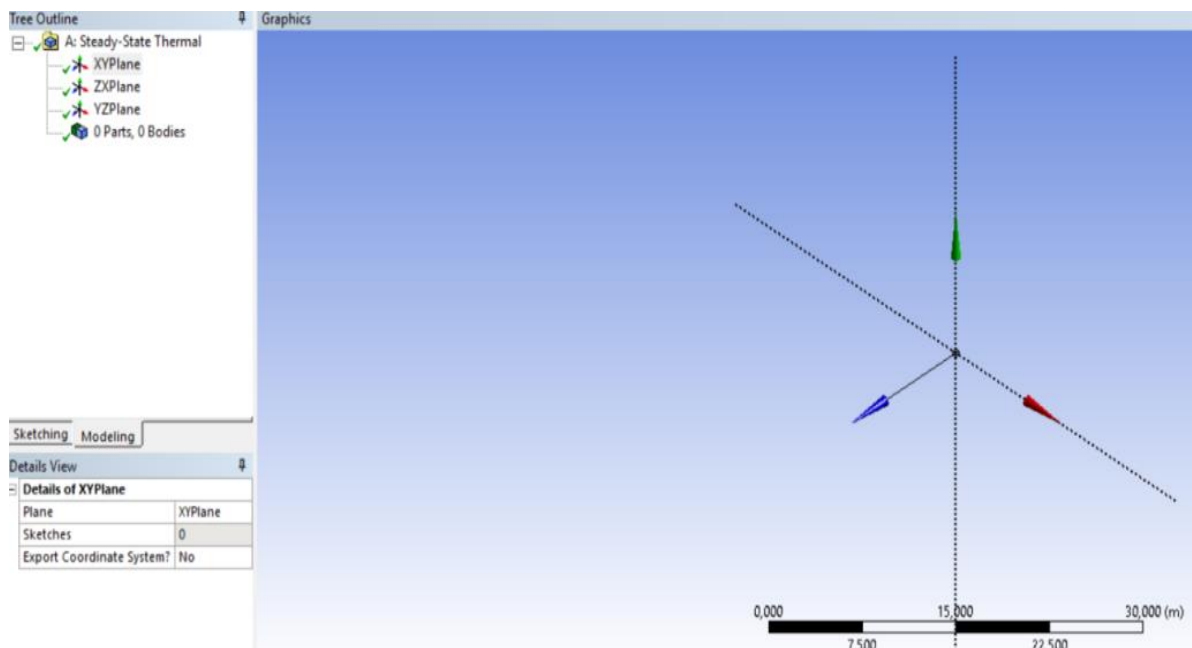


Figure 6.28 – Selecting a working plane

Next, go to the Sketching tab and use the Rectangle by 3 Point tool from Draw to create a basic sketch, the dimensions of which we will set using the Semi-Automatic tool from Dimensions (Figure 6.29).

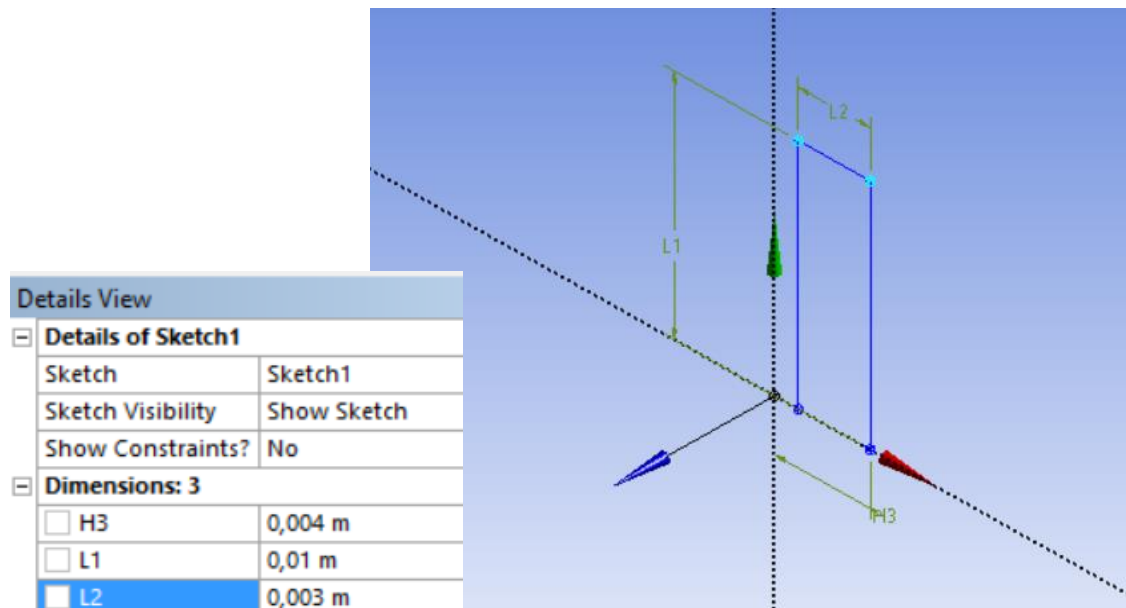


Figure 6.29 – Basic sketch

Using the Revolve operation, we obtain a body of revolution by rotating the base sketch around a given axis (Figure 6.30). The Axis option specifies the axis of rotation (in our case, the OY coordinate axis). The Angle parameter, which determines the angle of rotation, is set to 90°.

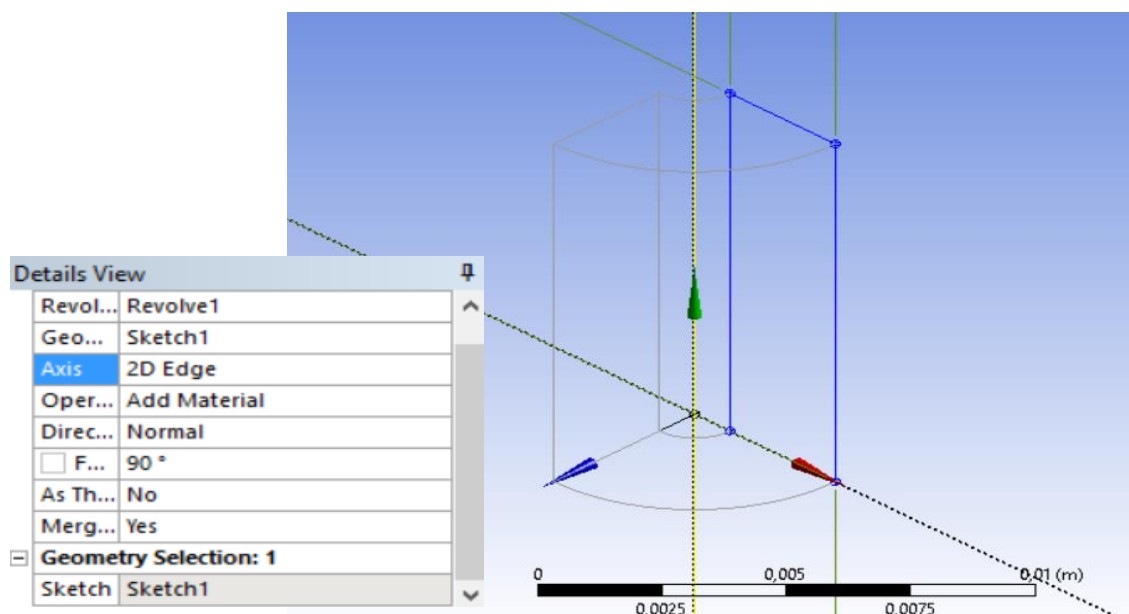


Figure 6.30 – Using the Revolve operation

After executing Generate, we will create a $\frac{1}{4}$ cylinder with a radius of 4 mm with an internal hole with a radius of 1 mm and a height of 10 mm (Figure 6.31).

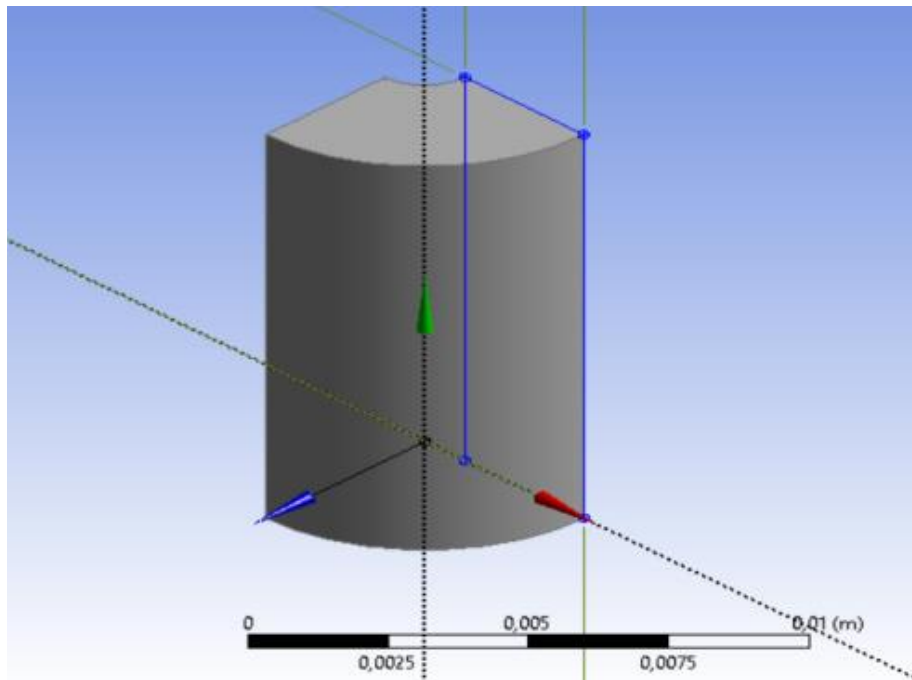


Figure 6.31 – Post-construction sample visualization

To generate a finite element mesh, right-click on the Model item in the project template and select Edit. Workbench Mechanical will be launched. After that, right-click on the Mesh item in the project tree and then Generate Mesh. The resulting FE mesh is shown in Figure 6.32.

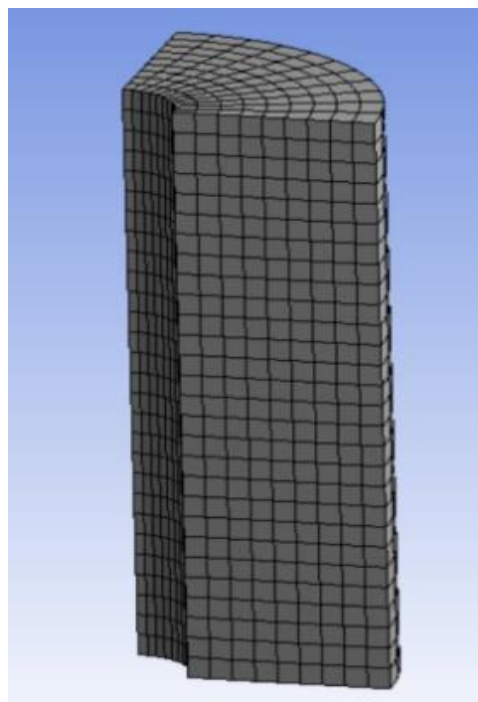


Figure 6.32 – Finite element model of the sample

It is necessary to set thermal loads in accordance with the task conditions. To do this, select the required face, right-click the Steady-State Thermal project tree item, then Insert – Convection. In the Details of Convection menu, set the film coefficient to 30,000 W/(m² °C) at a temperature of 300 °C. To set the heat generation per unit volume, select the appropriate volume, right-click the Steady-State Thermal project tree item, then Insert – Internal Heat Generation. In the Details of Internal Heat Generation menu, in Magnitude, set the heat generation value to 109 W/m³ (Figure 6.33).

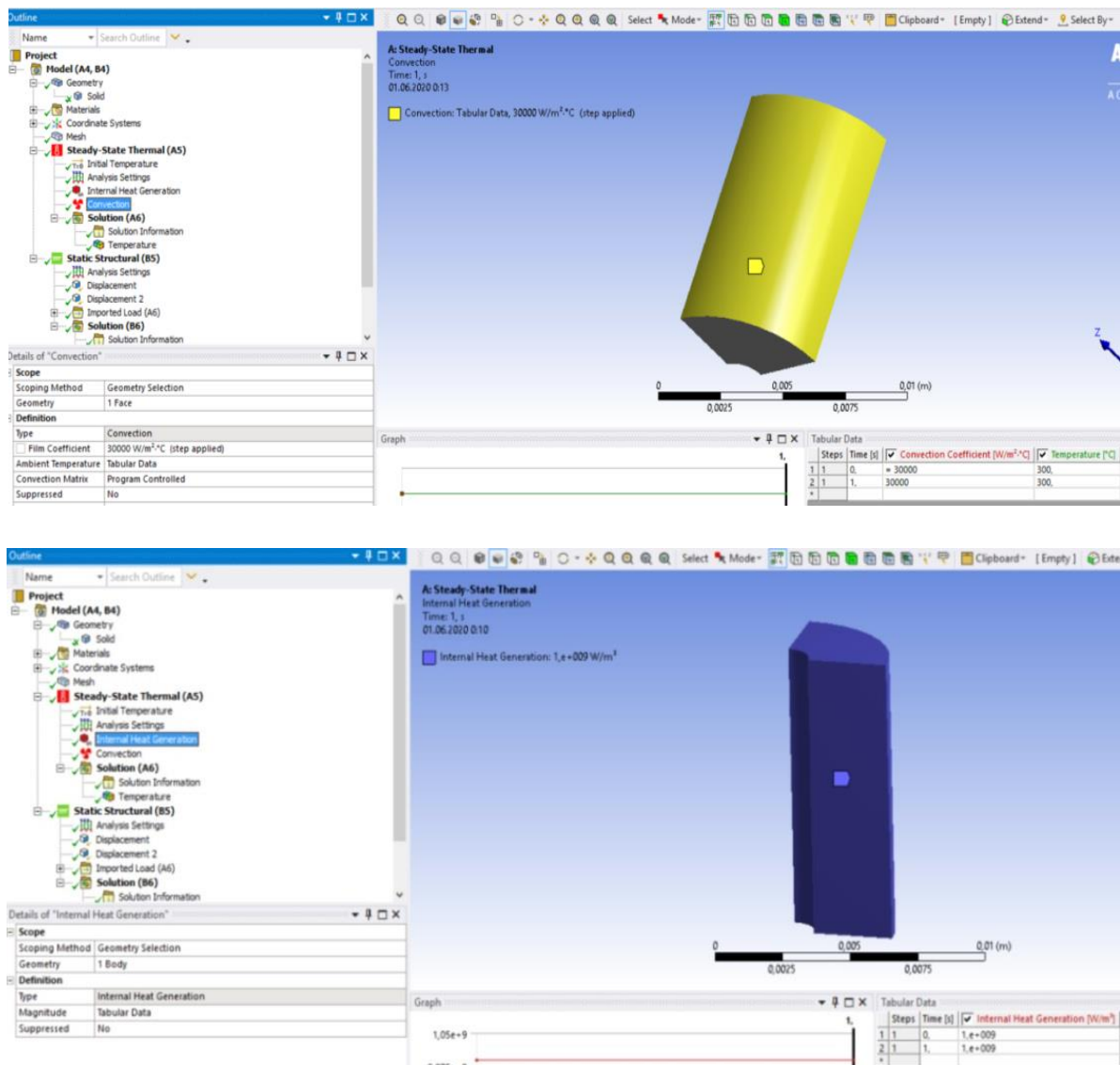


Figure 6.33 – Setting loads

It is necessary to set the boundary conditions for the strength analysis, and the transfer of the results of the solution of the thermal problem to the input of the solution of the strength problem is carried out automatically (see point 2).

Due to symmetry, only one quarter of the disk is modeled, and we fix the displacements along the OX and OZ axes of the corresponding faces of the model (Figure 6.34). Select the desired face, right-click on the Static Structural project tree item, then Insert – Displacement. In the Details of Displacement menu, we set the value of the corresponding component to 0.

To start the solution, you need to click the Solve button.

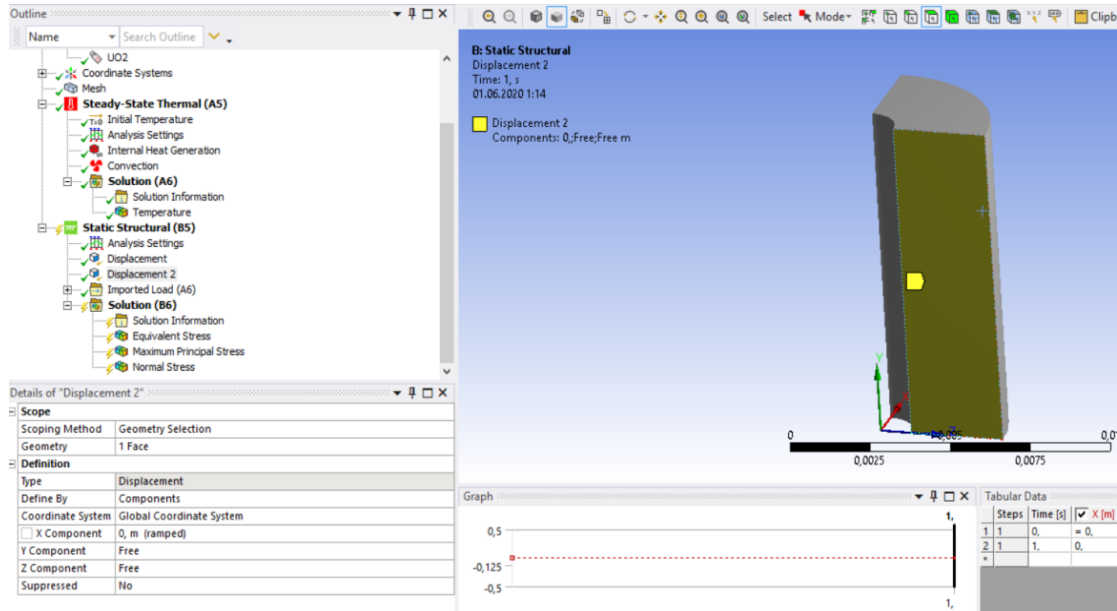


Figure 6.34 – Setting boundary conditions for strength analysis

We visualize the obtained solution by two parameters: Temperature (Figure 6.35) and Normal Stress (Figures 6.36–6.38). For this, select the corresponding items in the Solution menu in the Steady-State Thermal branch and in the Static Structural branch of the project tree.

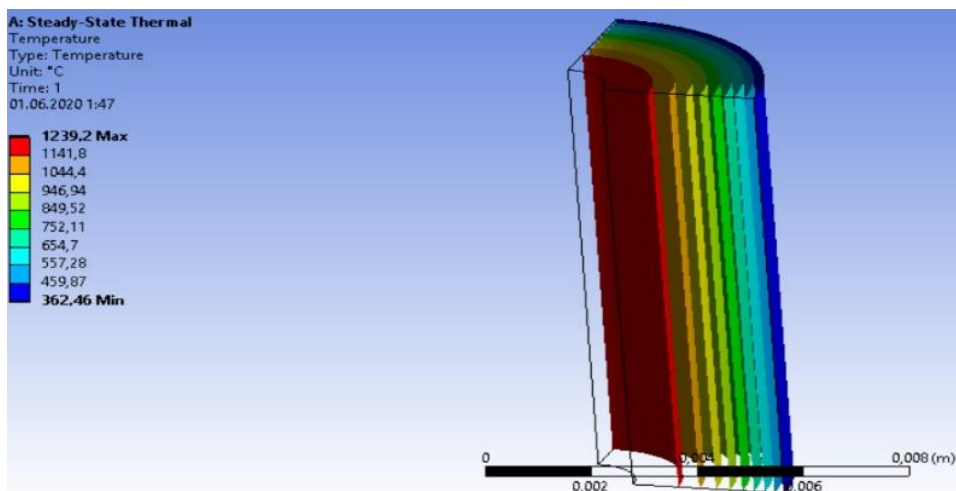


Figure 6.35 – Calculated temperature values

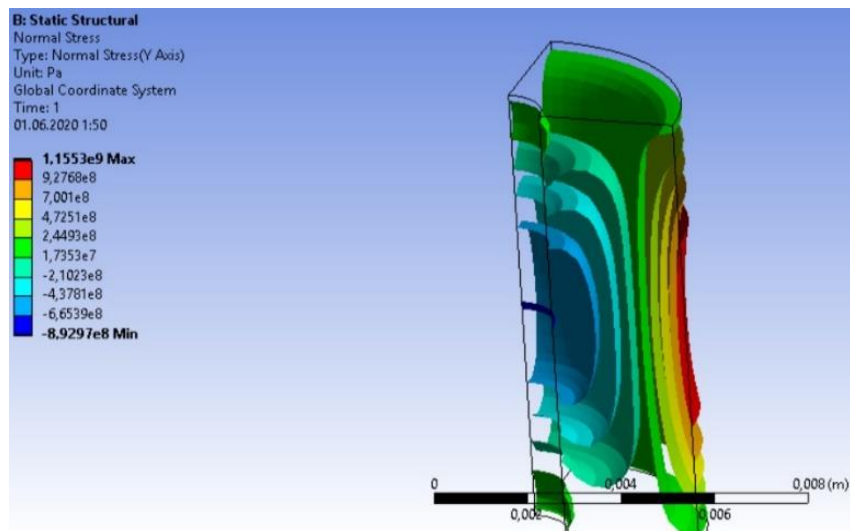


Figure 6.36 – Calculated values of stresses σ_{yy}

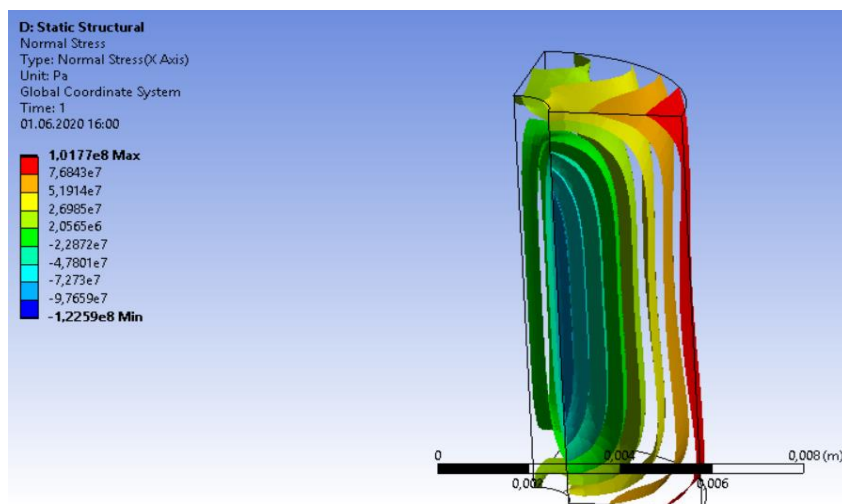


Figure 6.37 – Calculated values of stresses σ_{xx}

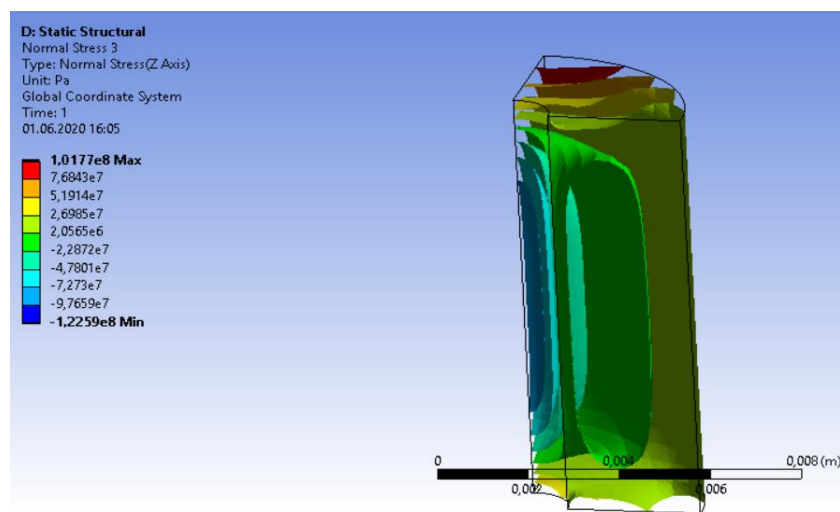


Figure 6.38 – Calculated values of stresses σ_{zz}

The data on the geometry of the sample, its geometric and finite element model, generated during the performance of the coupled static thermal strength analysis, are then used to solve the non-stationary problem of thermal conductivity with the subsequent solution of the problem of finding the thermoelastic stress fields for certain moments in time.

To implement this approach, in the Toolbox section, hold down the left mouse button on Transient Thermal and drag it to block A (Steady-State Thermal) of the Project Schematic window (activating the Engineering Data, Geometry and Model sections), release the left mouse button. As a result, a template of the corresponding project will appear in the Project Schematic window (Figure 6.39).

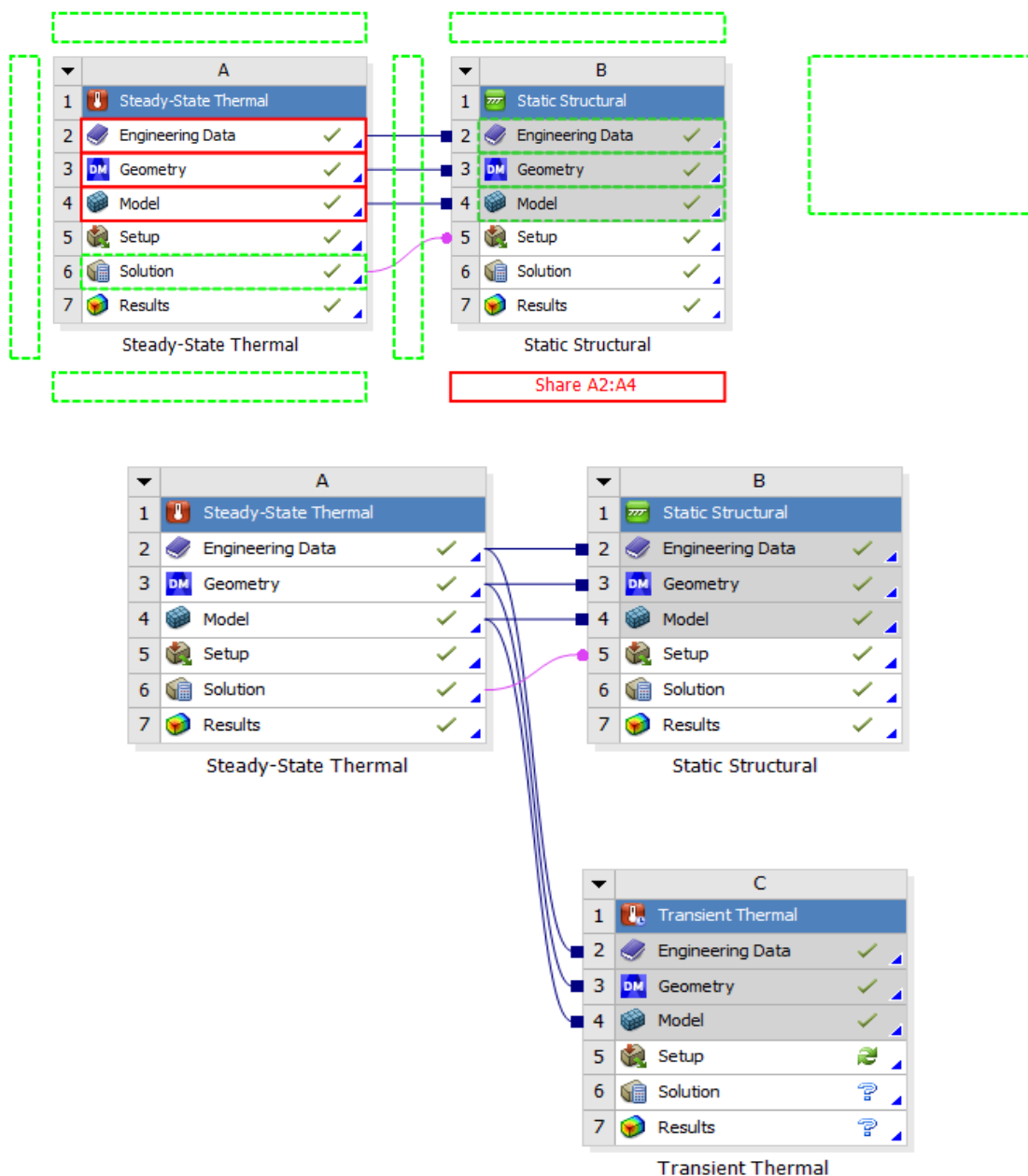


Figure 6.39 – Connection of a non-stationary thermal conductivity block to the project

Next, in the Toolbox section, hold down the left mouse button on Static Structural and drag it to the C (Transient Thermal) block of the Project Schematic window (activating the Engineering Data, Geometry, Model and Solution sections), and release the left mouse button. As a result, the project template shown in Figure 6.40 will appear in the Project Schematic window.

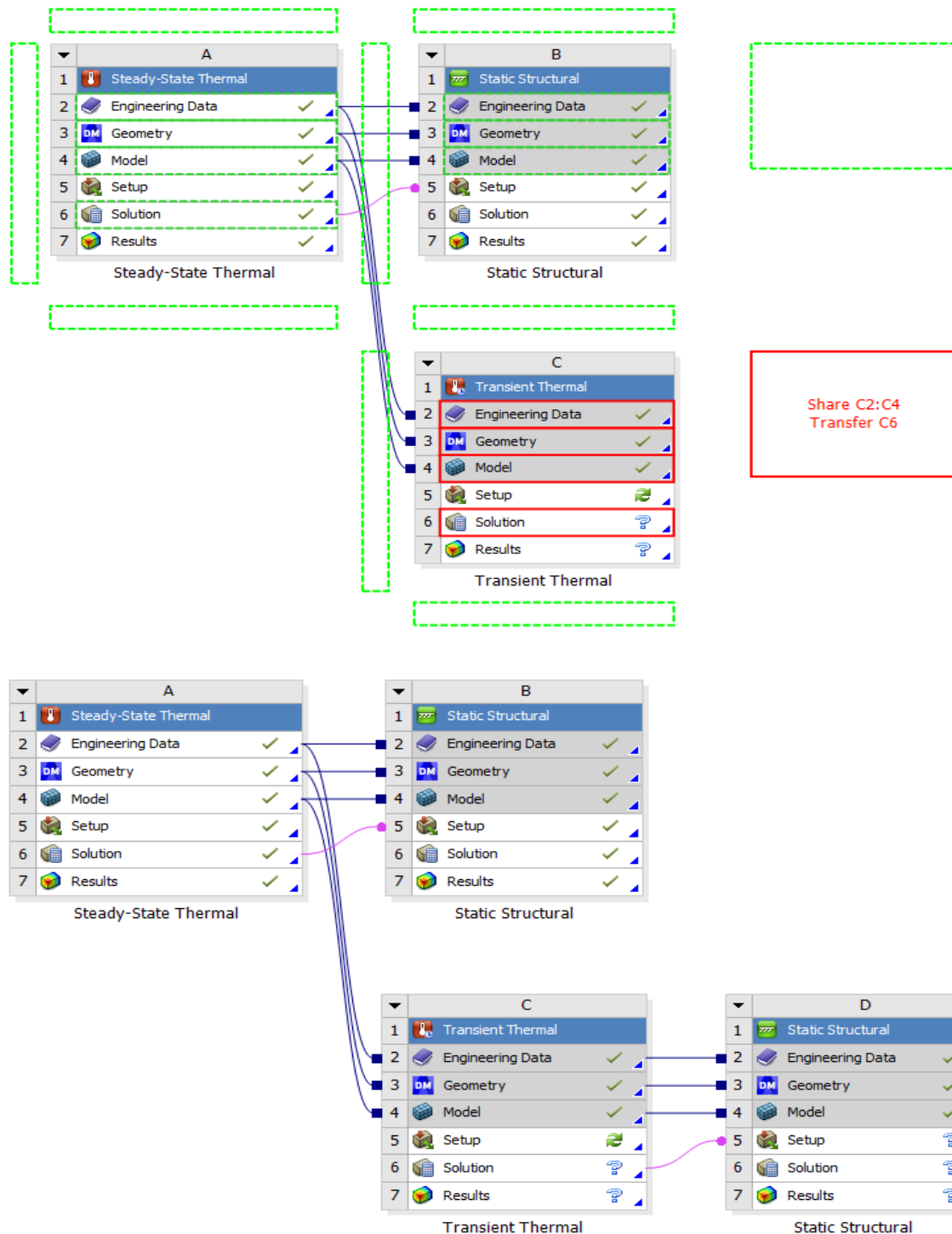


Figure 6.40 – Connecting the second block of stationary structural analysis to the project

To set the material properties considering their temperature dependence, right-click on Engineering Data and then click Edit. In the group of windows that appears, go to the Outlook of Schematic A2, B2, C2, D2: Engineering Data window, enter the name of the new material UO2_T and set the required properties, while for density, thermal conductivity and specific heat capacity we will set a tabular dependence of these properties on temperature (Figure 6.41).

Properties of Outline Row 4: UO2_T				
	A	B	C	D E
1	Property	Value	Unit	
2	Material Field Variables	Table		
3	Density	Tabular		
4	Scale	1		
5	Offset	0	kg m^{-3}	
6	Isotropic Secant Coefficient of Thermal Expansion			
7	Coefficient of Thermal Expansion	9,75E-06	$^{\circ}\text{C}^{-1}$	
8	Isotropic Elasticity			
9	Derive from	Young's Modulus and Poisson's Ratio		
10	Young's Modulus	1,75E+11	Pa	
11	Poisson's Ratio	0,316		
12	Bulk Modulus	1,5851E+11	Pa	
13	Shear Modulus	6,6489E+10	Pa	
14	Isotropic Thermal Conductivity	Tabular		
15	Scale	1		
16	Offset	0	$\text{W m}^{-1} \text{C}^{-1}$	
17	Specific Heat, C_p	Tabular		
18	Scale	1		
19	Offset	0	$\text{J kg}^{-1} \text{C}^{-1}$	

Table of Properties Row 3: Density

	A	B
1	Temperature (C)	Density (kg m^{-3})
2	25	10958
3	500	10798
4	1000	10627
5	2000	10168
6	3000	8716
*		

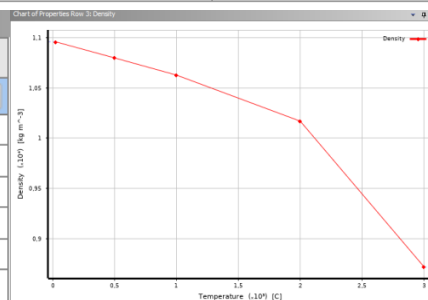


Table of Properties Row 17: Specific Heat, C_p

	A	B
1	Temperature (C)	Specific Heat ($\text{J kg}^{-1} \text{C}^{-1}$)
2	25	295
3	500	313
4	1000	319
5	2000	433
6	3000	460
*		

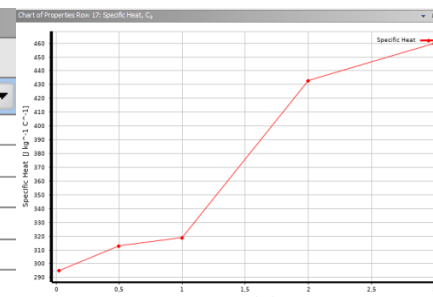


Table of Properties Row 14: Isotropic Thermal Conductivity

	A	B
1	Temperature (C)	Thermal Conductivity ($\text{W m}^{-1} \text{C}^{-1}$)
2	25	7,6
3	500	4,27
4	1000	2,79
5	2000	2,12
6	3000	4,62
*		

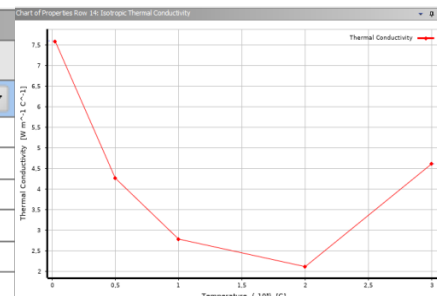


Figure 6.41 – Properties of uranium dioxide as a function of temperature

Next, in the Outline window of the Workbench Mechanical graphical interface, select the Model – Geometry item from the tree structure of the calculation model. In the Details of «Solid» window, in Material – Assignment, select UO2_T (Figure 6.42).

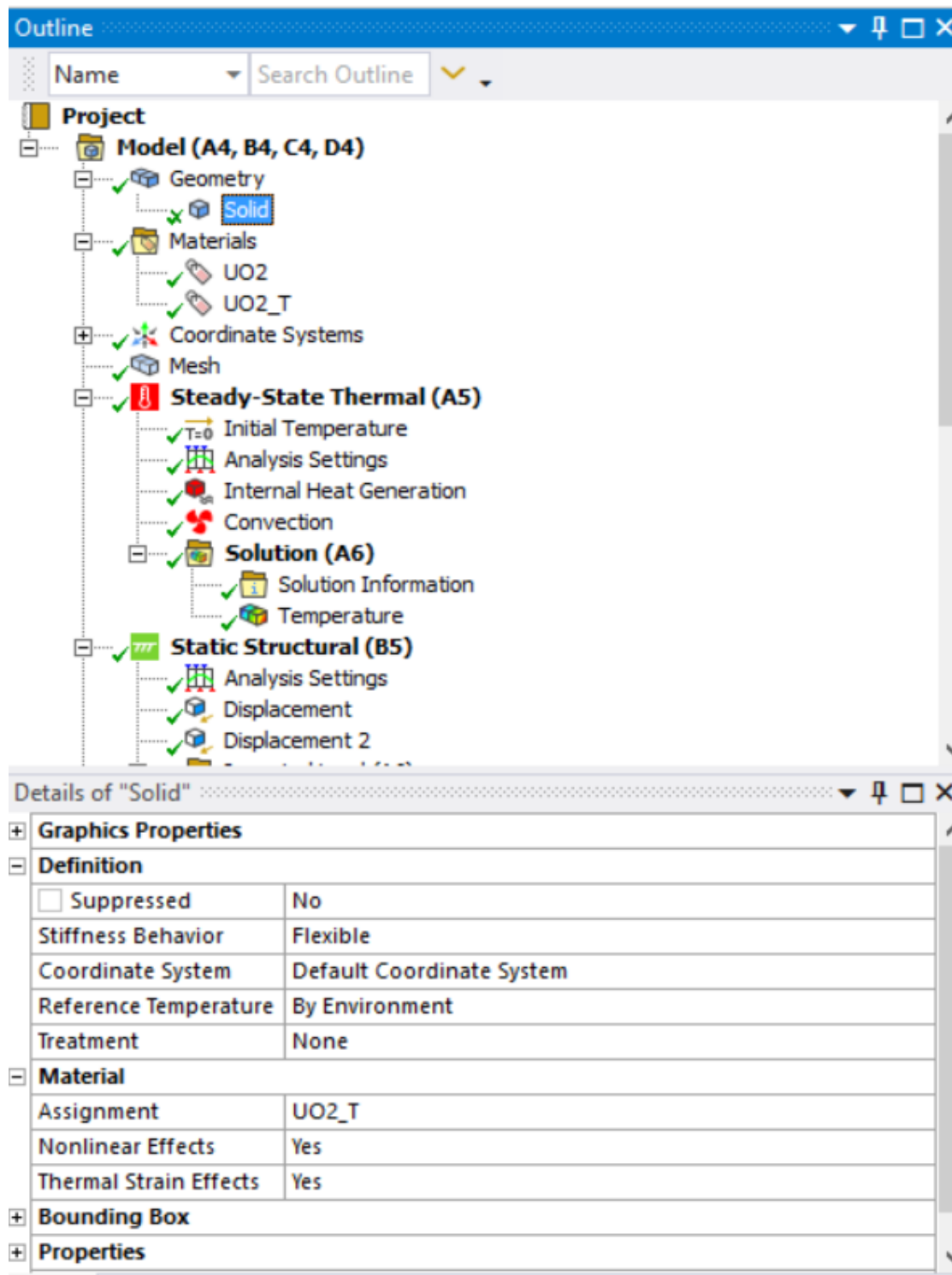


Figure 6.42 – Correlation of uranium dioxide properties with respect to temperature dependency

Having selected the corresponding face, we select the Transient Thermal project tree item with the right mouse button, then, having selected the Insert – Convection item and in the Details of Convection menu, we set the film coefficient value equal to 30,000 W/(m² °C) at an ambient temperature of 300 °C. To set the heat generation in the volume, having selected the corresponding volume, we select the Steady-State Thermal project tree item with the right mouse button and set the heat loads. To do this, we select the Insert – Internal Heat Generation item and in the Details of Internal Heat Generation menu in the Magnitude item, we set the heat generation value equal to 109 W/m³ (similar to item 6).

Going to the Initial Temperature section, we define the initial temperature value in the Definition menu as 22 °C (Figure 6.43).

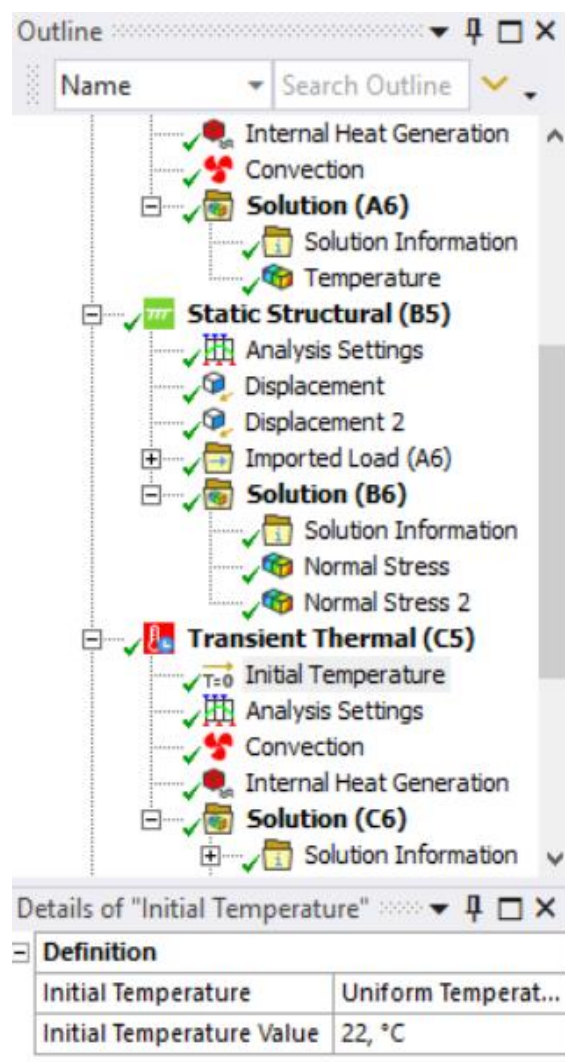


Figure 6.43 – Setting the initial temperature

In the Analysis Settings section, we set the parameters for non-stationary analysis, including the heating process analysis time, equal to 30 s (Figure 6.44).

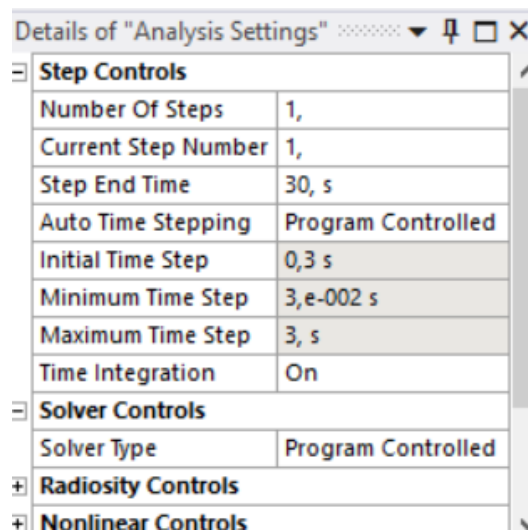


Figure 6.44 – Setting parameters for non-stationary analysis

We shall establish the boundary conditions for the strength analysis. Select the corresponding face, right-click the Static Structural project tree item, and then Insert – Displacement. In the Details of Displacement menu, set the value of the corresponding component to 0 (similar to item 7).

To start the solution, you need to click the Solve button.

We visualize the obtained solution by several parameters (Figures 6.45–6.47).

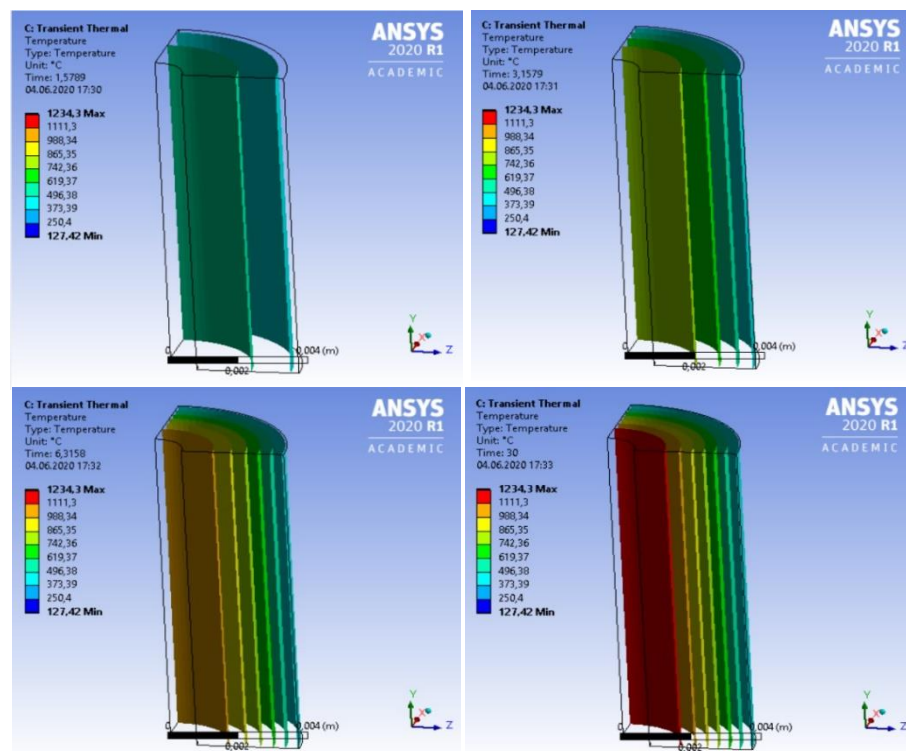


Figure 6.45 – Calculated temperature values at different points in time

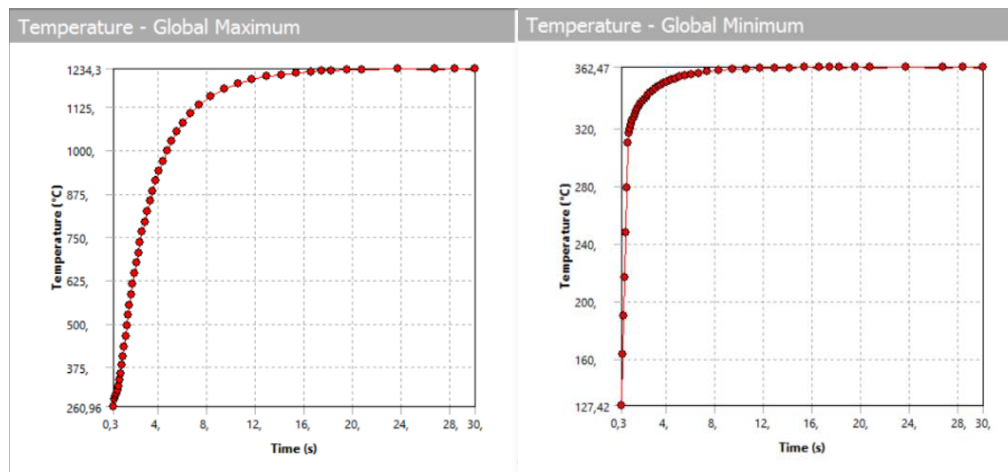


Figure 6.46 – Calculated maximum and minimum temperature values at different points in time

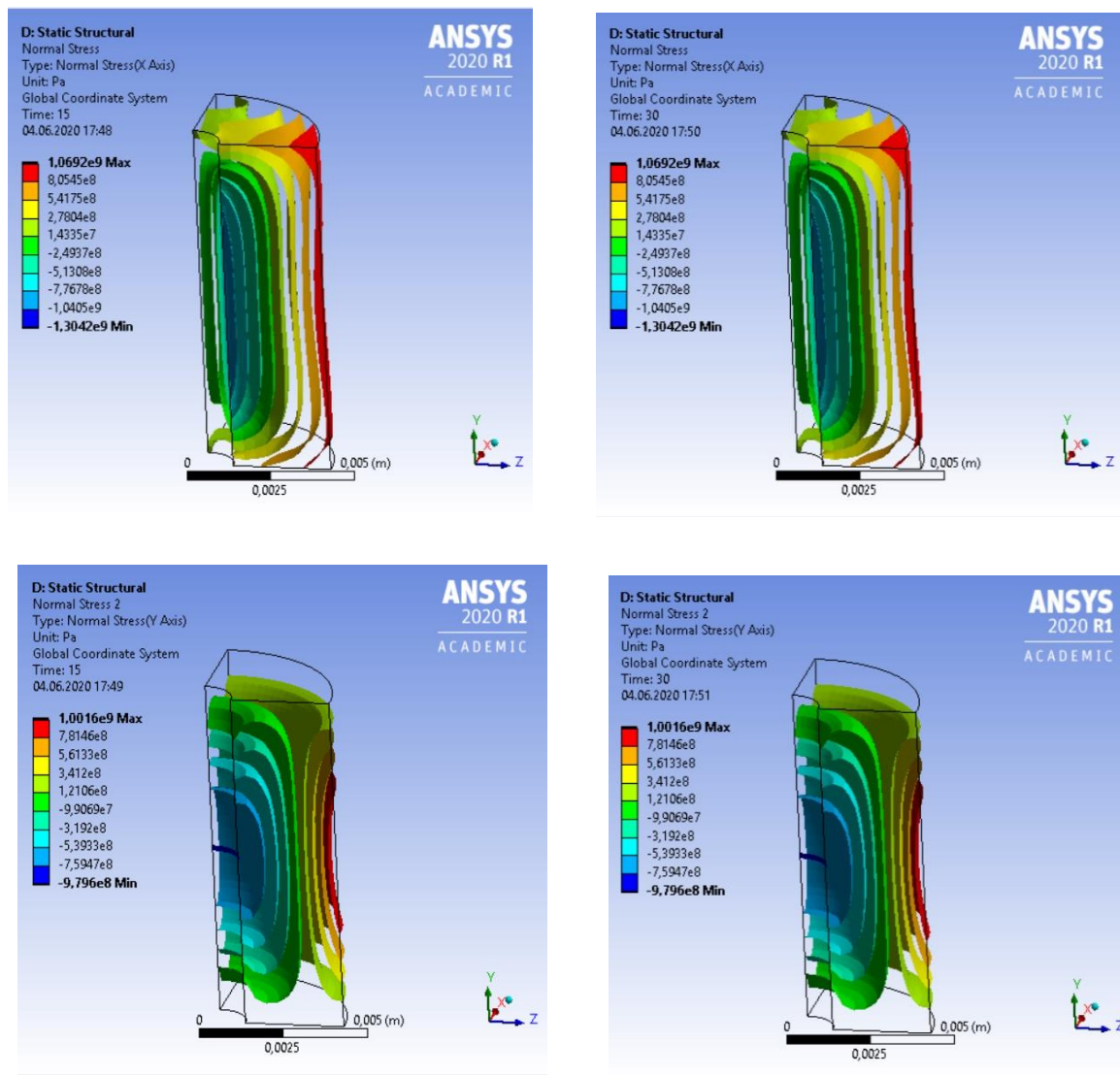


Figure 6.47 – Calculated stresses σ_{xx} , σ_{yy} at different moments in time

7 IMPLEMENTATION EXAMPLES FOR ANSYS DESIGNXPLORER MODULE

Problem 7.1 “What-If” Parametric Study in ANSYS Workbench

7.1.1 Problem statement

This example introduces the capabilities of Ansys Workbench for performing parametric “What if?” studies. These studies are executed using manually defined design points without requiring DesignXplorer. While “What-If?” studies deliver preliminary insights, their results can inform subsequent advanced analyses in DesignXplorer.

The tow hitch created in ANSYS DesignModeler consists of three parts (Figure 7.1). Figure 7.1 also shows the loads and constraints applied to the tow hitch.

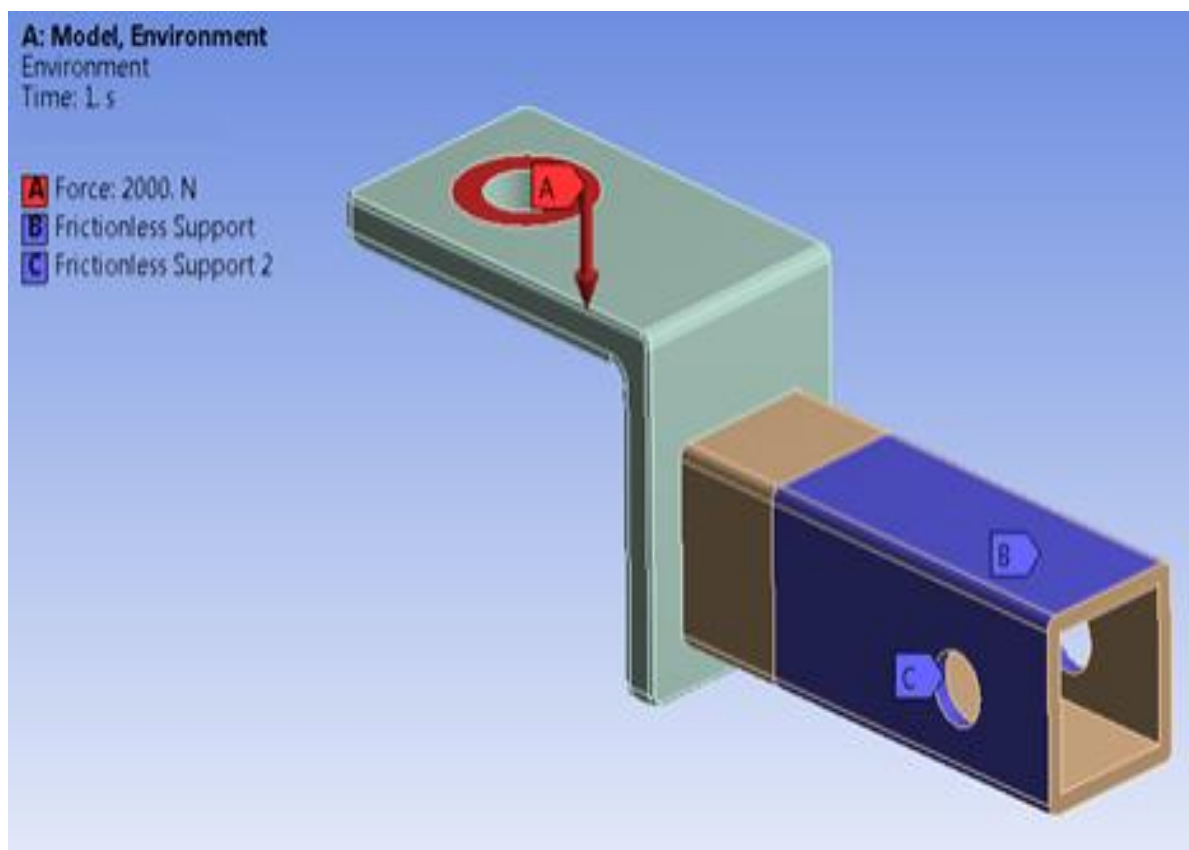


Figure 7.1 – Towing device

Figure 7.2 illustrates the geometric dimensions of the towing device, which will be used as input parameters when performing the work. In this case, the values of the towing device mass, stresses and deformations formed when applying the load will be used as output parameters.

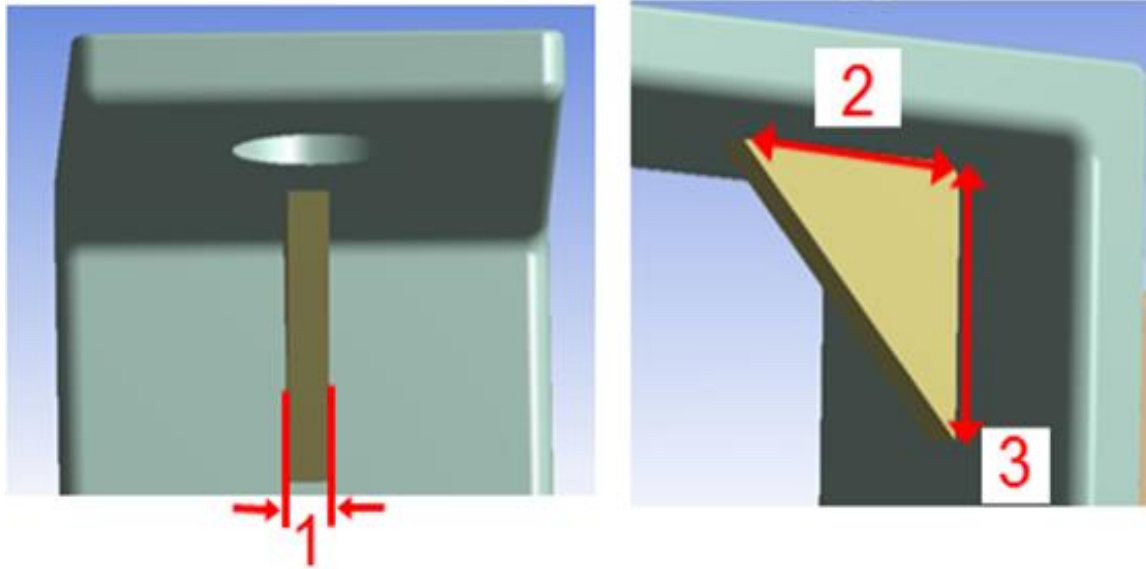


Figure 7.2 – Geometric parameters of the towing device, used as input parameters of the model

7.1.2 Implementation of the problem using the Ansys Workbench graphical interface

1 The solution to the problem begins with launching Ansys Workbench, which is executed from the main menu Start → Programs → Ansys 2020R1 → Workbench 2020R1.

2 Select File → Open. Then, in the window that appears, select the file named receiver.wbpz. Save the project with the name receiver and the extension wbpj. As a result, an analytical block will appear in the Project Schematic window, which we will use as a template for our model (Figure 7.3).

3 Double-click the Geometry cell. The towing device model will appear in the DesignModeler window that opens. In the Tree Outline window, select Gusset. Next, in the Details View, check the FD1, Depth box to make it an input parameter. After that, a dialog box for specifying the parameter name will appear. Specify the name thick_ds (Figure 7.4).

4 In the Tree Outline window, select YZPlane. In YZPlane, select Sketch4, in the Dimensions section, as in point 3, set the flags on H6 (name – horiz_ds) and V7 (name – vert_ds) (Figures 7.5, 7.6). Close DesignModeler.

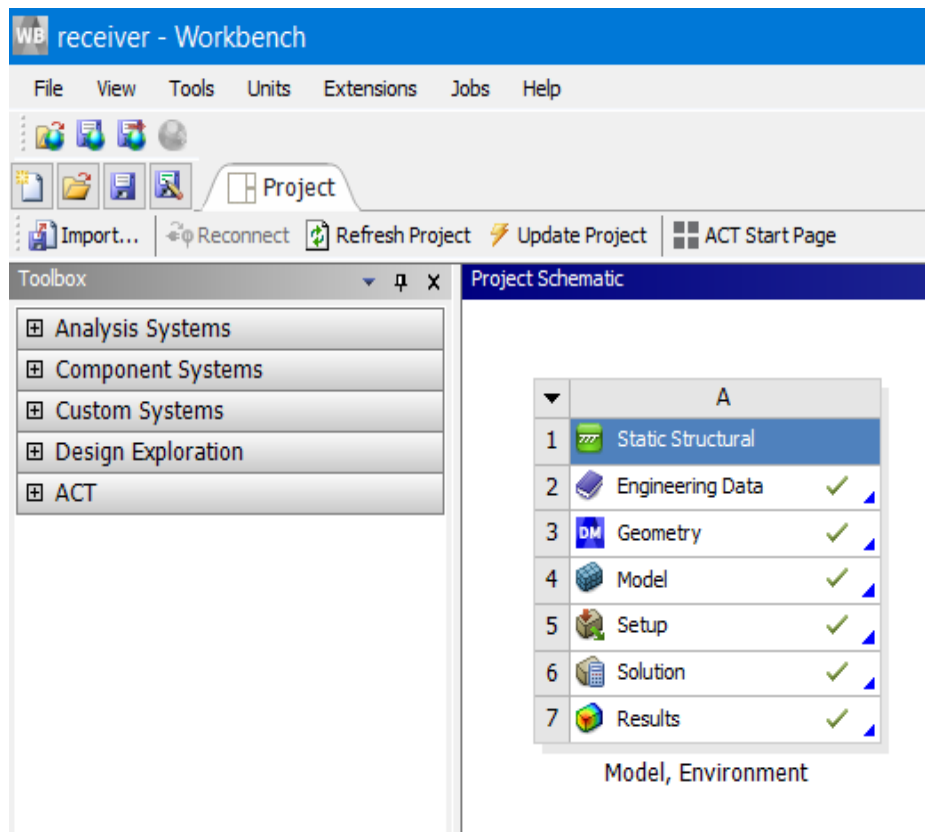


Figure 7.3 – Model import results

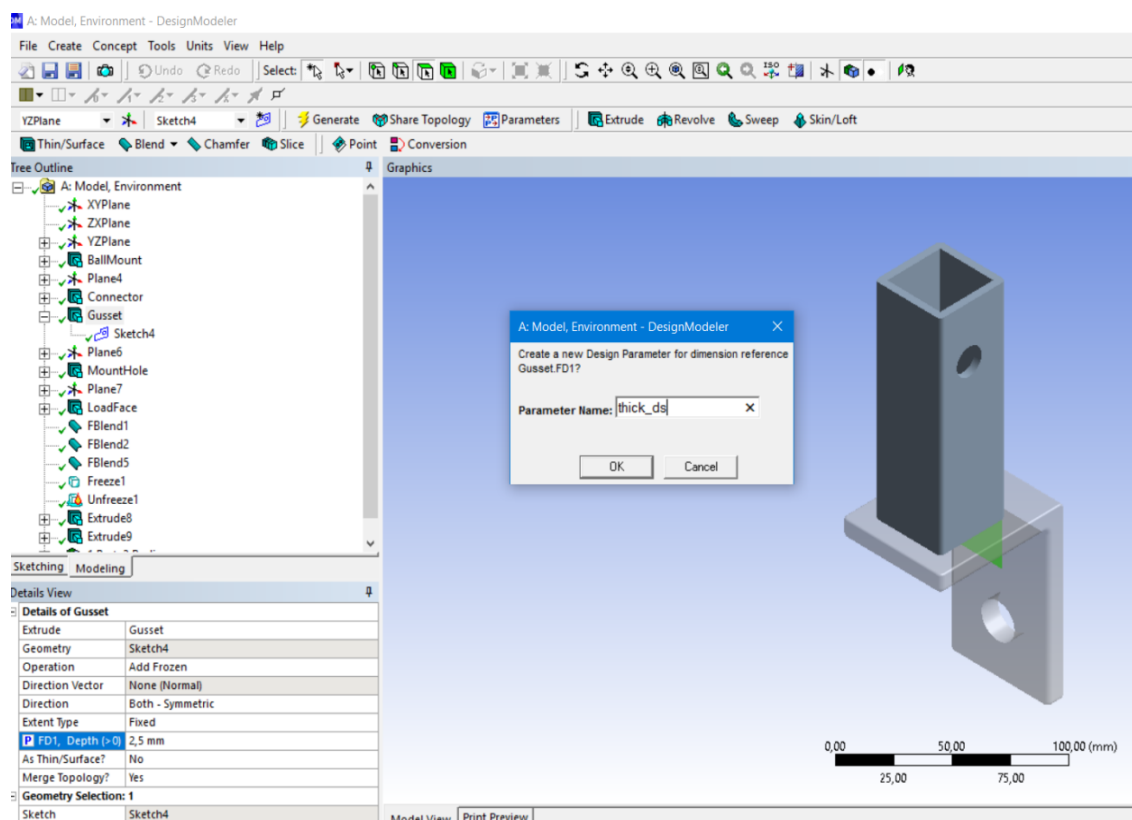


Figure 7.4 – Setting the thick_ds parameter

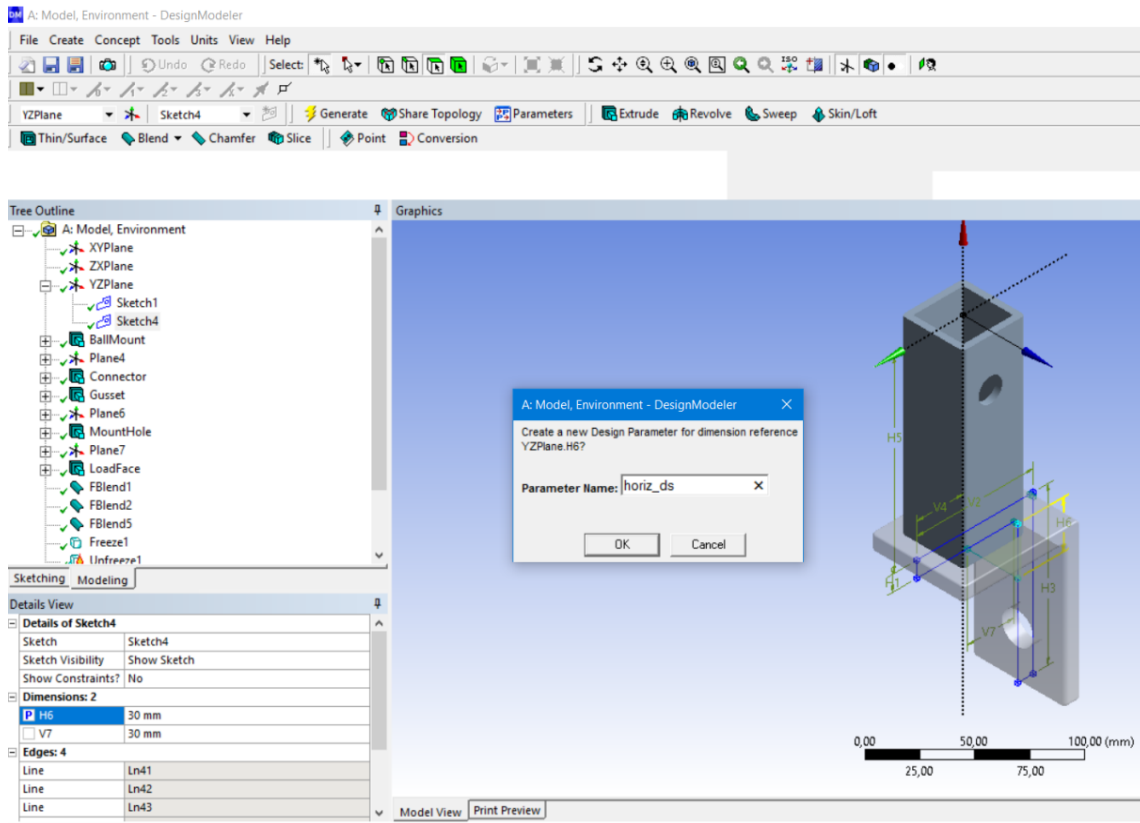


Figure 7.5 – Setting the horiz_ds parameter

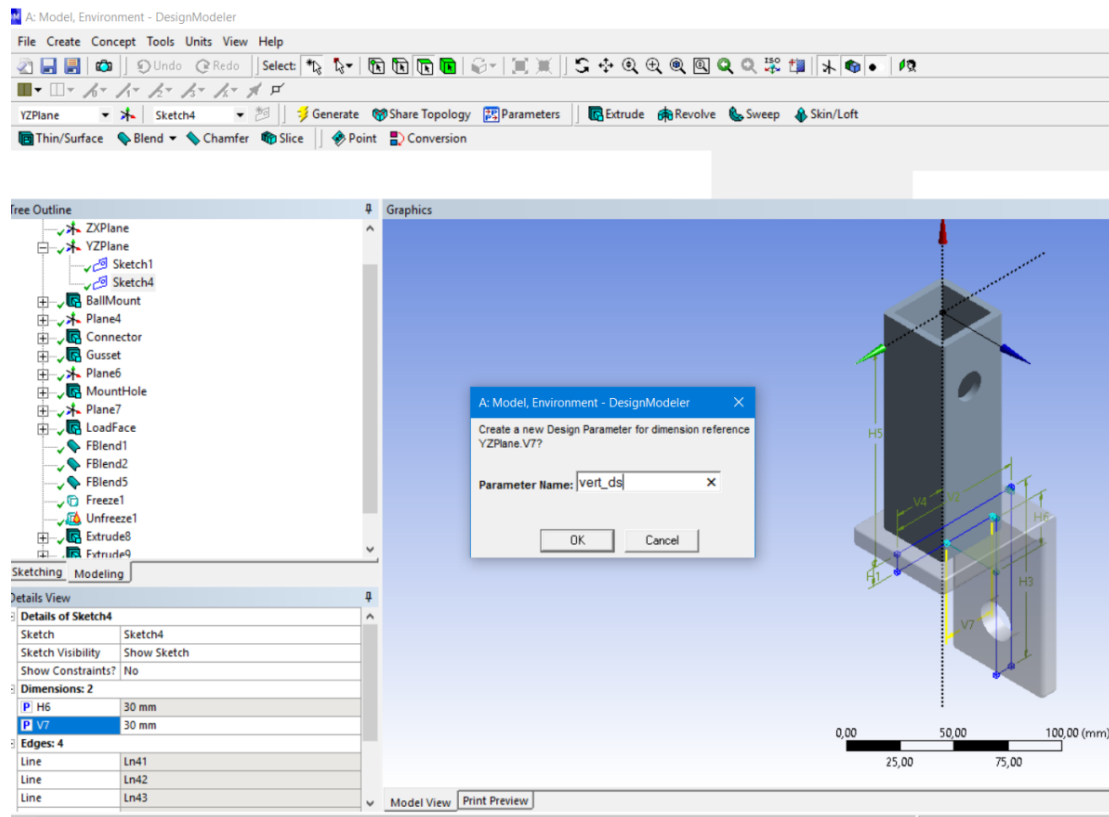


Figure 7.6 – Setting the vert_ds parameter

5 The Project Schematics now includes a Parameter Set panel. An arrow coming out of the Parameter Set and connecting to the Parameters cell in the Static Structural indicates that the input parameters have been defined (Figure 7.7). Double-clicking on the Parameters cell opens the corresponding window, in which the defined input parameters of the project are visible (Figure 7.8).

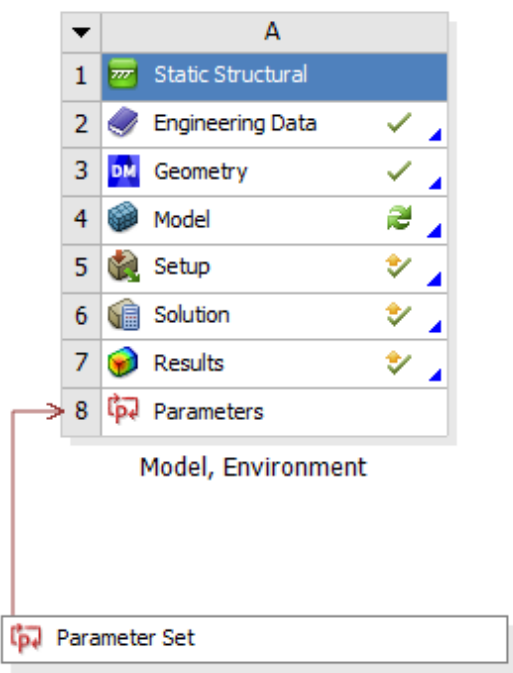


Figure 7.7 – Project diagram with parameters panel

Outline of Schematic A8: Parameters				
	A	B	C	D
1	ID	Parameter Name	Value	Unit
2	[-] Input Parameters			
3	[-] [Static Structural] Model, Environment (A1)			
4	[P] P1	thick_ds	2,5	mm [v]
5	[P] P2	horiz_ds	30	mm [v]
6	[P] P3	vert_ds	30	mm [v]
*	[P] New input parameter	New name	New expression	
8	[-] Output Parameters			
*	[P] New output parameter		New expression	
10	Charts			

Figure 7.8 – Project input parameters

6 Double-click Setup (Project Schematic) (Figure 7.7). Depending on the ANSYS version, a dialog box may or may not appear indicating that the upstream data must be re-read. If this dialog box appears, click Yes to open the Mechanical window. If the confirmation window does not appear, the Mechanical window opens automatically (Figure 7.9).

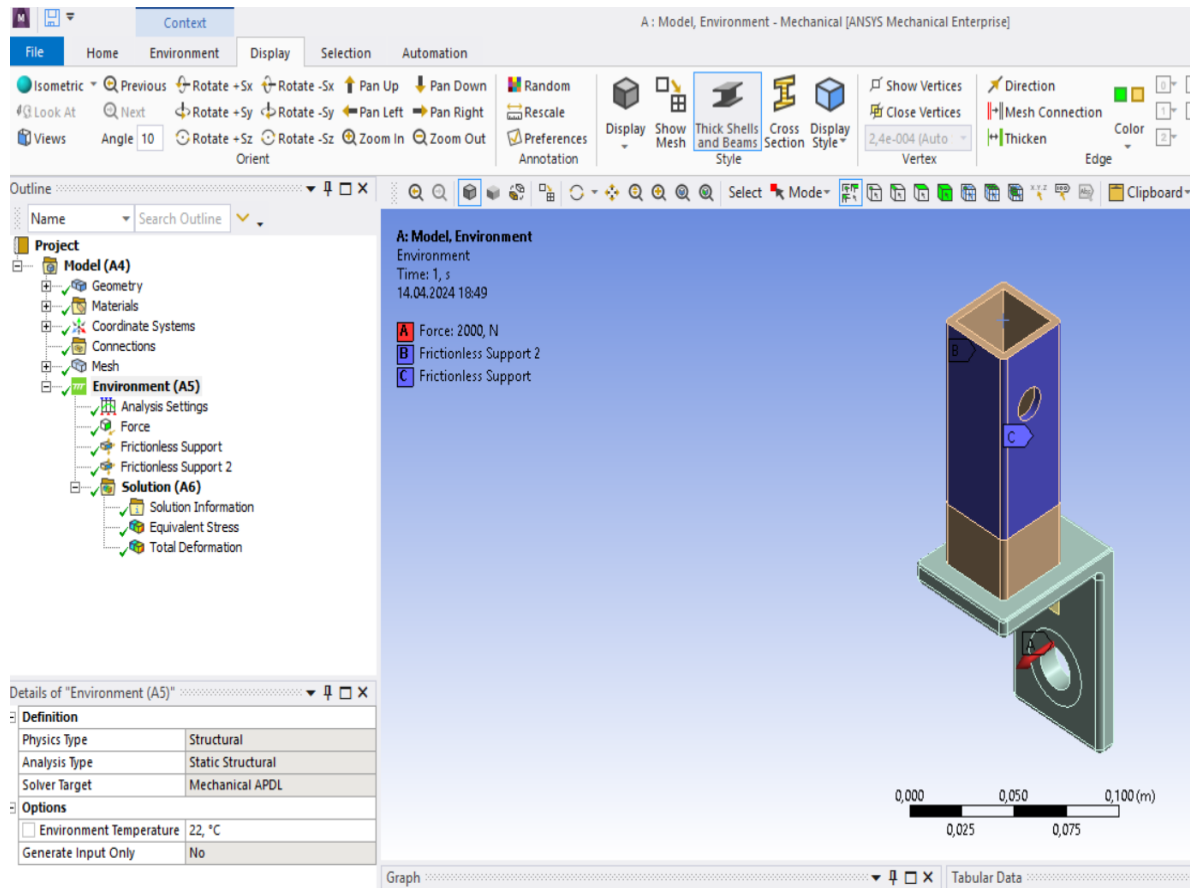


Figure 7.9 – Workbench Mechanical graphical interface

7 Next, in the Outline window of the Workbench Mechanical graphical interface, select the Model → Geometry item from the tree structure of the calculation model. In the Details of “Geometry” window, in the Properties tab, check the Mass cell (Figure 7.10).

8 In the Outline window, find Solution (A6) and select Equivalent Stress. Next, in the Details section, under the Results cell, check the Maximum box to define the maximum equivalent stress as the output parameter (Figure 7.11).

9 In the same Solution section, select Total Deformation and also check the Maximum cell to define the maximum deformations as the output parameter (Figure 7.12).

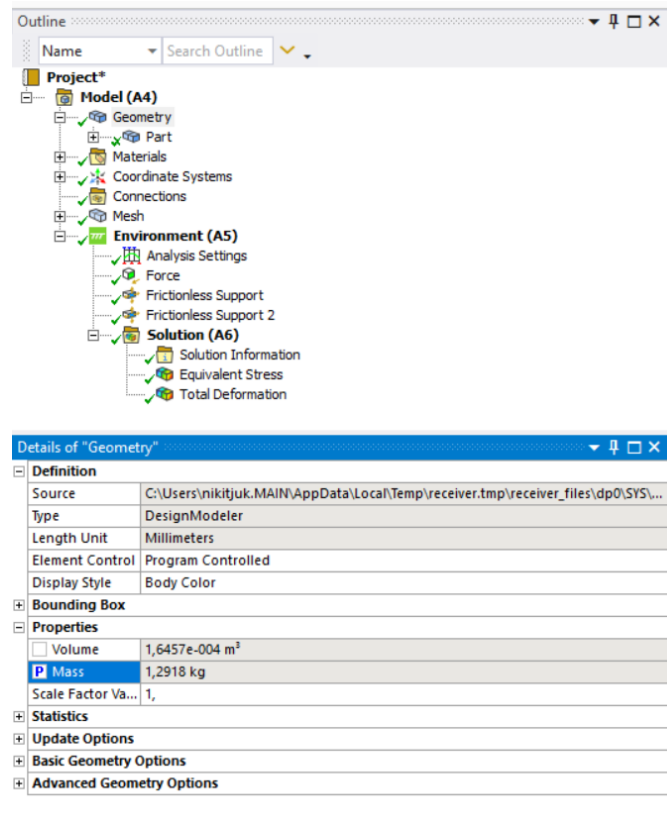


Figure 7.10 – Defining mass as an output parameter

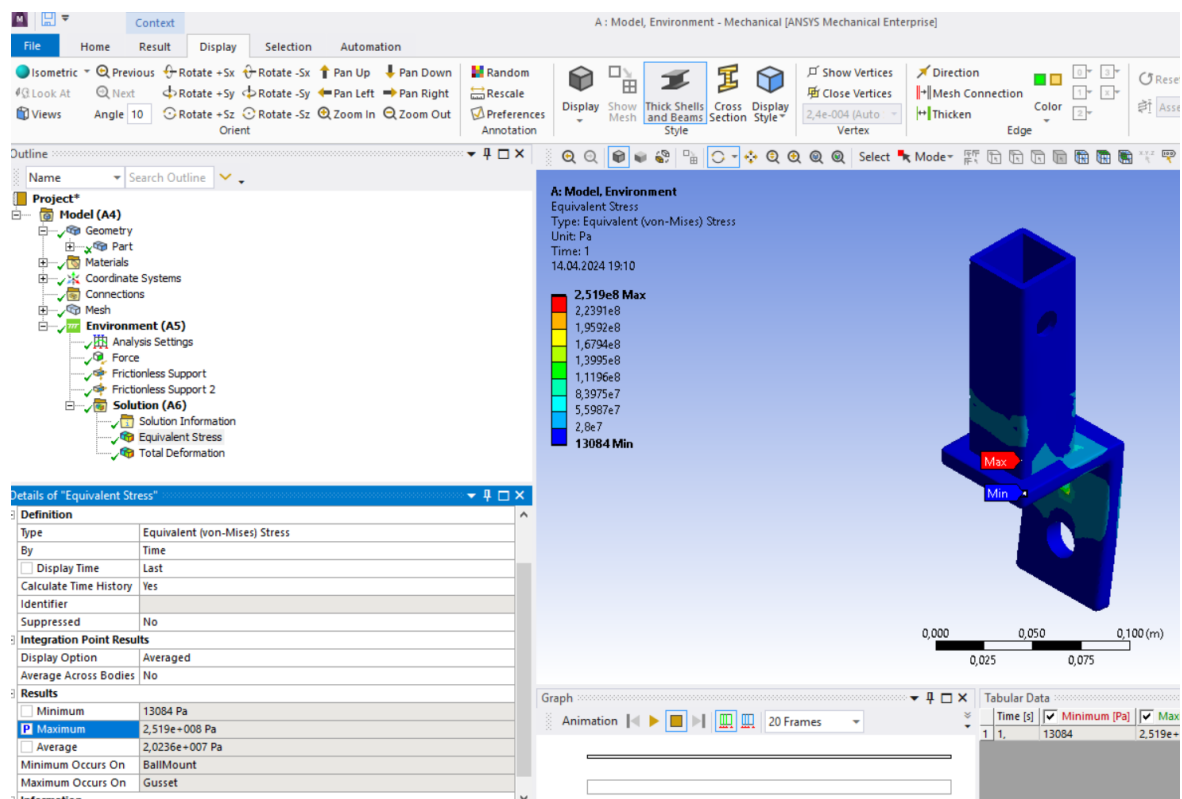


Figure 7.11 – Determination of maximum equivalent stresses as an output parameter

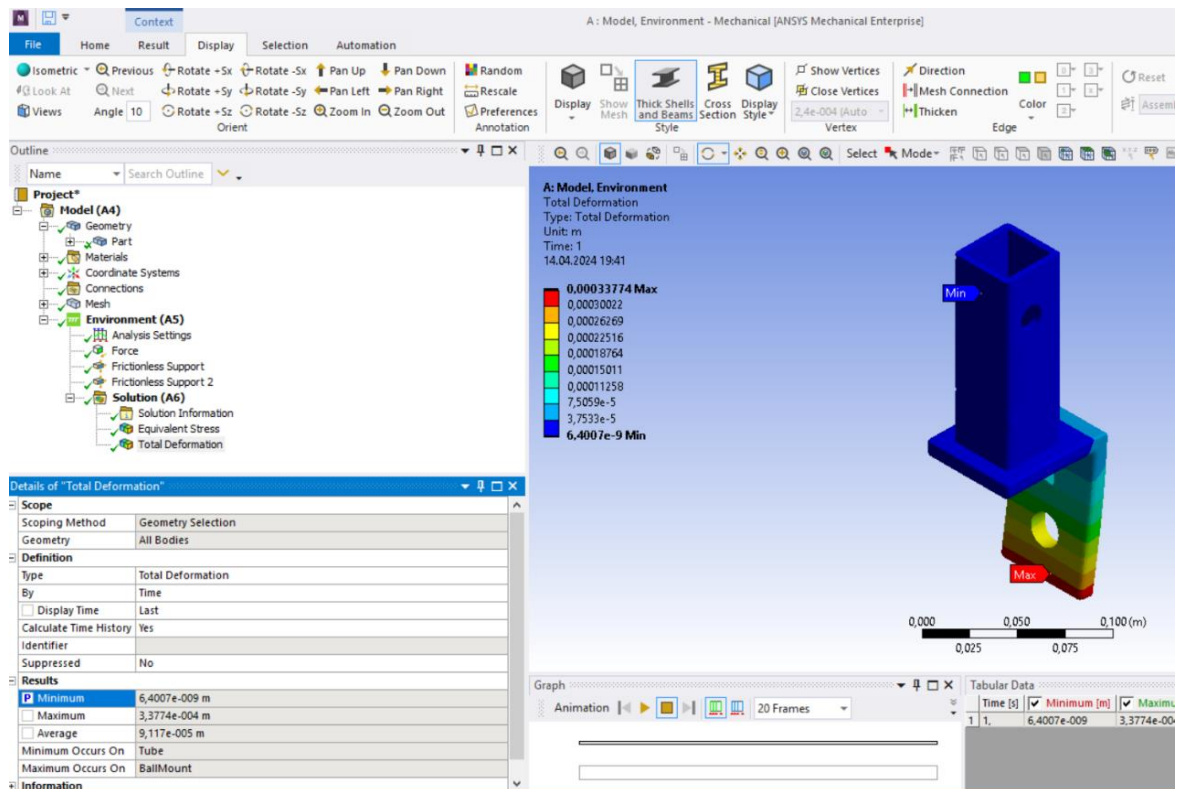


Figure 7.12 – Determining maximum deformations as an output parameter

10 We exit Mechanical and update the project in Project Schematic (Update Project). Double-click on the Parameters cell and open the corresponding window, displaying all defined input/output parameters and generated design points (Figures 7.13, 7.14).

Outline of All Parameters				
	A	B	C	D
1	ID	Parameter Name	Value	Unit
2	Input Parameters			
3	Model, Environment (A1)			
4	P1	thick_ds	2,5	mm
5	P2	horiz_ds	30	mm
6	P3	vert_ds	30	mm
*	New input parameter	New name	New expression	
8	Output Parameters			
9	Model, Environment (A1)			
10	P4	Geometry Mass	1,2918	kg
11	P5	Equivalent Stress Maximum	2,519E+08	Pa
12	P7	Total Deformation Maximum	0,00033774	m
*	New output parameter		New expression	
14	Charts			

Figure 7.13 – Input and output parameters of the project

Table of Design Points										
	A	B	C	D	E	F	G	H	I	J
1	Name ▾	P1 - thick_ds ▾	P2 - horiz_ds ▾	P3 - vert_ds ▾	P4 - Geometry Mass ▾	P5 - Equivalent Stress Maximum ▾	P7 - Total Deformation Maximum ▾	Retain	Retained Data	Note ▾
2	Units	mm ▾	mm ▾	mm ▾	kg	Pa	m			
3	DP 0 (Current)	2,5	30	30	1,2918	2,519E+08	0,00033774	☒	✓	
*								☐		

Figure 7.14 – Generated project point

11 Add 3 more points (RMB → Duplicate Design Point) (Figure 7.15).

Table of Design Points							
	A	B	C	D	E	F	G
1	Name ▾	P1 - thick_ds ▾	P2 - horiz_ds ▾	P3 - vert_ds ▾	P4 - Geometry Mass ▾	P5 - Equivalent Stress Maximum ▾	P7 - Total Deformation Maximum ▾
2	Units	mm ▾	mm ▾	mm ▾	kg	Pa	m
3	DP 0 (Current)	2,5	30	30	1,2918	2,519E+08	0,00033774
4	DP 1	2,25	27	33	⚡	⚡	⚡
5	DP 2	2,75	33	30	⚡	⚡	⚡
6	DP 3	2,5	33	33	⚡	⚡	⚡
*							

Figure 7.15 – Adding 3 project points

12 On the toolbar, click Update All Design Points. When the update is complete, the calculation results will appear in the table (Figure 7.16).

Table of Design Points							
	A	B	C	D	E	F	G
1	Name ▾	P1 - thick_ds ▾	P2 - horiz_ds ▾	P3 - vert_ds ▾	P4 - Geometry Mass ▾	P5 - Equivalent Stress Maximum ▾	P7 - Total Deformation Maximum ▾
2	Units	mm ▾	mm ▾	mm ▾	kg	Pa	m
3	DP 0 (Current)	2,5	30	30	1,2918	2,519E+08	0,00033774
4	DP 1	2,25	27	33	1,2896	2,9768E+08	0,00034848
5	DP 2	2,75	33	30	1,296	2,6407E+08	0,00031457
6	DP 3	2,5	33	33	1,2958	2,2366E+08	0,00030356
*							

Figure 7.16 – Calculation results

13 Let us add a chart that displays the relationship between the selected parameter and each point in the project. If you do not see the Chart panel in the lower right corner of the work window, select View → Chart. In the Outline panel under Output Parameters, select P7 – Total Deformation (maybe a different number). There are currently no charts below in the Charts section.

14 If Toolbox is not open in the working window, select View → Toolbox and open it.

In the Toolbox, double-click on Design Points vs P7 to add the corresponding diagram.

15 In the Outline window, under Charts, select Parameter Chart 0. In the Properties panel, X-Axis (Bottom), set Design Points and Y-Axis (Left), select P7 – Total Deformation Maximum (Figure 7.17).

Properties of Outline A14: 0		
	A	B
1	Property	Value
2	Parameter Chart: General	
3	Exclude Current Design point	☐
4	X-Axis (Bottom)	Design Points
5	X-Axis (Top)	
6	Y-Axis (Left)	P7 - Total Deformation Maximum
7	Y-Axis (Right)	

Figure 7.17 – Setting up chart parameters

The Graph panel will now display a graph. The project points are located on the X-axis at the bottom of the graph, and the maximum strain value is displayed on the Y-axis on the left side of the graph (Figure 7.18).

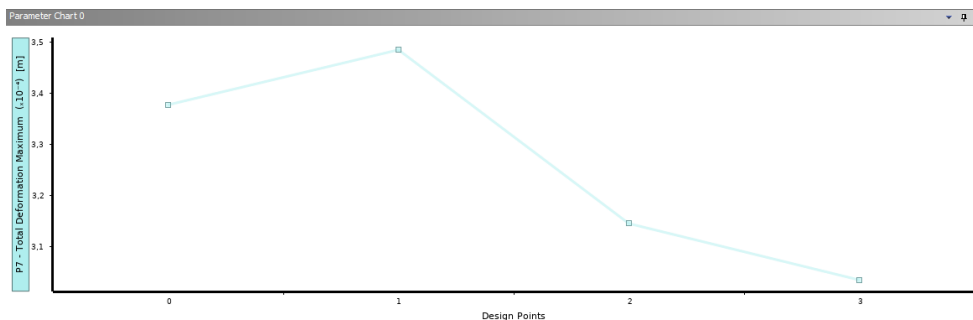


Figure 7.18 – Displaying the Graph panel

By additionally selecting Properties Y-Axis (Right) P4 – Geometry Mass on the properties panel, we display the mass values of the part along the Y-axis in the right part of the diagram (Figure 7.19).

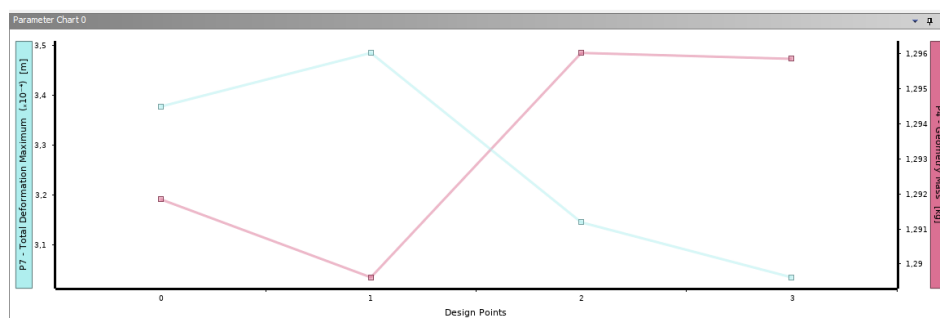


Figure 7.19 – Displaying the Graph panel

16 Let us add a Parameters Parallel Chart, which shows the relationships between all parameters and each point in the project. In the Toolbox, double-click Parameter Parallel Chart to add this chart. Then, in the Outline table, under Charts, select Parameter Parallel Chart 0. This will cause the corresponding chart to appear in the window at the bottom right. The values at the top and bottom show the spread related to each parameter. Each colored line in the chart represents a corresponding point in the project (Figure 7.20).

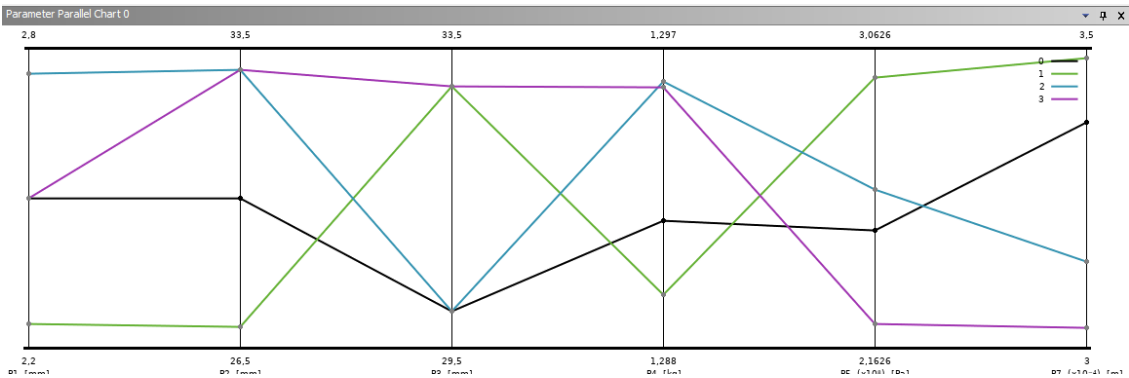


Figure 7.20 – Displaying the Parameters Parallel Chart panel

17 The solution is generated for the design point designated as Current in the Table panel. If, based on your study of the design, you decide that another design point is preferable, you can define it as the current design point. In the Table of Design Points, right-click DP 2 (if you consider design point 2 preferable) and select Copy inputs to Current (Figure 7.21).

Table of Design Points							
	A	B	C	D	E	F	G
1	Name	P1 - thick_ds	P2 - horiz_ds	P3 - vert_ds	P4 - Geometry Mass	P5 - Equivalent Stress Maximum	P7 - Total Deformation Maximum
2	Units	mm	mm	mm	kg	Pa	m
3	DP 0 (Current)	2,5	30	30	1,2918	2,519E+08	0,00033774
4	DP 1	2,25	27	33	1,2896	2,9768E+08	0,00034848
5	DP 2				1,296	2,6407E+08	0,00031457
6	DP 3				1,2958	2,2366E+08	0,00030356
*							

Copy

Set Update Order by Row

Show Update Order

Optimize Update Order

Duplicate Design Point

Delete Design Point

Copy inputs to Current

Export Selected Design Points

Set as Current

Update Selected Design Points

Export Table Data as CSV

Import from Design Point Service

Figure 7.21 – Defining a new current project point

Then, right-click on DP 0 and select Update Selected Design Points. When the update is complete, the results for DP 0 now match DP 1 in the table (Figure 7.22).

Table of Design Points						
	A	B	C	D	E	F
1	Name	P1 - thick_ds	P2 - horiz_ds	P3 - vert_ds	P4 - Geometry Mass	P5 - Equivalent Stress Maximum
2	Units	mm	mm	mm	kg	Pa
3	DP 0 (Current)	2,75	33	30	1,2	0,00033774
4	DP 1	2,25	27	33	1,2	0,00034848
5	DP 2	2,75	33	30	1,2	0,00031457
6	DP 3	2,5	33	33	1,2	0,00030356
*						

Figure 7.22 – Updating the current project point

18 Mechanical is launched. In the project tree Outline → Environment (A5) → Solution, we select Equivalent Stress and obtain a solution for the selected current point of the project (Figure 7.23).

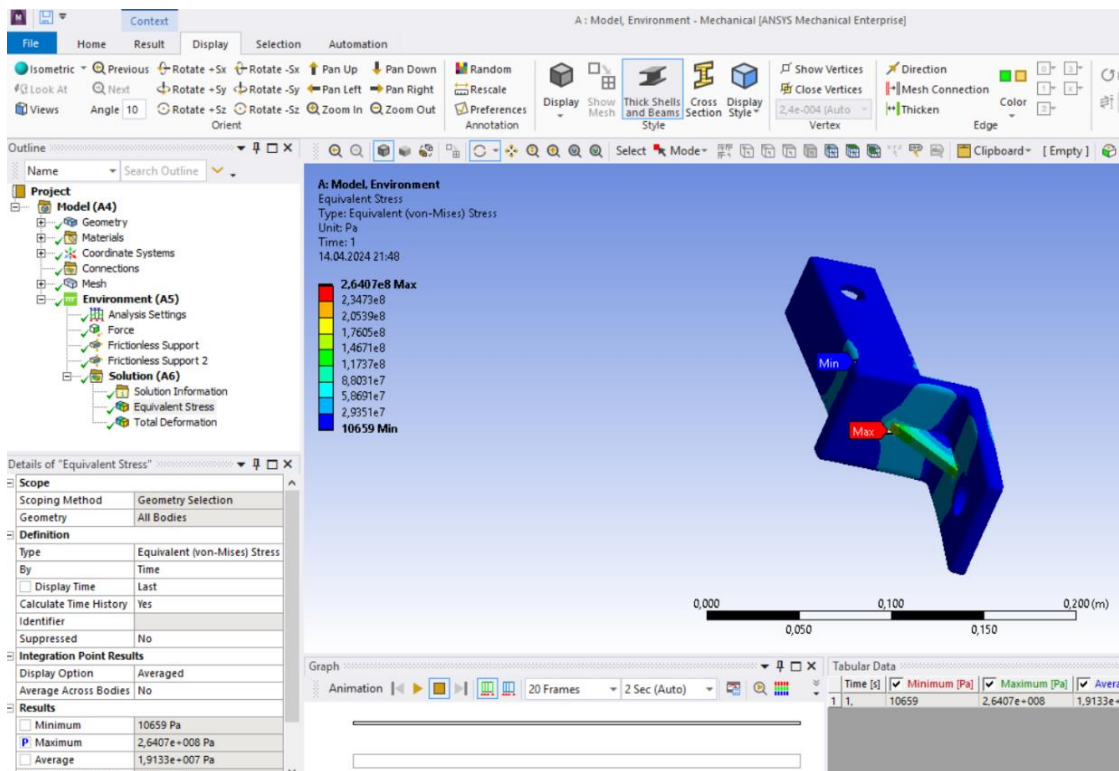


Figure 7.23 – Equivalent stresses for the current point of the project

Problem 7.2 Optimization in ANSYS DESIGNXPLORER

7.2.1 Introduction

It is difficult to obtain explicit analytical dependencies of object characteristics on design parameters for real objects of modeling. In such cases, computer modeling is used to reduce the number of expensive experiments. At the same time, the use of complex models for finite element analysis makes the modeling time very large. If one experiment with a model requires more than an hour, conducting hundreds of experiments when solving optimization problems becomes impractical. The solution to this problem is advisable with the use of surrogate modeling, within the framework of which mathematical predictive models for complex systems are built using the results of computational experiments with computer models. The formation of such models is based on the ideas of machine learning, where models are “trained” on a set of input and output data – the results of experiments. The constructed surrogate models replace the sources of data. As a rule, such models have significantly higher computational efficiency compared to the original models. Thus, the purpose of constructing surrogate models is to provide approximate values of output parameters based on input parameters without performing full calculations. This goal is achieved by using the method of planning computational experiments. Another goal of surrogate modeling is to use the created models in optimizing the parameters of structures and technological processes.

In the Ansys Workbench software environment, model optimization is performed in the DesignXplorer module. DesignXplorer uses input and output parameters, which can be DesignModeler parameters or parameters from various automated design systems. Input parameters can be thickness values, lengths, mechanical loads, material properties, etc. Output parameters are usually mass, stress, etc.

Once the input and output parameters have been configured, the following types of analyses can be run in DesignXplorer:

- Design of Experiments (DOE);
- Response Surface;
- Optimization.

Experimental design is the core of DesignXplorer. The experimental design technique allows determining the number of runs of a computational problem in order to obtain the most accurate response surface possible. Each solved combination of parameters is referred to as a Design Point.

After the experimental design is completed, a response surface is created based on the calculated results. A response surface is created for

each output parameter. The initial data for constructing the response surface are the numerical data presented in the Design Point results table. In general, the response surface helps to analyze the calculation results.

In order to provide the strength indicators of a part with a smaller mass or dimensions, in particular, it is necessary to use the optimization module. This module will allow you to select the necessary input parameters regarding the user-specified restrictions on the output parameters. For example, by specifying the mass indicator and stress level, the program will select the values of any dimensions of the part.

7.2.2 Problem statement

The purpose of the optimization study is to minimize the mass of the part while maintaining a safe stress level. The input parameters are the geometric dimensions of the part.

Following the methodology from Conducting a Parametric What-If Study in ANSYS Workbench, this study utilizes a tow hitch model created in ANSYS DesignModeler (Figure 7.1). Figure 7.1 presents the loads and constraints applied to the tow hitch.

The geometric dimensions of the tow hitch (Figures 7.2, 7.24) will serve as input parameters for this analysis.

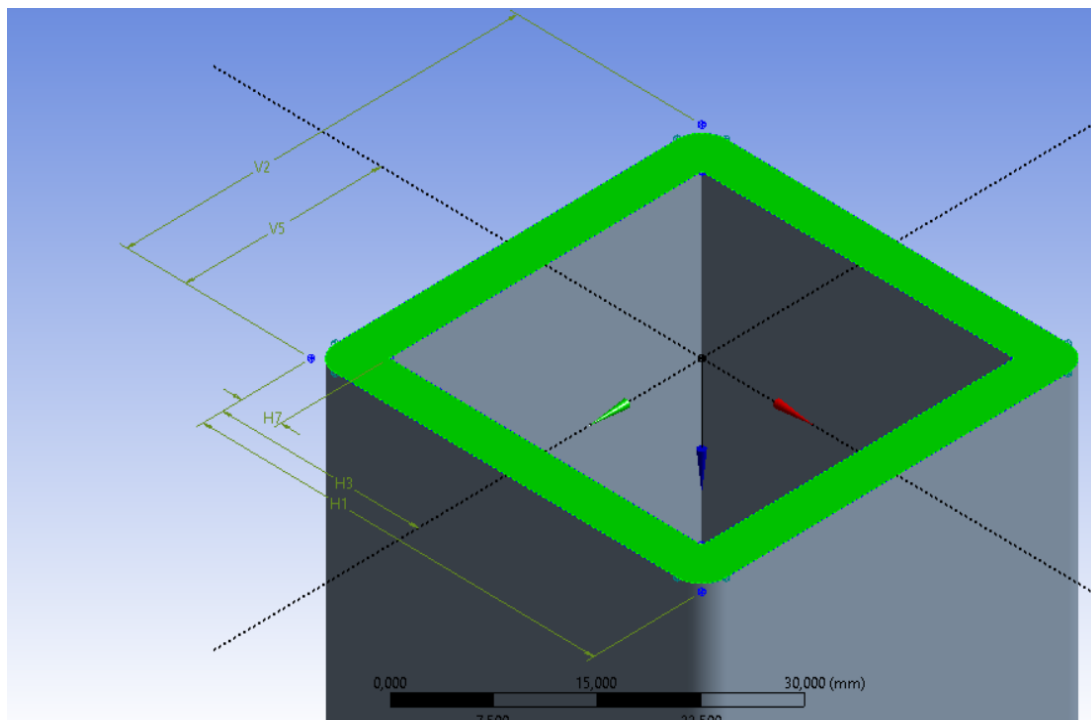


Figure 7.24 – Geometrical parameter H7 (square pipe thickness) used as an input parameter of the model

7.2.3 Implementation of the problem using the Ansys Workbench graphical interface

1 The solution to the problem begins with launching Ansys Workbench, which is executed from the main menu Start → Programs → Ansys 2020R1 → Workbench 2020R1.

2 Select File → Open. Then, in the window that appears, select the file named receiver.wbpz. Save the project with the name receiver and the extension wbpj. As a result, an analytical block will appear in the Project Schematic window, which we will use as a template for our model. (Figure 7.3).

3 Double-click on the Geometry cell. The DesignModeler window that opens will display the towing device model. In the Tree Outline window, select Gusset. Next, in the Details View, check the box for FD1, Depth to make it an input parameter. After that, a dialog box will appear for specifying the parameter name. Specify the name thick_ds (Figure 7.4).

4 In the Tree Outline window, select YZPlane. In YZPlane, select Sketch4, in the Dimensions section, as in step 3, set the flags on H6 (name – horiz_ds) and V7 (name – vert_ds) (Figures 7.5, 7.6).

5 In the Tree Outline window, select Plane4. In Plane4, select Sketch2, in the Dimensions section, as in points 2–3, check the boxes for H7 (name – horiz_ds) and V7 (name – thick_tube) (Figure 7.25). Close DesignModeler.

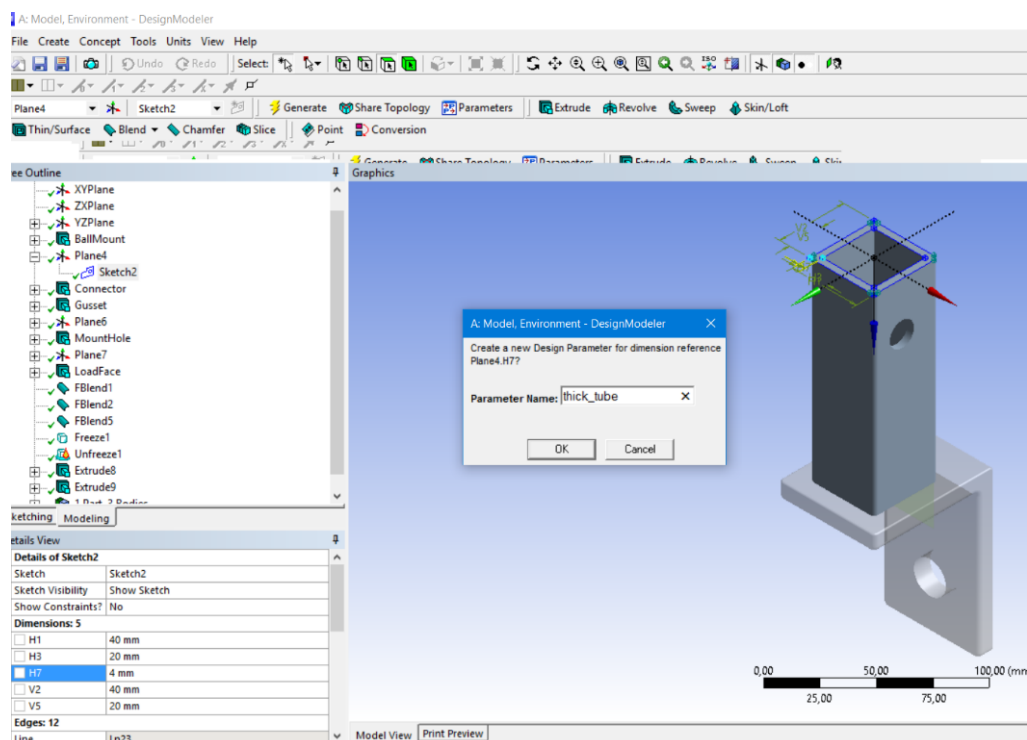


Figure 7.25 – Setting the thick_tube parameter

6 Double-click on the Parameters cell and open the corresponding window, in which the specified input parameters of the project are visible (Figure 7.26).

Outline of All Parameters				
	A	B	C	D
1	ID	Parameter Name	Value	Unit
2	Input Parameters			
3	Model, Environment (A1)			
4	P1	thick_ds	2,5	mm
5	P2	horiz_ds	30	mm
6	P3	vert_ds	30	mm
7	P8	thick_tube	4	mm
*	New input parameter	New name	New expression	
9	Output Parameters			
*	New output parameter		New expression	
11	Charts			
12	Parameter Parallel Chart 0			
13	Parameter Chart 0			

Figure 7.26 – Project input parameters

7 Double-click on Setup (Project Schematic) and open the Mechanical window (Figure 7.9).

8 Next, in the Outline window of the Workbench Mechanical graphical interface, select the Model → Geometry item from the tree structure of the calculation model. In the Details of “Geometry” window, in the Properties tab, check the Mass cell (Figure 7.10).

9 In the Outline window, find Solution (A6) and select Equivalent Stress. Next, in the Details section, under the Results cell, check the Maximum box to determine the maximum equivalent stress of the output parameter (Figure 7.11).

10 Exit Mechanical and update the project in Project Schematic (Update Project). Double-click on the Parameters cell and open the corresponding window, which shows the specified input and output parameters of the project and the generated project point (Figure 7.27).

Outline of All Parameters				
	A	B	C	D
1	ID	Parameter Name	Value	Unit
2	Input Parameters			
3	Model, Environment (A1)			
4	P1	thick_ds	2,5	mm
5	P2	horiz_ds	30	mm
6	P3	vert_ds	30	mm
7	P8	thick_tube	4	mm
*	New input parameter	New name	New expression	
9	Output Parameters			
10	Model, Environment (A1)			
11	P9	Geometry Mass	1,2918	kg
12	P10	Equivalent Stress Maximum	2,519E+08	Pa
*	New output parameter		New expression	
14	Charts			
15	Parameter Parallel Chart 0			
16	Parameter Chart 0			

Figure 7.27 – Input and output parameters of the project

11 In the Toolbox → Design Exploration → window, double-click on Response Surface and add a new response surface to the project (Figure 7.28).

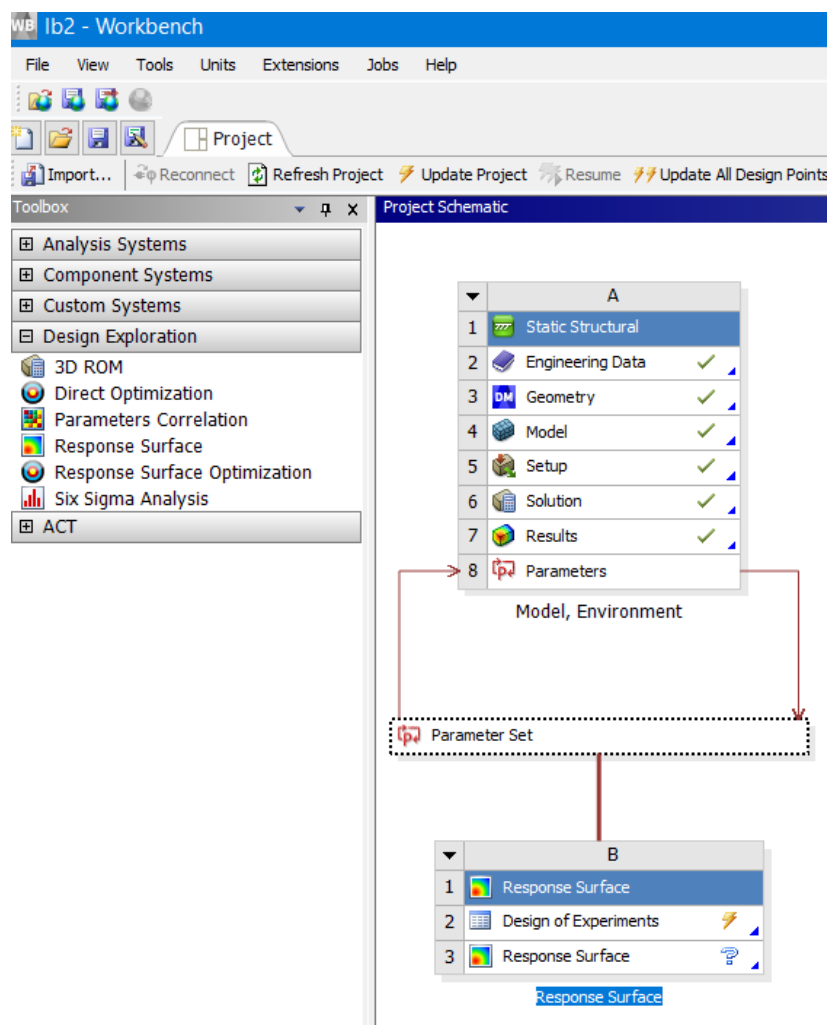


Figure 7.28 – Adding a Response Surface to a Project

12 In the newly created Response Surface block, double-click “Design of Experiments” (the lightning bolt icon cell). The resulting window displays all predefined input and output parameters in the Outline table (Figure 7.29).

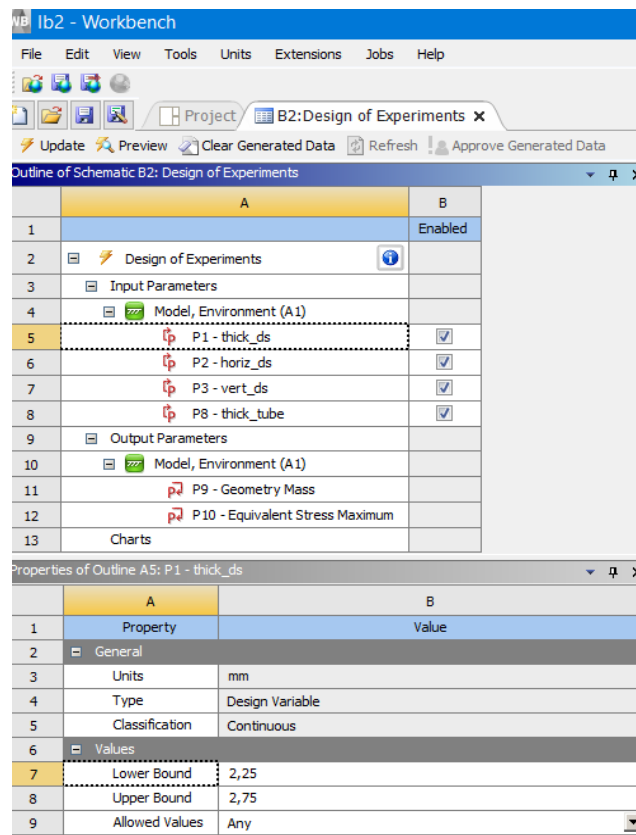


Figure 7.29 – Design of Experiments window

13 Having selected the parameters P2 – horiz_ds, P3 – vert_ds, set the upper and lower boundaries equal to 25 mm and 35 mm. For the parameter P8 – thick_tube, set the boundaries equal to 1 mm and 5 mm. Disable the parameter P1 – thic_ds from the simulation by unchecking the corresponding box (Figure 7.30).

14 Next, select Design of Experiment and in Properties, make sure that Design of Experiment Type is set to Central Composite Design, and select the face-centered version of the central composite design of the experiment (Figure 7.31).

In Ansys Workbench, there is a possibility to choose different variants of experiment planning: Latin hypercube, central compositional plan, etc. In turn, each of these types has several more settings, which can be found in more detail in the documentation for the software environment. In addition, the user can manually enter the input factor planning matrix by selecting the Custom planning type.

15 On the toolbar, click Preview to see which points were generated in the design of the experiment. The Table of Outline shows fifteen points (Figure 32). Refresh Design of Experiments (Update). When the update is complete, the table will be filled with the calculation results (Figure 7.33).

Outline of Schematic B2: Design of Experiments

	A	B
1		Enabled
2	✓ Design of Experiments	
3	Input Parameters	
4	Model, Environment (A1)	
5	P1 - thick_ds	
6	P2 - horiz_ds	✓
7	P3 - vert_ds	✓
8	P8 - thick_tube	✓
9	Output Parameters	
10	Model, Environment (A1)	
11	P9 - Geometry Mass	
12	P10 - Equivalent Stress Maximum	
13	Charts	
14	Parameters Parallel	
15	Design Points vs Parameter	

Properties of Outline A8: P8 - thick_tube

	A	B
1	Property	Value
2	General	
3	Units	mm
4	Type	Design Variable
5	Classification	Continuous
6	Values	
7	Lower Bound	1
8	Upper Bound	5
9	Allowed Values	Any

Figure 7.30 – Changing the range of input parameters

	A	B
1		Enabled
2	✓ Design of Experiments	
3	Input Parameters	
4	Model, Environment (A1)	
5	P1 - thick_ds	
6	P2 - horiz_ds	✓
7	P3 - vert_ds	✓
8	P8 - thick_tube	✓
9	Output Parameters	
10	Model, Environment (A1)	
11	P9 - Geometry Mass	
12	P10 - Equivalent Stress Maximum	
13	Charts	
14	Parameters Parallel	
15	Design Points vs Parameter	

Properties of Outline A2: Design of Experiments

	A	B
1	Property	Value
2	Design Points	
3	Preserve Design Points After DX Run	
4	Failed Design Points Management	
5	Number of Retries	0
6	Design of Experiments	
7	Design of Experiments Type	Central Composite Design
8	Design Type	Face-Centered
9	Template Type	Face-Centered
10	Design Point Report	Rotatable
11	Report Image	VIF-Optimality G-Optimality Auto Defined

Figure 7.31 – Selecting an experimental design

	A	B	C	D	E	F
1	Name	P2 - horiz_ds (mm)	P3 - vert_ds (mm)	P8 - thick_tube (mm)	P9 - Geometry Mass (kg)	P10 - Equivalent Stress Maximum (Pa)
2	1	30	30	3	⚡	⚡
3	2	25	30	3	⚡	⚡
4	3	35	30	3	⚡	⚡
5	4	30	25	3	⚡	⚡
6	5	30	35	3	⚡	⚡
7	6	30	30	1	⚡	⚡
8	7	30	30	5	⚡	⚡
9	8	25	25	1	⚡	⚡
10	9	35	25	1	⚡	⚡
11	10	25	35	1	⚡	⚡
12	11	35	35	1	⚡	⚡
13	12	25	25	5	⚡	⚡
14	13	35	25	5	⚡	⚡
15	14	25	35	5	⚡	⚡
16	15	35	35	5	⚡	⚡

Figure 7.32 – Numerical experiment plan

	A	B	C	D	E	F
1	Name	P2 - horiz_ds (mm)	P3 - vert_ds (mm)	P8 - thick_tube (mm)	P9 - Geometry Mass (kg)	P10 - Equivalent Stress Maximum (Pa)
2	1	30	30	3	1,1703	2,677E+08
3	2	25	30	3	1,1671	3,0062E+08
4	3	35	30	3	1,1735	2,8806E+08
5	4	30	25	3	1,1671	3,6382E+08
6	5	30	35	3	1,1735	2,6119E+08
7	6	30	30	1	0,90453	2,9499E+08
8	7	30	30	5	1,4059	2,4931E+08
9	8	25	25	1	0,89864	3,5926E+08
10	9	35	25	1	0,90404	4,1295E+08
11	10	25	35	1	0,90404	3,3006E+08
12	11	35	35	1	0,9114	2,5812E+08
13	12	25	25	5	1,4	3,1737E+08
14	13	35	25	5	1,4054	3,6737E+08
15	14	25	35	5	1,4054	3,3213E+08
16	15	35	35	5	1,4127	2,0592E+08

Figure 7.33 – Calculation results

16 We return to the Project Schematic window. In the Response Surface block, double-click on the Response Surface cell. Update everything in the window that opens by selecting the Response Surface item in the Outline table, and set Full 2nd-Order Polynomial – the standard algorithm for constructing the metamodel in the Properties of Outline table (Figure 7.34).

The following algorithms are also available for constructing a metamodel using the ANSYS Design Exploration module: Kriging; Non-Parametric Regression; Neural Network, etc.

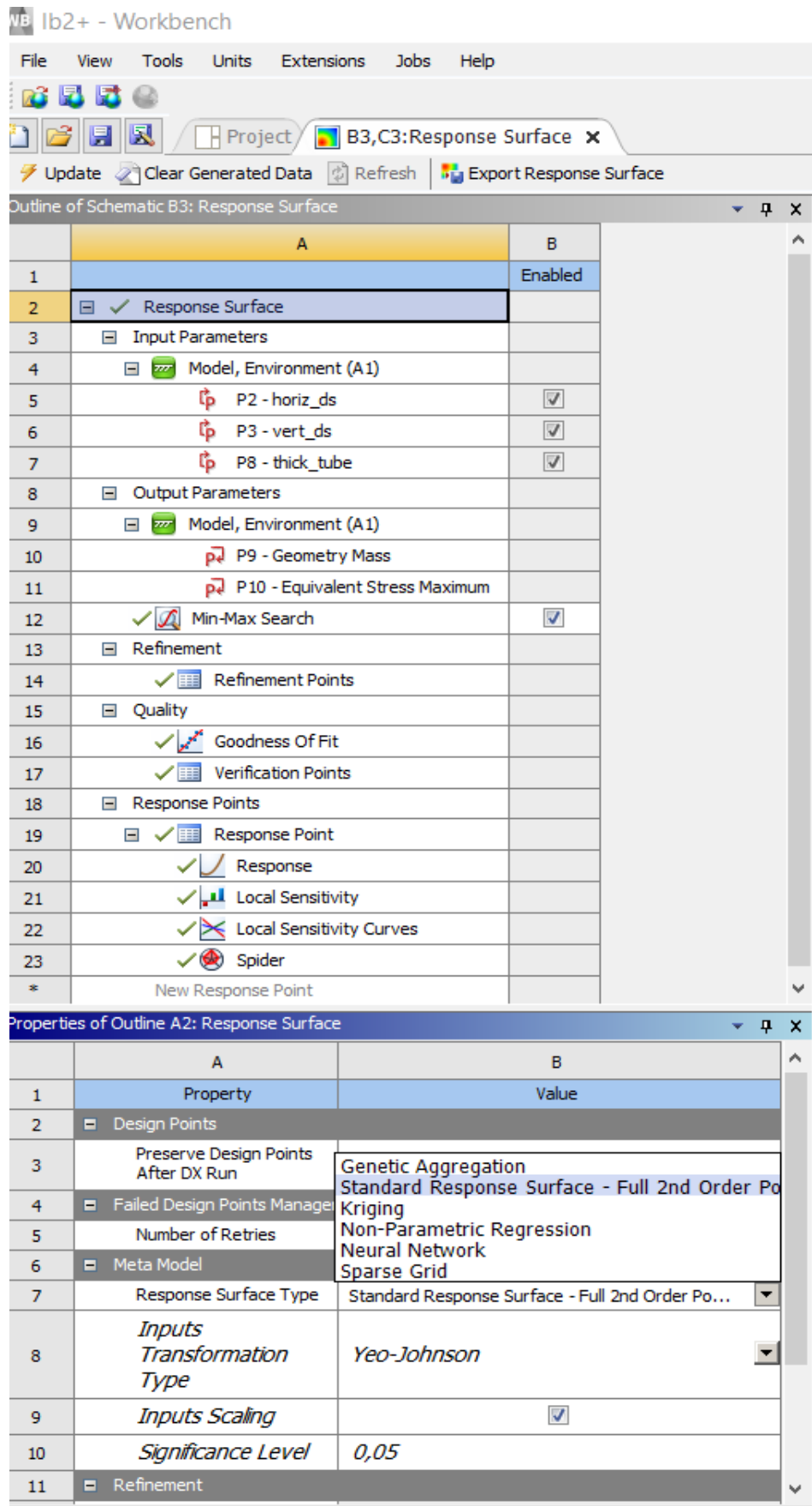


Figure 7.34 – Response Surface window

17 In the Outline panel, in the Response Points section, we see that one point has been generated along with its diagram (Figure 7.34). By selecting the diagram of this point (Response), we can view the automatically generated graph. By changing the corresponding parameters, we can see how the output parameters change when the input parameters change in three viewing options: 2D, 3D, and 2D Slices (Figure 7.35).

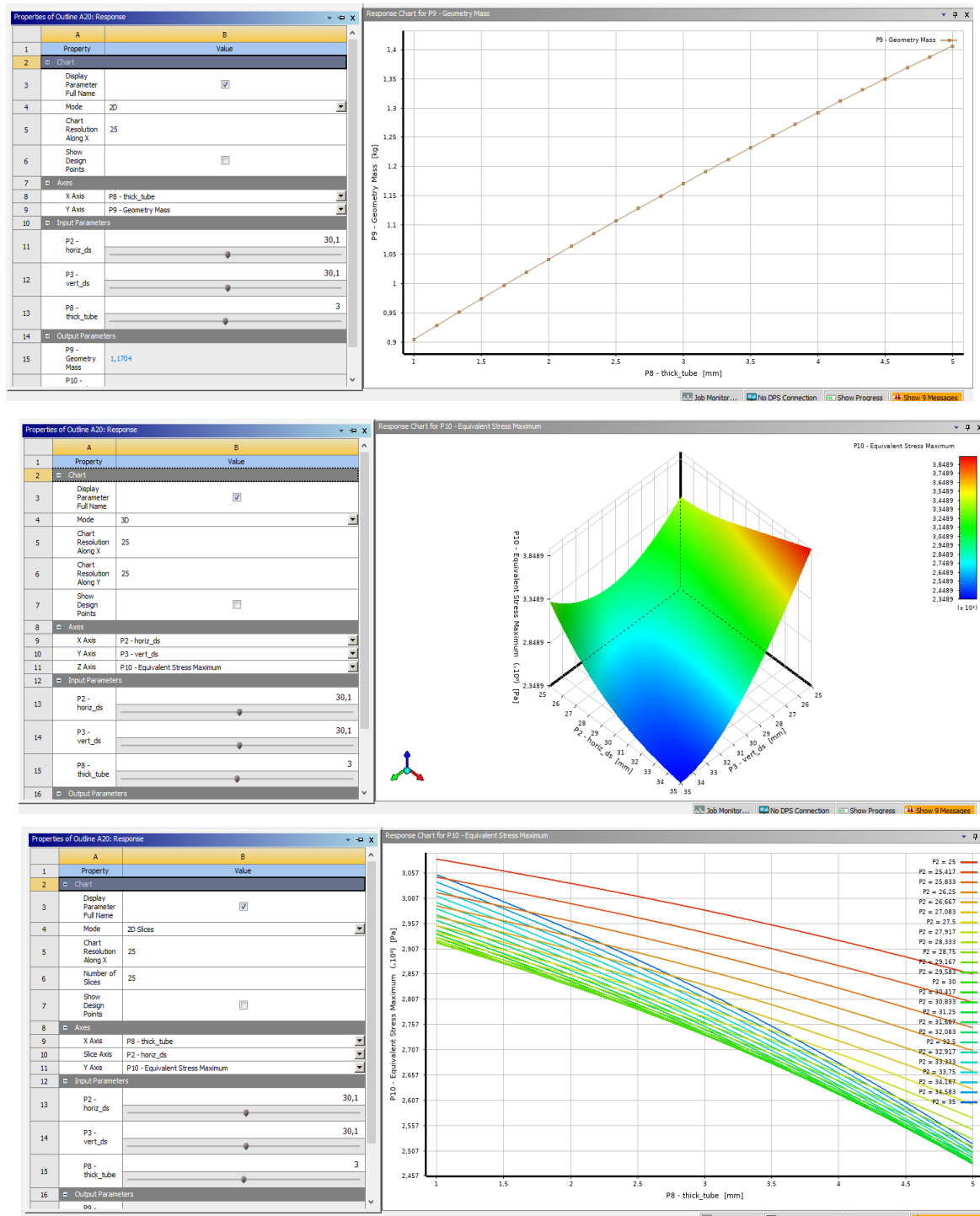


Figure 7.35 – Response Surface Viewing Options

18 When you switch to the Goodness Of Fit section on the Outline panel, data is generated that makes it possible to judge the quality of the metamodel (Figure 7.36). The abscissa axis of the graph shows the values of finite element modeling, and the ordinate axis illustrates the values obtained using the corresponding regression equations. The closer the obtained points are to the diagonal, the higher the accuracy of the model.

To evaluate the resulting models, the criteria presented in Figure 7.35 may also be applied:

- coefficient of determination

$$R^2 = 1 - \frac{\sum_{i=1}^n (d_i - y_i)^2}{\sum_{i=1}^n (d_i - \bar{d})^2},$$

- Root Mean Square Error (RMSE)

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (d_i - y_i)^2},$$

where d_i represents values determined by the finite element method, y_i denotes values determined using regression models.

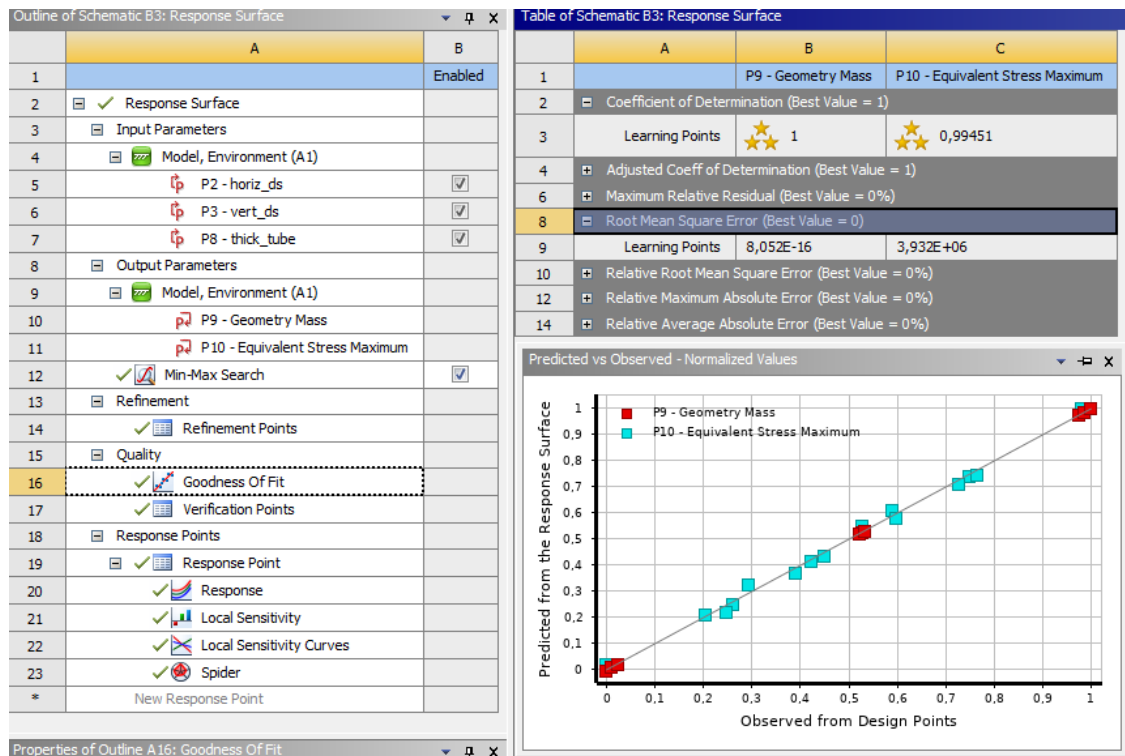


Figure 7.36 – Metamodel quality assessment

19 When you go to the Local Sensitivity section in the Outline panel, a diagram is built showing the influence of input parameters on output parameters (see Figure 7.37).

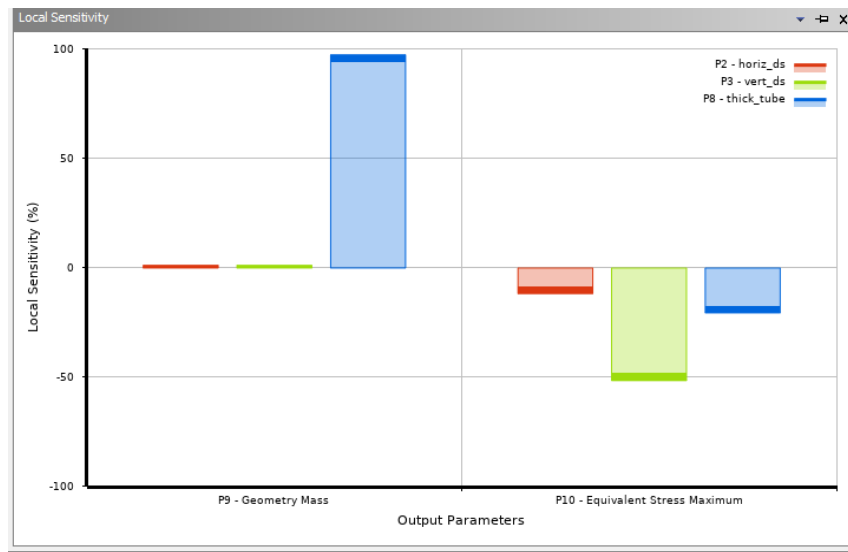


Figure 7.37 – Sensitivity diagram of the optimized parameters

20 In the Toolbox → Design Exploration → window, double-click on Response Surface Optimization and add a new optimization block to the project (Figure 7.38).

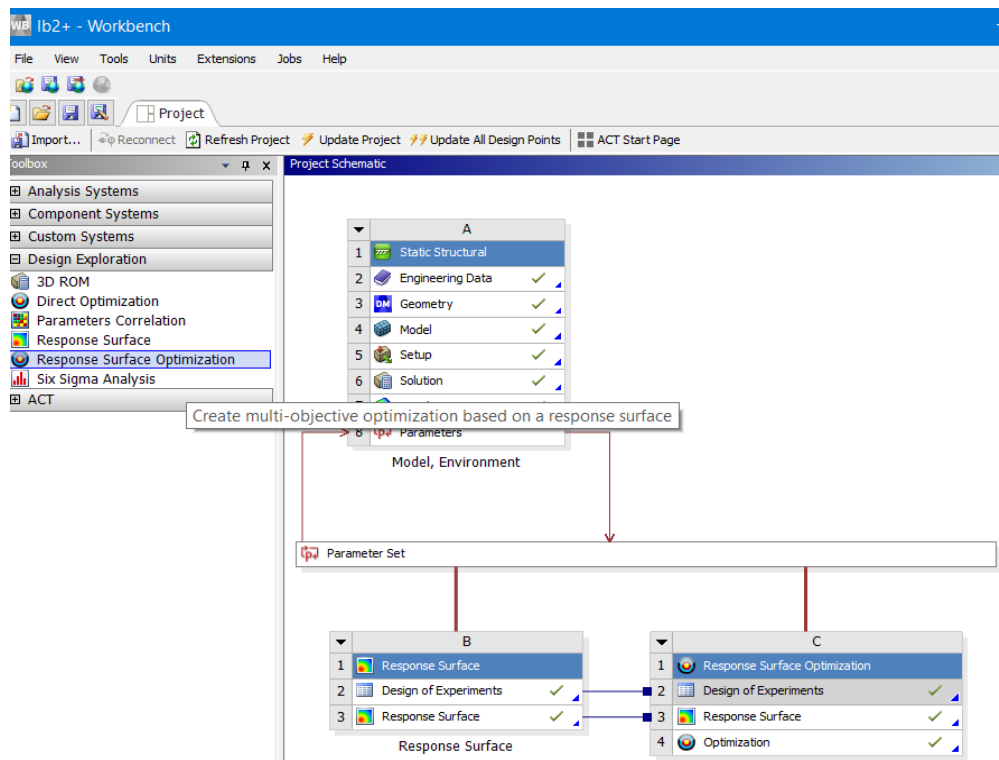


Figure 7.38 – Adding Response Surface Optimization to a project

21 In the Response Surface Optimization block, double-click the Optimization cell to open it (Figure 7.39).

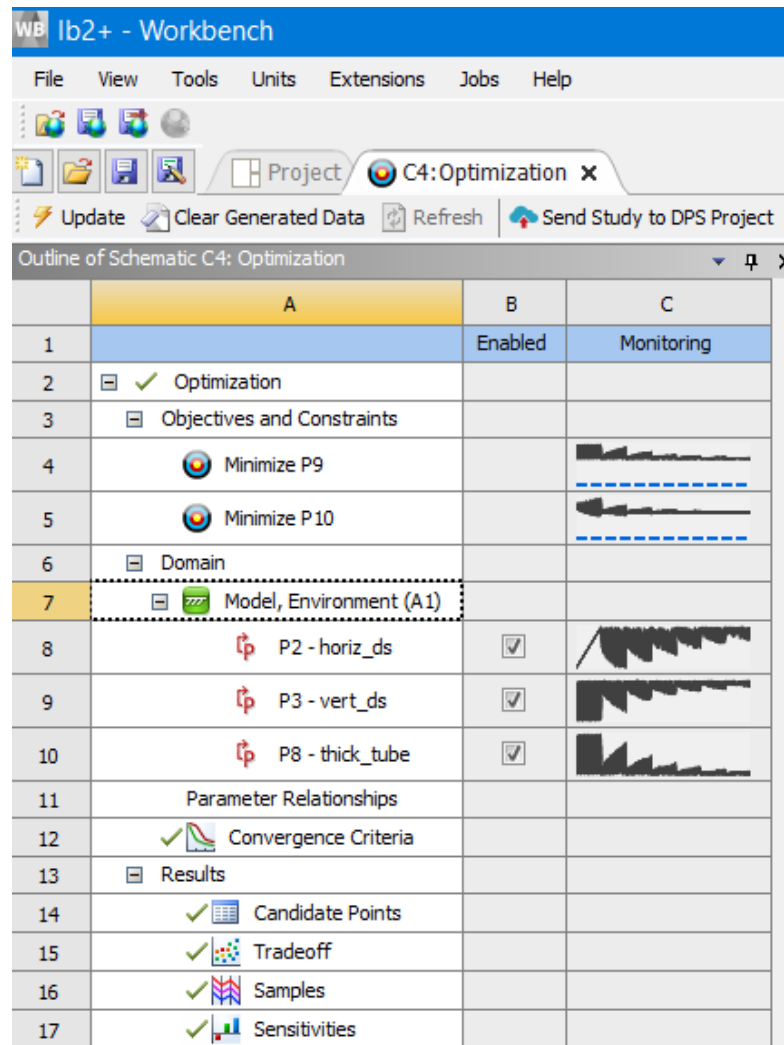


Figure 7.39 – Response Surface Optimization window

22 In the Outline panel, select Optimization. To perform multi-criteria optimization, in the Properties panel, set the optimization parameters as shown in Figure 7.40, selecting the MOGA genetic algorithm.

Multicriterial optimization is the simultaneous optimization of two or more objective functions. One of the methods for solving such problems is to identify a set of Pareto-optimal solutions that do not dominate each other, and the improvement of one parameter leads to the deterioration of others. Genetic algorithms are a special case of evolutionary methods and provide a search for the best solutions using inheritance and strengthening of the useful properties of many objects in the process of simulating their evolution. One of the effective genetic algorithms is the MOGA (Multi-Objective Genetic Algorithm) algorithm.

Properties of Outline A2: Optimization		
	A	B
1	Property	Value
2	Design Points	
3	Preserve Design Points After DX Run	☐
4	Failed Design Points Management	
5	Number of Retries	0
6	Optimization	
7	Method Selection	Manual
8	Method Name	MOGA
9	Estimated Number of Evaluations	10000
10	Tolerance Settings	☒
11	Verify Candidate Points	☐
12	Type of Initial Sampling	Screening
13	Number of Initial Samples	500
14	Number of Samples Per Iteration	500
15	Maximum Allowable Pareto Percentage	70
16	Convergence Stability Percentage	2
17	Maximum Number of Iterations	20
18	Mutation Probability	0,01
19	Crossover Probability	0,98
20	Maximum Number of Candidates	3

Figure 7.40 – Selecting an optimization algorithm

23 In the Outline panel, select Objectives and Constraints. In the Table area, for the P9 – Geometry Mass and P10 – Equivalent Stress Maximum parameters, specify the objectives as shown in Figure 7.41, and update the Optimization cell.

Table of Schematic C4: Optimization									
	A	B	C	D	E	F	G	H	I
1	Name	Parameter	Objective			Constraint			
2			Type	Target	Tolerance	Type	Lower Bound	Upper Bound	Tolerance
3	Minimize P9	P9 - Geometry Mass	Minimize	0		No Constraint			
4	Minimize P10	P10 - Equivalent Stress Maximum	Minimize	0		No Constraint			
*		Select a Parameter							

Figure 7.41 – Statement of the optimization problem

24 To view the candidate points identified as a result of optimization, select Candidate Points in the Results section of the Outline panel (Figure 7.42).

Table of Schematic C4: Optimization , Candidate Points								
	A	B	C	D	E	F	G	H
1	Reference	Name	P2 - horiz_ds (mm)	P3 - vert_ds (mm)	P8 - thick_tube (mm)	P9 - Geometry Mass (kg)		P10 - Equivalent Stress Maximum (Pa)
2						Parameter Value	Variation from Reference	Parameter Value
3		Candidate Point 1	34,632	34,807	1,018	✗ 0,91351	-0,35%	✗ 2,5412E+08
4		Candidate Point 2	34,784	34,531	1,0178	✗ 0,91339	-0,37%	✗ 2,5475E+08
5		Candidate Point 3	33,792	34,782	1,0457	✗ 0,91676	0,00%	✗ 2,5543E+08
*		New Custom Candidate Point	.30	.30	.3			

Figure 7.42 – Optimization results

25 To verify that Candidate Point 1 is an effective option, right-click it and select Verify by Design Point Update. When the update is complete, Candidate Point 1 (verified) will additionally appear in the Table panel (Figure 7.43).

Table of Schematic C4: Optimization , Candidate Points								
	A	B	C	D	E	F	G	H
1	Reference	Name	P2 - horiz_ds (mm)	P3 - vert_ds (mm)	P8 - thick_tube (mm)	P9 - Geometry Mass (kg)		P10 - Equivalent Stress Maximum (Pa)
2						Parameter Value	Variation from Reference	Parameter Value
3		Candidate Point 1	34,632	34,807	1,018	✗ 0,91351	-0,35%	✗ 2,5412E+08
4		Candidate Point 1 (verified)				✗ 0,91351	-0,35%	✗ 2,5355E+08
5		Candidate Point 2	34,784	34,531	1,0178	✗ 0,91339	-0,37%	✗ 2,5475E+08
6		Candidate Point 3	33,792	34,782	1,0457	✗ 0,91676	0,00%	✗ 2,5543E+08
*		New Custom Candidate Point	.30	.30	.3			

Figure 7.43 – Checking the optimization results

26 In the Outline panel, under Results, select Candidate Points. Right-click Candidate Point 1 and select Insert as Design Point (see Figure 7.44). Close the Optimization cell.

Table of Schematic C4: Optimization , Candidate Points								
	A	B	C	D	E	F	G	H
1	Reference	Name	P2 - horiz_ds (mm)	P3 - vert_ds (mm)	P8 - thick_tube (mm)	P9 - Geometry Mass (kg)		P10 - Equivalent Stress Maximum (Pa)
2						Parameter Value	Variation from Reference	Parameter Value
3		Candidate Point 1	34,632	34,807	1,018	✗ 0,91351	-0,35%	✗ 2,5412E+08
4		Candidate Point 1 (verified)				✗ 0,91351	-0,35%	✗ 2,5355E+08
5		Candidate Point 2	34,784	34,531	1,0178	✗ 0,91339	-0,37%	✗ 2,5475E+08
6		Candidate Point 3	33,792	34,782	1,0457	✗ 0,91676	0,00%	✗ 2,5543E+08
*		New Custom Candidate Point	.30	.30	.3			

Figure 7.44 – Selecting Candidate Point 1 as a Design Point

27 In the main Project Schematic window, double-click Parameter Set. In the Table pane, right-click DP 1 and select Copy inputs to Current (Figure 7.45). The current design point now has the same input parameter values as your best candidate. Right-click the current design point and select Update Selected Design Points (Figure 7.46).

	A	B	C	D	E	F	G
1	Name ▾	P1 - thick_ds ▾	P2 - horiz_ds ▾	P3 - vert_ds ▾	P8 - thick_tube ▾	P9 - Geometry Mass ▾	P10 - Equivalent Stress Maximum ▾
2	Units	mm ▾	mm ▾	mm ▾	mm ▾	kg	Pa
3	DP 0 (Current)	2,5	34,632	34,807	1,018	⚡ 0,91351	⚡ 2,5355E+08
4	DP 1	2,5	34,632	34,807	1,018	0,91351	2,5355E+08
*							

Copy
Paste
Set Update Order by Row
Show Update Order
Optimize Update Order
Duplicate Design Point
Delete Design Point
Copy inputs to Current
Export Selected Design Points
Set as Current
Update Selected Design Points
Export Table Data as CSV
Import from Design Point Service
Expand All
Collapse All
Copy Design Points to

Figure 7.45 – Selecting DP1 as the current project point

	A	B	C	D	E	F	G
1	Name ▾	P1 - thick_ds ▾	P2 - horiz_ds ▾	P3 - vert_ds ▾	P8 - thick_tube ▾	P9 - Geometry Mass ▾	P10 - Equivalent Stress Maximum ▾
2	Units	mm ▾	mm ▾	mm ▾	mm ▾	kg	Pa
3	DP 0 (Current)	2,5	34,632	34,807	1,018	⚡ 0,91351	⚡ 2,5355E+08
4	DP 1	2,5	34,632	34,807	1,018	0,91351	2,5355E+08
*							

Copy
Set Update Order by Row
Show Update Order
Optimize Update Order
Duplicate Design Point
Export Selected Design Points
Set as Current
Update Selected Design Points
Export Table Data as CSV
Import from Design Point Service
Expand All
Collapse All
Copy Design Points to

Figure 7.46 – Updating the calculation point

28 Close the Parameter Set window. Double-click the Solution cell to open it (Figure 7.47).

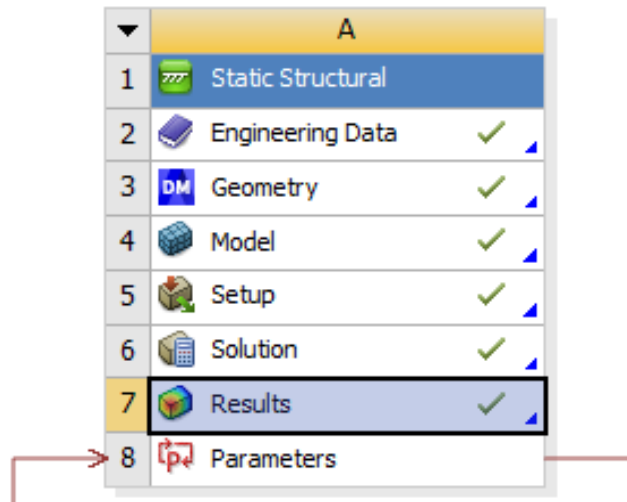


Figure 7.47 – Results cell

29 In the Outline window, select Equivalent Stress to see the results (Figure 7.48).

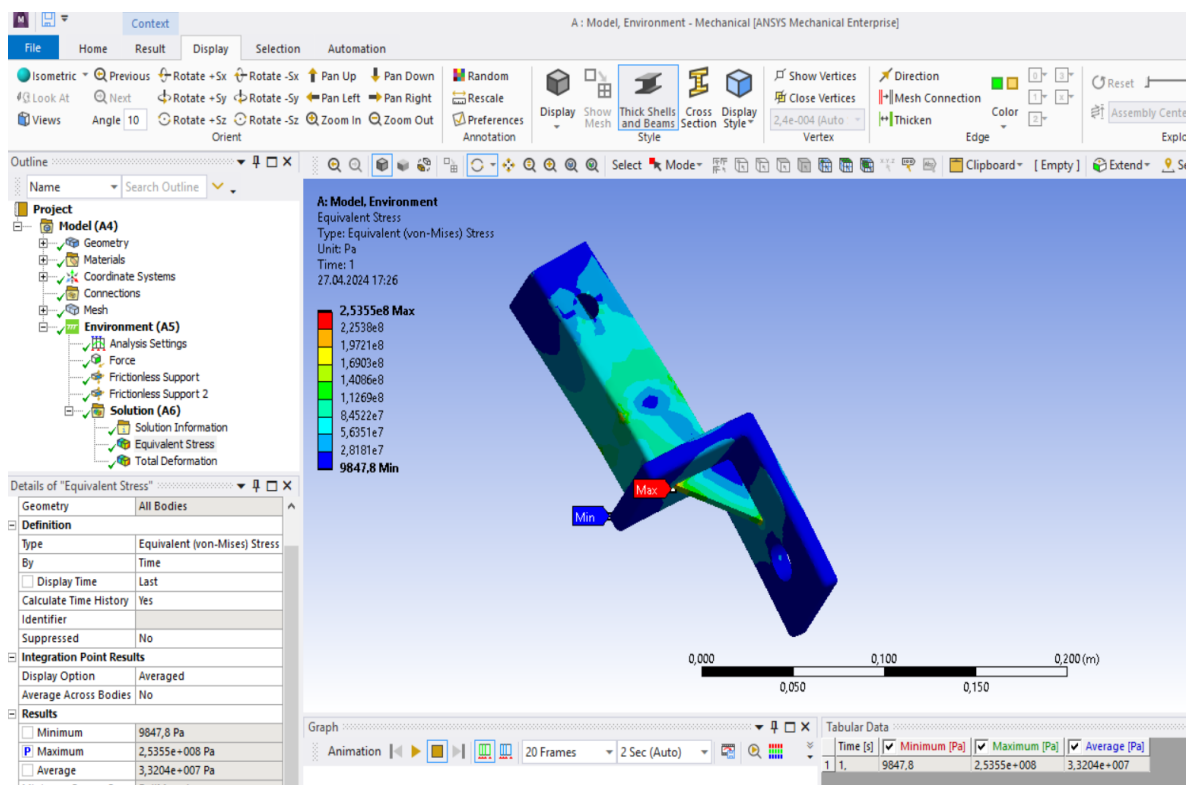


Figure 7.48 – Calculation results

8 TEST ASSIGNMENTS

Test 8.1 Introduction to computer modeling technologies

1 In ANSYS, which module is used for constructing geometric models, assigning material properties, and applying boundary conditions?

- a) SOLUTION;
- b) PREP7;
- c) POST26;
- d) POST1.

2 In ANSYS, which module is used for setting boundary conditions, selecting and configuring the solver as well as executing the solution?

- a) PREP7;
- b) POST26;
- c) POST1;
- d) SOLUTION.

3 In ANSYS, which module is used for reviewing steady-state solution results, analyzing data by load/time steps, and graphically visualizing results?

- a) POST26;
- b) SOLUTION;
- c) PREP7;
- d) POST1.

4 In ANSYS, which module is specifically designed for reviewing solution results in the form of graphs result – load step or result – time?

- a) PREP7;
- b) POST1;
- c) POST26;
- d) SOLUTION.

5 The Main Menu window in the ANSYS provides:

- a) quick access to a number of commands;
- b) access to commands accessible from any processor;
- c) command input;
- d) access to all processor operations – preprocessor, solution processor and postprocessor;
- e) graphical output of objects.

6 The Utility Menu window in ANSYS serves to:

- a) access commands available from any processor;
- b) graphically display objects;
- c) quickly access a number of commands;
- d) enter commands;
- e) access all processor operations – preprocessor, solution processor and postprocessor.

7 The Toolbar window in ANSYS serves for:

- a) access to all processor operations – preprocessor, solution processor and postprocessor;
- b) graphical output of objects;
- c) quick access to a number of commands;
- d) access to commands available from any processor;
- e) entering commands.

8 The Input window in ANSYS is used for:

- a) entering commands;
- b) quick access to a number of commands;
- c) access to all processor operations – preprocessor, solution processor and postprocessor;
- d) access to commands available from any processor;
- e) graphical output of objects.

9 The Graphics window in ANSYS is used for:

- a) access to all processor operations – preprocessor, solution processor and postprocessor;
- b) access to commands available from any processor;
- c) quick access to a number of commands;
- d) command input;
- e) graphical output of objects.

10 In ANSYS APDL programming, there can be no more than _____ statement(s) on each line.

11 In ANSYS APDL programming, the maximum number of characters per line, including spaces, must not exceed _____.

12 In ANSYS APDL programming, which of these are valid scientific notation formats for the number 25,000?

- a) 25000;
- b) 25D3;
- c) 25E3.

13 In ANSYS APDL programming, which of these symbols are prohibited in command syntax?

- a) !;
- b) @;
- c) #.

14 SOLID elements in ANSYS are used to model:

- a) beam structures;
- b) contact conditions;
- c) three-dimensional objects;
- d) two-dimensional problems.

15 PLANE type elements in ANSYS are used to model:

- a) three-dimensional objects;
- b) rod structures;
- c) two-dimensional problems;
- d) contact conditions.

16 BEAM type elements in ANSYS are used to model:

- a) contact conditions;
- b) beam structures;
- c) two-dimensional problems;
- d) three-dimensional objects.

17 To perform a transient (non-stationary) analysis in ANSYS, the correct menu selection is:

- a) STATIC;
- b) TRANS;
- c) SUBSTR;
- d) MODAL;
- e) HARMIC.

18 To perform a stationary (steady-state) analysis in ANSYS, the correct menu selection is:

- a) TRANS;
- b) HARMIC;
- c) MODAL;
- d) STATIC;
- e) SUBSTR.

19 The database file created by ANSYS has the extension:

- a) db;
- b) log;
- c) err;
- d) rth;
- e) rst.

20 The mechanical analysis results file generated by ANSYS has the extension:

- a) log;
- b) db;
- c) rst;
- d) rth;
- e) err.

21 The thermal analysis results file created by ANSYS has the extension:

- a) rth;
- b) err;
- c) log;
- d) db;
- e) rst.

22 To review solution results in ANSYS, which module provides the necessary commands?

- a) POST1;
- b) POST26;
- c) PREP7;
- d) SOLU.

23 To define material properties in ANSYS, which module provides the necessary commands?

- a) POST1;
- b) POST26;
- c) PREP7;
- d) SOLU.

24 To apply loads to a model in ANSYS, which module must be used?

- a) POST26;
- b) SOLU;
- c) PREP7;
- d) POST1.

25 To determine the analysis type in ANSYS, which module provides the necessary commands?

- a) SOLU;
- b) PREP7;
- c) POST26;
- d) POST1.

26 The main properties of mathematical models include:

- a) simplicity (complexity);
- b) potentiality;
- c) adequacy;
- d) feasibility.

27 The modeling is based on the following fundamental principles:

- a) information sufficiency;
- b) feasibility;
- c) multiplicity of models;
- d) aggregation.

28 The main aspects that determine the effectiveness of using CALS technologies include:

- a) computer automation, which allows increasing the productivity of the main processes and operations of information creation;
- b) information integration of processes, which ensures joint and repeated use of the same data;
- c) transition to paperless organization of processes and application of new models of their organization.

29 The main sources of errors in the results of computer modeling are:

- a) model error;
- b) data error;
- c) method error;
- d) computational error.

30 Which of the following are valid parameter names in ANSYS?

- a) ABC;
- b) PI;
- c) NEW_VALUE;
- d) 2CF3.

31 Which of the following are valid parameter names in ANSYS?

- a) X;
- b) M&E;
- c) 2CF3;
- d) 1x.

32 Which of the following are valid parameter names in ANSYS?

- a) X_OR_Y;
- b) ABC;
- c) CBV.

33 What is the result of executing the command *GET,ww1, NODE,6,LOC,Z in ANSYS?

- a) returns the z-coordinate of node number 5 and stores the result as parameter ww1;
- b) returns the y-coordinate of node number 6 and stores the result as parameter ww;
- c) returns the z-coordinate of node number 6 and stores the result as parameter ww1.

34 When executing the get-function KP(x,y,z) in ANSYS, the following is/are determined:

- a) the number of the selected node closest to the point (x,y,z) in the active coordinate system;
- b) the number of the selected key point closest to the point (x,y,z) in the active coordinate system;
- c) the coordinates of the node with number n in the active coordinate system;
- d) the displacements UX, UY and UZ in the node with number n.

35 When executing the get-function UX(n) in ANSYS, the following is/are determined:

- a) the X coordinate of the node with number n in the active coordinate system;
- b) the Y coordinate of the node with number n in the active coordinate system;
- c) the displacement UX in the node with number n;
- d) the displacement UZ in the node with number n.

36 If you sequentially enter two commands in ANSYS N,1,0,0,0,,, and *repeat,4,1,0,2,1, the execution will yield the following results:

- a) five key points will be generated;
- b) three nodes will be generated;
- c) four key points will be generated;
- d) four nodes will be generated.

37 If you sequentially enter three commands in ANSYS

***do,i,1,10,2**

N,i,i,0,0,,,

***enddo**

the execution will yield the following results:

- a) three nodes will be generated;
- b) five nodes will be generated;
- c) six nodes will be generated;
- d) eight nodes will be generated.

38 Determine which problem is described by the equations below:

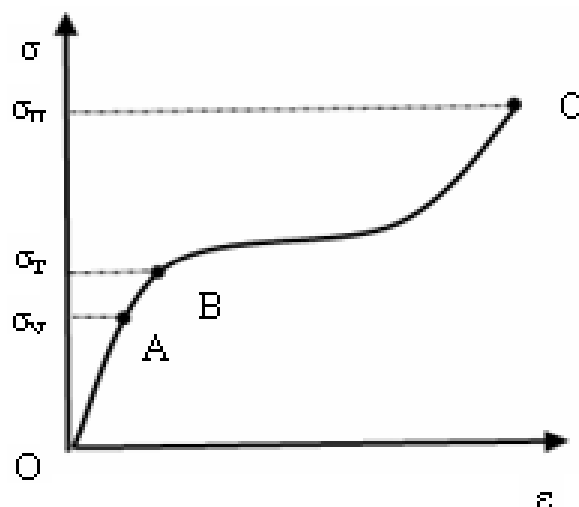
$$\mu \nabla^2 \vec{u} + (\lambda' + \mu) \text{grad} \text{div} \vec{u} - (2\mu + 3\lambda') \alpha_T \text{grad}(T - T_0) = 0,$$

$$c\rho \frac{\partial T}{\partial t} = \lambda \nabla^2 T + Q.$$

- a) coupled dynamic thermoelasticity problem;
- b) uncoupled dynamic thermoelasticity problem;
- c) uncoupled quasi-static thermoelasticity problem.

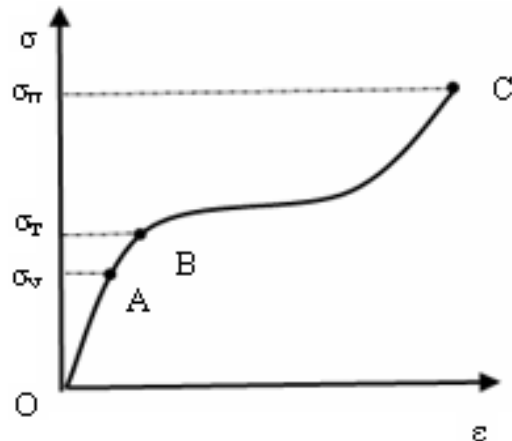
39 The figure shows a deformation diagram. Hooke's law is satisfied in the section(s) of the diagram:

- a) OB;
- b) AB;
- c) OC;
- d) OA;
- e) BC.



40 The figure shows a deformation diagram. The plastic region corresponds to the section(s) of the diagram:

- a) OA;
- a) BC;
- b) OC;
- c) OB;
- d) AB.



41 Which methods does ANSYS provide for creating a model?

- a) creating a geometric model using the ANSYS package;
- b) using direct generation (nodes and elements);
- c) importing models created using CAD tools.

42 Establish a correspondence for using existing nodes when specifying a local coordinate system in ANSYS.

- | | |
|-----------------------------|---|
| 1) the first node defines; | a) the X-Y plane; |
| 2) the second node defines; | b) the origin of the local coordinate system; |
| 3) the third node defines. | c) the direction of the X-axis. |

43 Match the modules in ANSYS with their functions:

- | | |
|--------------|--|
| 1) PREP7; | a) viewing the results obtained for the specified points of the model as a function of time; |
| 2) SOLUTION; | b) creating a model (building geometry, specifying materials, etc.); |
| 3) POST1; | c) viewing the results obtained for the specified point in time; |
| 4) POST26. | d) applying loads and obtaining the FEM calculation. |

44 Match the names of windows in ANSYS with their functional purpose:

- | | |
|---------------------|---|
| 1) Main Menu; | a) contains the primary functions of the ANSYS complex, divided into modules; |
| 2) Graphics Window; | b) the graphic representation window; |
| 3) Output Window. | c) the complex output message window. |

45 Match the names of windows in ANSYS with their functional purpose:

- | | |
|----------------------|---|
| 1) Utility Menu; | a) contains service functions used during the entire session of working with the ANSYS complex; |
| 2) Standard Toolbar; | b) contains graphic buttons that call frequently used commands of the ANSYS complex; |
| 3) Input Window; | c) a changeable (supplemented) toolbar; |
| 4) ANSYS Toolbar. | d) allows direct input of commands. |

46 Match the names of service functions in ANSYS with their functional purpose:

- | | |
|------------|--|
| 1) File; | a) allows you to graphically view points, lines, surfaces, volumes, nodes, elements and other objects that can be displayed graphically; |
| 2) Select; | b) contains functions for using an active set of objects and creating components; |
| 3) List; | c) contains functions for working with files and databases; |
| 4) Plot. | d) allows you to view lists of any objects available in the complex's database. |

47 Match the names of service functions in ANSYS with their functional purpose:

- | | |
|----------------|---|
| 1) PlotCtrls; | a) calls the ANSYS help system; |
| 2) WorkPlane; | b) allows you to call or disable the use of a work plane; |
| 3) Parameters; | c) includes functions for controlling the image, style, and other characteristics of the graphical display; |
| 4) Help. | d) includes functions for creating, editing, and deleting scalar parameters and parameter arrays. |

48 Which of these special characters are prohibited in alphanumeric arguments in ANSYS?

- a) !;
- b) @;
- c) #;
- d) \$;
- e) _.

49 Match the ANSYS file type to its correct extension.

- | | |
|---------------------------------------|-----------------|
| 1) Log file; | a) Jobname.RTH; |
| 2) Error file; | b) Jobname.DB; |
| 3) Database file; | c) Jobname.LOG; |
| 4) Temperature analysis results file. | d) Jobname.ERR. |

50 What is the result of executing the command *GET,ww,NODE,5,LOC,Y in the ANSYS program:

- a) returns the y-coordinate of node number 4 and stores the result as parameter w;
- b) returns the y-coordinate of node number 5 and stores the result as parameter ww;
- c) returns the x-coordinate of node number 5 and stores the result as parameter ww;
- d) returns the z-coordinate of node number 5 and stores the result as parameter w.

Test 8.2 ANSYS WORKBENCH software package

Choose the correct answer(s) for each question from the given options.

1 In ANSYS, calculations can be performed in the following mode(s)...

- a) in batch mode;
- b) in interactive mode;
- c) in list mode.

2 Launching an interactive session of ANSYS is possible...

- a) in the classic version (Classic);
- b) in the ANSYS Workbench environment version;
- c) in the non-classic version (NonClassic).

3 The advantages of the Workbench user shell compared to the classic ANSYS version include...

- a) clarity of modeling;
- b) adaptability to various CAD and CAE packages;
- c) automation of all important operations.

4 The advantages of the Workbench user interface compared to the classic ANSYS version do not include...

- a) visual modeling;
- b) adaptability to various CAD and CAE packages;
- c) automation of all important operations.

5 Choose one or more of the options that enable you to launch ANSYS Workbench.

- a) directly from the CAD program menu;
- b) via the Windows start menu;
- c) via the Ansys Mechanical APDL Launcher desktop shortcut.

6 The ANSYS Workbench start window contains the following elements:...

- a) Project Schematic window;
- b) Toolbar;
- c) Properties panel;
- d) Toolbox panel;
- e) Output Window.

7 The ANSYS Workbench startup window does not contain the following element(s):...

- a) Project Schematic window;
- b) Toolbar;
- c) Properties panel;
- d) Toolbox panel;
- e) Output Window.

8 From the descriptions provided regarding the contents and purposes of the elements within the Workbench user shell start window, select the one that relates to the Project Schematic window.

- a) displays all available modules and applications integrated into the Workbench environment;
- b) serves to display service messages;
- c) combines general settings and project options;
- d) combines general settings and project options that are hidden by default and called by the user through the context menu;
- e) contains structural components of the project (blocks) and links between them.

9 Which description correctly characterizes the Toolbar in ANSYS Workbench?

- a) displays all available modules and applications integrated into the Workbench environment;
- b) serves to display service messages;
- c) combines general settings and project options;
- d) combines general settings and project options that are hidden by default and are called by the user through the context menu;
- e) contains the structural components of the project (blocks) and the links between them.

10 Which description correctly characterizes the Toolbox panel in ANSYS Workbench?

- a) displays all available modules and applications integrated into the Workbench environment;
- b) serves to display service messages;
- c) combines general settings and project options;
- d) combines general settings and project options that are hidden by default and are called by the user through the context menu;
- e) contains the structural components of the project (blocks) and the links between them.

11 Which description correctly characterizes the Properties panel in ANSYS Workbench?

- a) displays all available modules and applications integrated into the Workbench environment;
- b) serves to display service messages;
- c) combines general settings and project options;
- d) combines general settings and project options that are hidden by default and are called by the user through the context menu;
- e) contains the structural components of the project (blocks) and the links between them.

12 Which description correctly characterizes the Messages window in ANSYS Workbench?

- a) displays all available modules and applications integrated into the Workbench environment;
- b) serves to display service messages;
- c) combines general settings and project options;

- d) combines general settings and project options that are hidden by default and are called by the user through the context menu;
- e) contains the structural components of the project (blocks) and the links between them.

13 The Toolbox panel in the Workbench user shell start window contains the following sections:...

- a) Analysis Systems;
- b) Component Systems;
- c) Custom Systems;
- d) Design Exploration;
- e) External Connection Systems;
- f) Messages.

14 Which section is NOT part of the Toolbox panel in ANSYS Workbench?

- a) Analysis Systems;
- b) Component Systems;
- c) Custom Systems;
- d) Design Exploration;
- e) External Connection Systems;
- f) Messages.

15 From the descriptions provided regarding the purpose of the Toolbox panel sections in the Workbench user shell start window, identify the one relevant to the Analysis Systems section.

- a) contains ready-made templates for various types of numerical analysis;
- b) includes the main and auxiliary modules used in solving problems;
- c) contains ready-made template bundles for solving interdisciplinary problems;
- d) allows solving optimization problems;
- e) allows integrating user external applications.

16 Which description correctly characterizes the Component Systems section in the ANSYS Workbench Toolbox?

- a) contains ready-made templates for various types of numerical analysis;
- b) includes the main and auxiliary modules used in solving problems;
- c) contains ready-made template bundles for solving interdisciplinary problems;
- d) allows solving optimization problems;
- e) allows integrating user external applications.

17 Which description correctly characterizes the Custom Systems section in the ANSYS Workbench Toolbox?

- a) contains ready-made templates for various types of numerical analysis;
- b) includes the main and auxiliary modules used in solving problems;
- c) contains ready-made template bundles for solving interdisciplinary problems;
- d) allows solving optimization problems;
- e) allows integrating user external applications.

18 Which description correctly characterizes the Design Exploration section in the ANSYS Workbench Toolbox?

- a) contains ready-made templates for various types of numerical analysis;
- b) includes the main and auxiliary modules used in solving problems;
- c) contains ready-made template bundles for solving interdisciplinary problems;
- d) allows solving optimization problems;
- e) allows integrating user external applications.

19 Which description correctly characterizes the External Connection Systems section in the ANSYS Workbench Toolbox?

- a) contains ready-made templates for various types of numerical analysis;
- b) includes the main and auxiliary modules used in solving problems;
- c) contains ready-made template bundles for solving interdisciplinary problems;
- d) allows you to solve optimization problems;

20 Which description correctly characterizes the Engineering Data module in ANSYS Workbench?

- a) an interface for managing a database of physical and mechanical properties of materials;
- b) an application for creating geometric 2D/3D models;
- c) this module contains the project tree.

21 Which description correctly characterizes Design Modeler in ANSYS Workbench?

- a) an interface for managing a database of physical and mechanical properties of materials;
- b) an application for creating geometric 2D/3D models;
- c) this module contains the project tree.

22 Which description correctly characterizes Multiple Systems – Mechanical (ANSYS Multiphysics) in Workbench?

- a) an interface for managing a database of physical and mechanical properties of materials;
- b) an application for creating geometric 2D/3D models;
- c) this module contains the project tree.

23 Which description correctly characterizes the Static Structural analysis type in ANSYS Mechanical (Multiphysics)?

- a) intended for solving problems of mechanics of a deformable solid in a static formulation;
- b) intended for analyzing a non-stationary thermal field;
- c) intended for analyzing a stationary thermal field.

24 Which description correctly characterizes the Transient Thermal analysis type in ANSYS Mechanical (Multiphysics)?

- a) intended for solving problems of solid mechanics in a static formulation;
- b) intended for analyzing a non-stationary thermal field;
- c) intended for analyzing a stationary thermal field.

25 Which description correctly characterizes the Steady-State Thermal analysis type in ANSYS Mechanical (Multiphysics)?

- a) intended for solving problems of solid mechanics in a static formulation;
- b) intended for analyzing a non-stationary thermal field;
- c) intended for analyzing a stationary thermal field.

BIBLIOGRAPHY

1 Agalakov, Yu. G. Reduction of Data Dimensionality in Simulation Modeling Problems / Yu. G. Agalakov, A. V. Bernshtein // Information Technologies and Computing Systems. – 2012. – No. 3. – P. 3–17.

2 Alawadhi, E. M. Finite Element Simulations Using ANSYS / E. M. Alawadhi. – CRC Press, 2009. – 418 p.

3 Alekseev, V. F. Software for engineering modeling of physical processes. Laboratory practical course: in 2 parts. Part 1: Thermal modes of operation and protection of structures of radio electronic means from mechanical influences : manual / V. F. Alekseev, I. N. Bogatko, G. A. Piskun. – Minsk : BSUIR, 2017. – 124 p.

4 Automated modeling and calculation of structures in ANSYS : one-dimensional models: a tutorial / O. A. Sachenkov [et al.]. – Kazan : Kazan. University, 2019. – 140 p.

5 Basics of Working in ANSYS 17 / N. N. Fedorova [et al.]. – M. : DMK Press, 2017. – 210 p.

6 Basov, K. A. ANSYS : User's Handbook / K. A. Basov. – M. : DMK Press, 2005. – 640 p.

7 Basov, K. A. ANSYS for designers / K. A. Basov. – M. : DMK-Press, 2016. – 248 p.

8 Basov, K. A. ANSYS in examples and problems / K. A. Basov; under the general editorship of D. G. Kraskovsky. – M. : ComputerPress, 2002. – 224 p.

9 Belogurov, E. A. Physics of nano- and microsystems: a methodological manual for practical classes for students of specialties 1-38 01 04 “Micro- and nanosystem technology” and 1-41 01 01 “Technology of materials and components of electronic equipment” / E. A. Belogurov, V. V. Khatko, Ya. I. Shukevich. – Minsk : BNTU, 2011. – 77 p.

10 Burkov, P. V. Computer Modeling of Technologies in the Oil and Gas Industry: Textbook / P. V. Burkov, S. P. Burkova. – Tomsk: Publishing House of Tomsk Polytechnic University, 2012. – 143 p.

11 CAD of optimization tools: methodological recommendations for laboratory work for students of the field of training 23 04 02 “Ground transport and technological complexes” of full-time and part-time forms of study / Interstate Educational Institution of Higher Education “Belarusian-Russian University”, Department of “Transport and Technological Machines”; [compiled by O. V. Leonenko]. – Mogilev: Belarusian-Russian University, 2019. – 42 p.

- 12 Chen X. Finite element modeling and simulation with ANSYS Workbench / X. Chen, Y. Liu. – CRC press, 2018. – 473 p.
- 13 Chigarev, A. V. ANSYS for engineers: a reference manual / A. V. Chigarev, A. S. Kravchuk, A. F. Smalyuk. – M. : Mashino-stroenie, 2004. – 512 p.
- 14 Computer Engineering: A Manual / A. I. Borovkov [et al.]. – St. Petersburg: Publishing House of the Polytechnic University, 2012. – 93 p.
- 15 Computer finite element modeling: a manual for students of the specialty 1-40 05 01 “Information systems and technologies (by areas)” areas of the specialty 1-40 05 01-01 “Information systems and technologies (in design and production)”: in 2 parts / V. V. Naprasnikov [et al.]. – Minsk: BNTU, 2021. – Part 1. – 83 p.
- 16 Computer finite element modeling: a manual in 2 parts. Part 2 / V. V. Naprasnikov [et al.]. – Minsk : BNTU, 2021. – 79 p.
- 17 Computer modeling of technological systems: a tutorial: in 2 parts / S. P. Kundas [et al.]. – Minsk : BSUIR, 2003. – Part 2. – 192 p.
- 18 Denisov, M. A. Mathematical modeling of thermal-physical processes. ANSYS and CAE design: a tutorial / M. A. Denisov. – Ekaterinburg: UrFU, 2011. – 149 p.
- 19 Digital twins in high-tech industry: a brief report (September 2019) / A. I. Borovkov [et al.]. – St. Petersburg: POLYTECH-PRESS, 2019. – 62 p.
- 20 Dyakonov, V. P. Fundamentals of Mathematics and Mathematical Modeling / V. P. Dyakonov, I. V. Abramenkova, A. A. Penkov // New Information Technologies: teaching aid in 5 parts. – Smolensk : SGPU, 2003. – Part 3. – 192 p.
- 21 Dyakonov, V. P. New information technologies: a tutorial: in 3 parts. Part 3. Fundamentals of mathematics and mathematical modeling / V. P. Dyakonov, I. V. Abramenkova, A. A. Penkov. – Smolensk : SGPU, 2003. – 192 p.
- 22 Eliseev, K. V. Computational practical training in modern CAE systems: textbook / K. V. Eliseev, T. V. Zinovieva. – St. Petersburg : Publishing house of the Polytechnic University, 2008. – 112 p.
- 23 Emelianov, V. V. Theory and Practice of Evolutionary Modeling / V. V. Emelianov, V. V. Kureichik, V. M. Kureichik. – M. : FIZMATLIT, 2003. – 432 p.
- 24 Engineering Analysis in ANSYS WORKBENCH: a Tutorial / V. A. Bruyaka [et al.]. – Samara : SamGTU, 2010. – 271 p.
- 25 GOST R 57412-2017. Computer models in the processes of development, production and operation of products. – Introduced. 2017-07-01. – M. : Standartinform, 2018. – 11 p.

- 26 GOST R 57700.22-2020. Computer models and modeling. Classification. – Introduction. 2020-11-13. – M. : Standartin-form, 2020. – 7 p.
- 27 High-tech computer engineering: an overview of markets and technologies / scientific editor K. V. Dorofeev, head of group V. N. Knyagin. – St. Petersburg: Publishing house of the Polytechnic University, 2014. – 110 p.
- 28 Introduction to ANSYS: strength and thermal analysis: study guide / A. S. Shalunov [et al.]. – Kovrov: KGTA, 2002. – 52 p.
- 29 Introduction to computer modeling technologies. ANSYS WORKBENCH shell : a practical guide / compiled by Yu. V. Nikityuk. – Gomel : F. Skorina Gomel State University, 2020. – 34 p.
- 30 Introduction to computer modeling technologies: a practical guide / compiled by: Yu. V. Nikityuk, A. A. Sereda, S. V. Shalupaev; F. Skorina Gomel State University. – Gomel : F. Skorina GSU, 2019. – 42 p.
- 31 Introduction to mathematical modeling: a tutorial / V. N. Ashikhmin [et al.]; edited by P. V. Trusov. - M. : Internet Engineering, 2000. – 336 p.
- 32 Ivanov, D. V. Introduction to Ansys Workbench: a teaching aid. manual for students of natural science disciplines / D. V. Ivanov. – Saratov : Amirit, 2016. – 56 p.
- 33 Ivanov, D. V. Introduction to the Finite Element Method: Textbook and Methodological Manual for Students of Natural Science Disciplines / D. V. Ivanov, A. V. Dol. – Saratov : Amirit, 2016. – 84 p.
- 34 Ivanov, S. E. Intelligent software packages for technical and technological preparation of production / textbook : part 5. Systems of engineering calculation and analysis of parts and assemblies / S. E. Ivanov; edited by D. D. Kulikov – St. Petersburg: St. Petersburg State University of Information Technologies, Mechanics and Optics, 2011. – 48 p.
- 35 Jiang P. Surrogate model-based engineering design and optimization / P. Jiang, Q. Zhou, X. Shao – Berlin : Heidelberg : Springer, 2020. – 240 p.
- 36 Kamensky, A. V. Practical application of the finite element package ANSYS to problems of blood vessel biomechanics: a teaching aid for students of natural sciences / A. V. Kamensky, Yu. V. Salkovsky. – Saratov : Publishing house of Saratov University, 2005. – 105 p.
- 37 Kaplun, A. B. ANSYS in the Hands of an Engineer: Practical Guide / A. B. Kaplun, E. M. Morozov, M. A. Shamraeva. – M. : LENAND, 2024. – 272 p.
- 38 Konyukhov, A. V. Fundamentals of Structural Analysis in ANSYS / A. V. Konyukhov. – Kazan : Kazan State University, 2001. – 142 p.
- 39 Kovalenko, L. D. Fundamentals of Thermoelasticity / L. D. Kovalenko. – Kyiv : Naukova Dumka, 1970. – 307 p.

- 40 Koziel S. Surrogate-based modeling and optimization / S. Koziel, L. Leifsson. – New York : Springer, 2013. – 412 p.
- 41 Kundas, S. P. Computer modeling of technological systems: a tutorial: in 2 parts / S. P. Kundas, T. A. Kashko. – Minsk : BSUIR, 2001. – Part 1. – 164 p.
- 42 Kundas, S. P. Computer modeling of technological systems : textbook : in 2 parts./ S. P. Kundas, B. A. Tonkonogov, T. A. Kashko, N. N. Grinchik. – Minsk. : BSUIR, 2004. – Part 2. – 191 p.
- 43 Laboratory workshop on the discipline “System analysis and mathematical modeling of processes in mechanical engineering” / compiled by A. M. Shchipachev; Ufa State Aviation Technical University. – Ufa : UGATU Publishing House, 2010. – 47 p.
- 44 Logan, D. L. A first course in the finite element method / D. L. Logan. – Cengage Learning, 2016. – 954 p.
- 45 Lukinskikh, S. V. Computer Modeling and Engineering Analysis in Design and Technological Preparation of Production: A Manual / S. V. Lukinskikh. – Ekaterinburg: Publishing House of the Ural. University, 2020. – 168 p.
- 46 Madenci, E. The finite element method and applications in engineering using ANSYS® / E. Madenci, I. Guven. – Springer, 2015. – 663 p.
- 47 Maslov, L. B. Practical training in the course of computational mechanics based on modern software for numerical analysis (ANSYS): a teaching aid / L. B. Maslov, N. A. Sabaneev. – Ivanovo: ISEU, 2009. – 76 p.
- 48 Mathematical modeling of brittle material fracture under thermal loads / M. P. Galanin [et al.]. – M. : Keldysh Institute of Applied Mathematics, 2013. – 36 p. – (Preprint / Keldysh Institute of Applied Mathematics, No. 100).
- 49 Methodological instructions for the implementation of laboratory works No. 1 – 4 on the discipline “Fundamentals of modeling and optimization” for students of the direction 11.03.04 “Electronics and nanoelectronics” (focus “Electronic engineering”) of full-time education / comp. A. A. Vinokurov, A. V. Arsenyev, E. Yu. Plotnikova. – Voronezh : FGBOU VO “Voronezh State Technical University”, 2017. – 33 p.
- 50 Moaveni, S. Finite Element Analysis Theory and Application with ANSYS® / S. Moaveni. – Pearson Education India, 2011. – 527 p.
- 51 Modern engineering education : a textbook / A. I. Borovkov [et al.]. – St. Petersburg : Publishing house of the Polytechnic University, 2012. – 80 p.
- 52 Nasedkin, A. V. Finite element modeling based on ANSYS. Programs for solving static problems of strength of materials with variants of individual tasks / A. V. Nasedkin. – Rostov-on-Don: UPL RSU, 1998. – 44 p.

53 Nasedkin, A. V. Theory and technology of the finite element method : lecture course / A. V. Nasedkin. – Rostov-on-Don : FGAOUVPO “Southern Federal University”, 2011. – 203 p.

54 Nikityuk, Yu. V. Introduction to computer modeling technologies. ANSYS WORKBENCH : a practical guide / Yu. V. Nikityuk, V. A. Prokhorenko, A. A. Sereda. – Gomel: F. Skorina Gomel State University, 2023. – 43 p.

55 Nikityuk, Yu. V. Introduction to computer modeling technology. CALS/PLM, CAE systems : a practical guide / Yu. V. Nikityuk, A. A. Sereda, A. L. Samofalov; Gomel State University named after F. Skorina. – Gomel : F. Skorina State State University, 2023. – 30 p.

56 Okonechnikov, A. S. Strength and Dynamic Calculations in the ANSYS WORKBENCH Software Package: a Tutorial / A. S. Okonechnikov, D. O. Serdyuk, G. V. Fedotenkov. – M. : MAI Publishing House, 2021. – 104 p.

57 Pronin, V. A. Introduction to the ANSYS WORKBENCH calculation platform: laboratory work : Part 1 / V. A. Pronin, D. V. Zhignovskaya, V. A. Tsvetkov. – St. Petersburg : ITMO University, 2019. – 46 p.

58 Samarskii, A. A. Mathematical modeling. Ideas. Methods. Examples / A. A. Samarskii, A. P. Mikhailov. – 2nd ed., corrected. – M. : Fizmatlit, 2001. – 320 p.

59 Shabrov, N. N. Finite Element Method in Calculating Heat Engine Parts / N. N. Shabrov. – L. : Mechanical Engineering, 1983. – 212 p.

60 Shchipachev, A. M. System analysis and mathematical modeling of processes in mechanical engineering : a textbook for university students in the field of 657800 (151000) “Design and technological support for automated machine production”, specialty 120100 (151001) “Technology of mechanical engineering”; directions 651400 (150200) “Engineering technologies and equipment”, specialty 120700 (150206) “Machines and technologies of highly efficient material processing processes” ; directions 653800 (200500) “Standardization, certification and Metrology”, specialty 072000 (200503) “Standardization and certification” ; bachelor’s degree in the direction 552900 (150900) “Technology, equipment and automation of machine-building industries” / A. M. Shchipachev ; Federal Agency for Education, State Educational Institution of Higher Professional Education Ufa State Aviation Technical University university. – Ufa : Ufa State Aviation Technical University, 2008. – 173 p.

61 Shkaruppa, S. P. Application of information and computing technologies in industrial ecology : a textbook / S. P. Shkaruppa ; Samara State Technical University, Chemical Technology and Industrial Ecology. – Samara, 2010. – 244 p. (in Russian)

62 Shimanovsky, A. O. Application of the finite element method in solving problems of applied mechanics: a teaching aid for students of technical specialties / A. O. Shimanovsky, A. V. Putyato. – Gomel : BelGUT, 2008. – 61 p.

63 Shirvel, P. I. Fundamentals of the Finite Element Method in Mechatronics: Textbook-method. manual: in 2 parts / P. I. Shirvel. – Minsk : BNTU, 2015. – Part 1. – 2015. – 89 p.

64 Shirvel, P. I. Mechanics of irreversible deformations: a methodological manual for laboratory work: in 2 parts / P. I. Shirvel. – Minsk : BNTU, 2014. – Part 1. – 176 p.

65 Shiryaev, N. CALS, PDM, PLM, hereinafter – everywhere... / N. Shiryaev // CAD and graphics. – 2002. – No. 12. – P.117–119.

66 Skorobogatova, N. V. Visual modeling of professionally oriented tasks in teaching mathematics to engineering students of technical universities : dissertation... Candidate of Pedagogical Sciences : 13.00.02 / Natalia Vladimirovna Skorobogatova. – Yaroslavl, 2006. – 183 p.

67 Skvortsov, Yu. V. Using the ANSYS FEM package to solve problems in solid mechanics [Electronic resource] : interactive multimedia manual / Yu. V. Skvortsov, S. V. Glushkov. – Mode of access : http://repo.ssau.ru/bitstream/Uchebnye-izdaniya/Ispolzovanie-MKEpaketa-ANSYS-dlya-resheniya-zadach-mehaniki-deformiruemogo-tverdogo-tela-Elektronnyi-resurs-interaktiv-multimed-posobie-72697/1/Skvortsov_Yu.V._Using_the_FEM_package.pdf. – Date of access :13.09.2023.

68 Soldusova, E. A. Basic information about the ANSYS software package. Geometric modeling : a teaching aid / E. A. Soldusova. – Samara : Samara state tech. un-t, 2010. – 54 p.

69 Solodov, A. P. Electronic course of heat and mass transfer. [Electronic resource] // Open education. – 2013. – No. 1 (96). p. 8-16. – [https://doi.org/10.21686/1818-4243-2013-1\(96\)-8-16](https://doi.org/10.21686/1818-4243-2013-1(96)-8-16)

70 Stolarski, Tadeusz. Engineering analysis with ANSYS software / Tadeusz Stolarski, Yuji Nakasone, Shigeka Yoshimoto. – Butterworth-Heinemann, 2018. – 509 p.

71 Thompson, M. K. ANSYS mechanical APDL for finite element analysis / M. K. Thompson, J. M. Thompson. – Butterworth-Heinemann, 2017. – 436 p.

72 Working in the CAE package ANSYS MECHANICAL : structural analysis by the finite element method: method. instructions / compiled by : A. O. Shklovets, V. S. Melentyev. – Samara : Publishing house of Samara University, 2017. – 73 p.

73 Zenkevich, O. K. Finite element method in engineering / O. K. Zenkevich. – M. : Mir, 1975. – 541 p.

Educational publication

Nikityuk Yuri Valerievich,
Sereda Andrei Alexandrovich,
Prokhorenko Vladislav Alexandrovich,
Kovalenko Dmitry Leonidovich,
Samofalov Andrey Leonidovich

INTRODUCTION TO COMPUTER MODELING TECHNOLOGIES

Manual

In the author's edition

Signed for printing: 15.10.2026. Format: 60x84 1/16.
Paper: offset. Printing method: risography.
Conditional printed sheets: 14.18. Academic publishing sheets: 15.25.
Print run: 15 copies. Order No. 506.

Publisher and printer:
Educational Institution
“Francisk Skorina Gomel State University.”
Special permit (license) No. 02330 / 450 dated December 18, 2013.
Certificate of state registration as a publisher, manufacturer,
and distributor of printed publications:
publisher of printed publications No. 1/87 dated November 18, 2013;
distributor of printed publications No. 3/1452 dated April 17, 2017.
104 Sovetskaya St., 246028, Gomel.

