MATHEMATICAL MODELING SYSTEM MATLAB

Dilobar Abdullayeva
Termiz State Pedagogical Institute, Faculty of Mathematics and Informatics, Termiz region, Termiz, 190111, Uzbekistan
E-mail: fevraldilobar@gmail.com

Rustam Malikov
Samarkand State Institute of Foreign Languages,
Samarkand, 140104, UZB, Bustonsaroy street 93.
E-mail: ruslan.malikov.1987@gmail.com

Siroj Amirqulov,
Physics Teacher at the 53th general education school of the Public Education Department of the Nurabad district of the Samarkand region

Axmad Yaxshimurodov
Physics Teacher at the 16th general education school of the Public Education Department of the Pastdargom district of the Samarkand region

Abstract
MatLab is a unique collection of implementations of modern numerical methods of computer mathematics created over the past three decades. She incorporated the experience, rules and methods of mathematical calculations accumulated over thousands of years of the development of mathematics. This is combined with powerful graphical visualization and even animated graphics. The system with the extensive documentation attached to it can well be considered as a fundamental multi-volume electronic handbook on computer mathematical support - from mass personal computers to supercomputers.

1. Introduction

MatLab (short for English «Matrix Laboratory») is a package of application programs for solving technical computing problems and the programming language of the same name used in this package. The MatLab system [1-3] refers to the average level of products intended for symbolic mathematics, but is designed for wide application in the field of CAE (that is, strong in other areas). MatLab is one of the oldest, carefully developed and time-tested systems for automating mathematical calculations, based on an expanded view and application of matrix operations.

In addition, MatLab integrates Simulink for model-based modeling and design for dynamic and embedded systems. It implements the principle of visual programming, allows you to build a logical scheme of a complex control system from standard blocks alone, without writing a line of code. After designing such a scheme, you can analyze its operation in detail.

The combination of MatLab and Simulink programs will create a wide class of professional tool applications (Toolboxes) for generating, analyzing and optimizing systems. This can be used in various industries for various applications.

2. History of the system MatLab

The MatLab system was developed by S.V. Moler in the late 1970s, when he was dean of the Department of Computer Science at the University of New Mexico. The goal of the development was to give faculty students the opportunity to use the Linpack and EISPACK software libraries without the need to study Fortran. Soon, the new language spread among other universities and was met with great interest by scientists working in the field of applied mathematics.

Until now, on the Internet, you can find a 1982 version written in Fortran, distributed with open source. Engineer John N. (Jack) Little became acquainted with this language during Cleave Mowler's visit to Stanford University in 1983. Realizing that the new language has great commercial potential, he teamed up with Cleave Mowler and Steve Bangert. Together, they rewrote MatLab on programming language «C» and founded in 1984 the company TheMathWorks for further development. These rewritten programming language «C» libraries have long been known under the name JACKPAC. MatLab was originally intended for the design of control systems, but quickly gained popularity in many other scientific and engineering fields. It has also been widely used in education, in particular for teaching linear algebra and numerical methods.

One of the main tasks of the system was to provide users with a powerful programming language focused on mathematical calculations and capable of surpassing the capabilities of traditional programming languages that have been used for many years to implement numerical methods. At the same time, special attention was paid to both increasing the speed of calculations and adapting the system to solve a wide variety of user problems.

3. Main features of the system MatLab

MatLab is a high-performance language for technical calculations. It involves computing, visualization and programming in a convenient environment, where problems and solutions are expressed in a form close to mathematical. Typical use of MatLab includes: mathematical calculations, algorithm creation, modeling, data analysis, research and visualization, scientific and engineering graphics, application development, including the creation of a graphical interface

MatLab is an interactive system in which the main data element is an array. This allows you to solve various problems related to technical calculations, especially in which matrices and vectors are used, several times faster than when writing programs using "scalar" programming languages such as programming language «C» or Fortran.

MatLab developed for several years, targeting various users. In the university environment, it was a standard tool for working in various fields of mathematics, mechanical engineering and science. In industry, MatLab is a tool for highly productive research, development and data analysis [4].

In MatLab, specialized groups of programs called toolboxes play an important role. They are very important for most MatLab users, as they allow you to study and use specialized methods. Toolboxes is a comprehensive collection of MatLab (M-file) functions that allow you to solve private task classes.

The MatLab system consists of five main parts.

The language of the MatLab. This is a high-level matrix and array language with flow control, functions, data structures, and object-oriented programming features.

System MatLab. This is a set of tools and devices that the user or programmer MatLab works with. It includes tools for managing variables in the MatLab workspace, input and output, and creating, monitoring, and debugging M-files and MatLab applications.

Managed graphics. This is a graphics system MatLab that includes high-level commands for rendering two- and three-dimensional data, image processing, animation, and illustrated graphics. It also includes low-level commands that allow you to completely edit the appearance of graphics, as well as when creating a graphical user interface (GUI) for MatLab applications.

Library of mathematical functions. This is an extensive collection of computational algorithms from elementary functions, such as sum, sine, cosine, complex arithmetic, to more complex ones, such as reversing matrices, finding eigenvalues, Bessel functions, fast Fourier transform.

Software interface. This is a library that allows you to write programs on C and Fortran that interact with MatLab. It includes means for calling programs from MatLab (dynamic communication), causing MatLab as a computing tool and for reading/writing MAT files.

Simulink, a companion MatLab program, is an interactive system for modeling nonlinear dynamic systems. It is a mouse-controlled environment that allows you to simulate the process by dragging and manipulating blocks of charts on the screen. Simulink works with linear, nonlinear, continuous, discrete, multidimensional systems.

Blocksets are additions to Simulink that provide block libraries for specialized applications such as communication, signal processing, energy systems.

Real-TimeWorkshop is a program that allows «C» code to be generated from blocks of charts and run on various real-time systems.

4. System capabilities MatLab

MatLab capabilities are extensive, and the system often exceeds its competitors in terms of speed. It is applicable for calculations in almost any field of science and technology. For example, it is very widely used in mathematical modeling of mechanical devices and systems, in particular in dynamics, hydrodynamics, aerodynamics, acoustics, energy, etc. This is facilitated not only by an expanded set of matrix and other operations and functions, but also by the presence of the Simulink (Toolbox), specially designed to solve the problems of block modeling of dynamic systems and devices, as well as dozens of other extension packages. The MatLab contains special facilities for electrical and radio calculations (operations with complex numbers, matrices, vectors and polynomials, data processing, signal analysis and digital filtering), image processing, implementation of neural networks, as well as tools related to other new areas of science and technology. They are illustrated by a variety of useful examples. The development of extensions for the system MatLab involved many scientific schools of the world and large scientists and university teachers leading them.

There are also extensive programming options in the MatLab system. Its «C» Math library (compiler MatLab) is object-based and contains more than 300 data processing procedures in language C. Within the package, you can use both the procedures of the MatLab itself and the standard procedures of language C, which makes this tool a powerful help in developing applications (using the
«C» Math compiler, you can embed any MatLab procedures in ready-made applications).

The C Math library allows you to use the following categories of functions: operations with matrices; matrix comparison; solving linear equations; Decomposition of operators and search for eigenvalues finding the inverse matrix; finding a determinant; calculation of matrix exponential; elementary mathematics; beta, gamma, erf, and elliptical functions; a framework for statistics and data analysis; Search for polynomial roots Fast Fourier Transform (FFT) interpolation etc.

At the same time, all MatLab libraries are distinguished by a high rate of numerical calculations. The versatility of the matrix calculus apparatus significantly increases interest in the MatLab system, which includes the best achievements in the field of fast matrix problem solving.

For more detailed calculations and visualization of modeling, the MatLab system has a number of unique (Toolboxes).

The main Toolboxes Matlab: artificial intelligence (Statistics and Machine Learning Toolbox, Deep Learning Toolbox, Text Analytics Toolbox, Predictive Maintenance Toolbox); artificial intelligence and parallel computing (Parallel Computing Toolbox, MatLab Parallel Server); mathematics and optimization (Curve Fitting Toolbox, Optimization Toolbox, Global Optimization Toolbox, Symbolic Math Toolbox, Mapping Toolbox, Partial Differential Equation Toolbox); generation of the code (MatLab Coder, Embedded Coder, HDL Coder, HDL Verifier, Filter Design HDL Coder, Fixed-Point Designer, GPU Coder); deploy applications and algorithms (MatLab Compiler, MatLab Compiler SDK, MatLab Production Server, MatLab Web App Server); databases and report automation (Database Toolbox, MatLab Report Generator); family products (Polyspace, Polyspace Bug Finder, Polyspace Code Prover); cloud solutions (MatLab Online, MatLab Drive, ThingSpeak, MatLab Mobile, MatLab Grader, MatLab in the cloud).

For example, the System Identification Toolbox is an application for building mathematical models of dynamic systems based on measured input-output of data. It supports both parametric and nonparametric methods. This allows you to create and use dynamic system models that are not easily modeled from the first principles or specifications. The toolkit also provides algorithms for built-in online parameter estimation.

The Deep Learning Toolbox (formerly the Neural Network Toolbox) provides a framework for designing and implementing deep neural networks using both pre-trained models and applications and tools for designing and developing neural network architectures.

The Parallel Computing Toolbox allows you to solve computational and demanding tasks using multi-core processors, graphics processors and computer clusters.

The Curve Fitting Toolbox provides an interactive application and features for fitting curves and surfaces to data.

The Partial Differential Equation Toolbox provides functions for solving structural mechanics, heat exchange, and common partial derivative (PDE) equations using finite element analysis.

The main Toolboxes of the Simulink system based on Matlab: event modeling (Stateflow, SimEvents); physical modeling (Simscape, Simscape Driveline, Simscape Electrical, Simscape Fluids, Simscape Multibody, Simulink Desktop Real-Time); real-time systems (Simulink Real-Time, Simulink Desktop Real-Time); generation of the code (Simulink Coder, Embedded Coder, AUTOSAR Blockset, Fixed-Point Designer, Simulink PLC Coder, Simulink Code Inspector, DO Qualification Kit (для DO- 178), IEC Certification Kit (для IEC 61508 и ISO 26262), HDL Coder, HDL Verifier); verification and validation (Simulink Requirements, Simulink Check, Simulink Coverage, Simulink Design Verifier, Simulink Test); 3D Simulink (3D Animation, Simulink Report Generator); deploy applications and algorithms (Simulink Compiler).

Simscape is designed to create models of physical systems in Simulink.

Simscape Driveline (formerly SimDriveline) provides content libraries for modeling and simulating rotational and translational mechanical systems. The tool contains models of worm gears, propellers and components of vehicles, such as engines, tires, transmissions and torque transformers. These components are used to simulate the transmission of mechanical power in transmissions of helicopters, industrial equipment, cars and other machines.

Simscape Fluids (formerly SimHydraulics) provides a library of components for modeling and simulation of hydraulic systems. Simscape Fluids includes models of hydraulic pumps, valves, actuators, piping and heat exchangers.

Simscape Multibody (before SimMechanics) is a Wednesday of modeling for three-dimensional mechanical systems, such as robots, pendants of vehicles, construction equipment and chassis of planes. Multi-unit systems are modeled with units describing bodies, joints, constraints, force elements and sensors. Simscape Multibody formulates and solves equations of motion for the entire mechanical system. You can import complete CAD assemblies into your models, including all masses, inertia, joints, constraints, and 3D geometry. An automatically generated 3D animation allows you to visualize system dynamics.

The unique features of MatLab are:

• Full-fledged framework for developing systems and training algorithms with artificial intelligence on big data for machine vision, signal processing, control systems and text data analysis.

• Designing embedded algorithms through simulation and automatic code generation for control and signal processing systems.

• Establishment of test stands based on instrumentation and data collection devices for test automation and regression testing.

• Modeling by systems of differential equations in numerical and symbolic form, statistical modeling and solving mathematical optimization problems.

• Seamless deployment of algorithms on user machines, enterprise servers and web resources in the form of programs or libraries.

• Using the script language in Simulink allows you to automate a wide range of tasks: retrieving parameters using the Monte Carlo method, analyzing the sensitivity of model parameters, paralleling models, matching optimal parameters, running scripts on any trigger in the model, starting verification and code generation algorithms, as well as automatic model building.

Popular mathematical systems such as Mathcad, Maple V, and Mathematica can integrate with the MatLab system.

In turn, many other mathematical systems, such as Mathcad and Maple, allow the establishment of object and dynamic links with the MatLab system, which allows them to use effective MatLab tools to work with matrices. This progressive trend of integrating computer mathematical systems will undoubtedly continue.

5. Visualization and graphics MatLab

Recently, the creators of mathematical systems have paid great attention to visualizing the solution of mathematical problems. Simply put, this means that the setting and description of the problem to be solved and the results of the solution should be extremely clear not only to those who solve the problems, but also to those who later study them or simply view them. A large role in visualizing the solution of mathematical problems is played by a graphical representation of the results, both finite and intermediate.

Visualization of the task setting in the MatLab is solved by using Notebook and assigning sufficiently clear names (IDs) to function names. And the visualization of the results of calculations is achieved using extensive means of graphics, including animation, as well as using (where necessary) the means of symbolic mathematics.

Graphics of almost all types known in science and technology are implemented at an increased speed. Using descriptor graphics allows you to create typical elements of the user interface - buttons, menus, information and dashboard, etc., that is, to implement elements of visual-oriented programming.

The schedules are displayed separately from the texts in the individual windows. In one graph, you can represent many curves that differ in color (with a color display) and distinctive characters (circles, crosses, rectangles, etc.). Graphs can be displayed in one or more windows. Compared to Mathcad, 3D shapes are built almost an order of magnitude faster by MatLab. In addition, when constructing such graphs, a sufficiently perfect algorithm for removing invisible lines is used, which, along with the large size of the graphs and the possibility of interpolation in color, makes the constructed three-dimensional surfaces and figures very aesthetic and visual. Shapes can be transparent.

In the USA, MatLab is taught in almost every university and is the standard for engineering development. In Russia, MatLab is used in all leading universities.

MatLab has a number of competitors: Mathematica, Maple, IDL from ITT Visual Information Solutions and Metlynx. There are also open source applications that are an alternative to MatLab, in particular GNU Octave, FreeMat and Scilab.

Connecting models of various mechanical systems in Simulink allows you to visualize the model of motion dynamics and control. The obtained dynamics and visualization models are convenient for quick prototyping and application in educational [5,6], presentation, demonstration and research purposes [7].

6. Simulink Mathematical Modeling

In the Simulink package of the MatLab environment, there is a separate Simscape Multibody library designed to simulate the mechanical motion of solids. In MatLab 7, this library is represented by the SimMechanics library. Its main purpose is to model the spatial movements of solid-state machines and mechanisms at the stage of engineering design using the laws of theoretical mechanics [8, 9].

When using the SimMechanics library integrated in Simulink, you can use all the capabilities of the MatLab system, in particular, adding components from other Simulink libraries and system extensions to the mechanical system model. The SimMechanics package allows solving spatial problems of static, kinematics and dynamics of ladder mechanical objects.

By means of SimMechanics in Simulink can be carried to advantages of realization of modeling of mechanical systems simplicity of creation of models by not too prepared users and the high speed of calculations when modeling movements of multilink objects with large number of degrees of freedom in big conveyances. The mechanical system is represented the connected block chart, like other Simulink models, with use of blocks from SimMechanics library. Mass and inertial properties of bodies (links of mechanisms), coordinates of characteristic points act as the set parameters of mechanical blocks bodies (such as centers of masses, points of application of the external and operating influences, points of connection of hinges and joints). Unlike other Simulink blocks which perform mathematical operations or process signals, mechanical SimMechanics blocks represent directly physical bodies or communications between them. The modelled mechanical systems can consist of any quantity of the solids connected by the hinges having forward and rotary degrees of freedom. SimMechanics can model mechanisms with the links organized in hierarchical structures as well as in the usual Simulink models. Imposing of kinematic restrictions, forces and driving torques, mutual trajectories of motion of bodies is possible.

In SimMechanics the term «mechanism» has two values. First, it designates physical system which includes, at least, one solid. Also it designates topologically the separate block chart presenting one physical mechanism. The model can include one or more mechanisms. The SimMechanics model consists of the block chart which, in turn, consists of one or more mechanisms, each of which represents set of the knots connected among themselves presenting the only physical mechanism. For example, the following model (figure 1) contains two mechanisms.

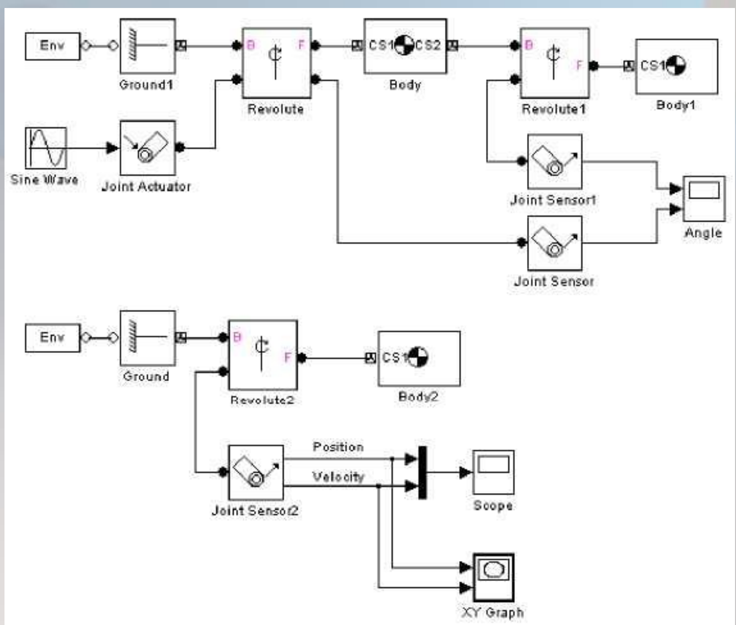


Figure 1. Example of a two- mechanism SimMechanics model.

The SimMechanics model differs significantly from other Simulink models in the way it depicts the mechanism. The usual Simulink model displays mathematical data of the motion of the mechanism, that is, algebraic and differential equations that calculate the future state of the mechanism depending on its real state.

The Simulink mathematical model allows you to simulate a mechanism. In contrast, the SimMechanics model depicts the physical structure of the mechanism, the geometric and kinematic relationships of its components. SimMechanics automatically converts this structure image to an internal, equivalent mathematical model. This saves the time and effort of the researcher to independently develop a mathematical model.

You create a SimMechanics model in the same way as you create any other Simulink model. The Simulink model window opens first. Then, you drag samples of SimMechanics and other Simulink blocks into the user model window and draw lines to connect the blocks to each other.

The main body blocks display the components of the mechanism and the fixed environment of the mechanism (base). Joint knot blocks represent degrees of freedom of one part of the mechanism relative to the other or the base. Constraints & Drivers fixed and non-stationary linkage blocks constrain or trigger movements of parts of the mechanism relative to each other. Actuators drive units define the applied forces, motions, changing mass and inertia, or initial conditions for parts of the mechanism, hinge assemblies, and bonds. Force Elements force elastic element units simulate forces between parts of the mechanism.

Sensor units of Sensors measure the forces or movements of parts of the mechanism, joints and connections.

Built-in visual observation tools SimMechanics allow you to start 3-D animation of an object during modeling using the graphical capabilities of the MatLab system.

The Custom Joint unit occupies a special position. Unlike all other blocks, it allows the user to set the number and order of primitives (and degrees of freedom) in the hinge, that is, to design his own chain of primitives. The number of degrees of freedom cannot be greater than the maximum possible - six.

Disassembled Joints (figure 2) contains a set of joints that the SimMechanics automatically collects at the beginning of the simulation. That is, the SimMechanics at the beginning of the simulation positions these hinges so that they meet assembly constraints determined by the type of hinge, for example, translational (Prismatic) or rotational (Revolute). Using these hinges eliminates the need for the user to assemble closed loops in the mechanism (manual adjustment of linear and angular dimensions in a chain of links with closed topology).

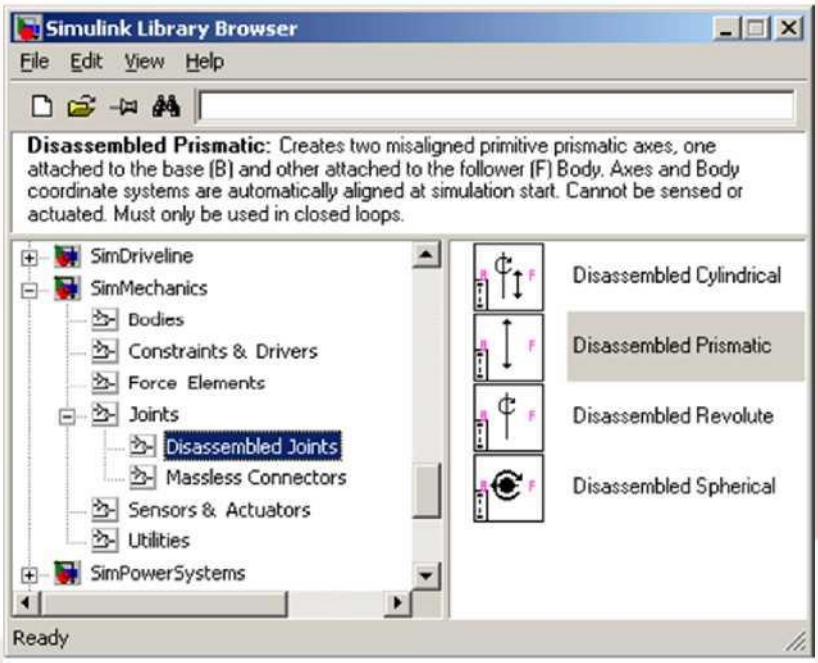


Figure 2. Disassembled joint units Disassembled Joints.

Disassembled hinges are only four types:

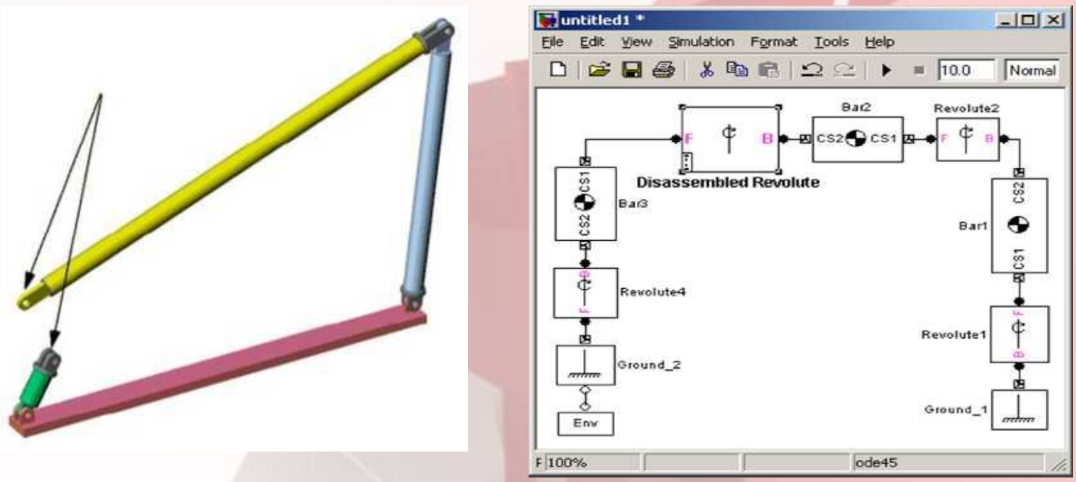
- Disassembled Prismatic;
- Disassembled Revolute;
- Disassembled Cylindrical;
- Disassembled Spherical.

Disassembled elementary hinges differ from similar assembled hinges in number of specified axes. The assembled elementary primitive has only one translation axis, or one rotation axis, or one spherical point. An exploded translational or rotational primitive has two axes of translational motion or rotation: one for a base body (Base) and the other for a Follower body (Follower). The disassembled spherical primitive also has two points.

All disassembled joints can be used only in closed circuits of the mechanism (if any). Each closed loop can contain only one disassembled hinge.

Before starting the simulation, the SimMechanics assembles the mechanism and can move the bodies associated with the disassembled hinges from their positions to assemble the hinges. The solution to the SimMechanics assembly problem cannot be predicted in advance except in simple cases. To prevent SimMechanics from moving bodies during assembly, you must use the Joint Initial Condition start condition blocks described below, which define the initial position of the bodies fixed during assembly. This will cause SimMechanics to find an assembly solution that satisfies the initial hinge movements defined by the Joint Initial Condition blocks.

An example of a closed loop where a disassembled Disassembled Revolution hinge can be used is shown in figure 3. Sketches of disassembled hinges explaining their purpose are given in figure 4.



(a) (b)

Figure 3. Sketch of four-link mechanism: (a) - forming a closed loop; (b) - SimMechanics model using Disassembled Revolute disassembled hinge.

(a) (b) (c) (d)

Figure 4. Sketches of disassembled hinges: (a) - Disassembled Prismatic; (b) - Disassembled Revolute; (c) - Disassembled Cylindrical; (d) - Disassembled Spherical.

7. Conclusions

The use of cyclic functions in the synthesis of centroids of gears allows you to obtain round and nonround gears with a corresponding coefficient of «non-circular» k. The obtained design dependencies make it possible to obtain non-circular gears with a different number of «waves». The use of mathematical and graphic software products allows you to automate the synthesis of teeth on the corresponding centroids.

References

1. Urinov, K. O., Jumanov, K. A., Khidirov, A. M., Urinov, S. K., Abdiyev, J. M., Jumaboyev, T. A., & Eshmirzayev, M. R. (2020, April). Magnetocaloric effect in polycrystalline cobalt. In Journal of Physics: Conference Series (Vol. 1515, No. 2, p. 022079). IOP Publishing.
2. Saidov, A. S., Saparov, D. V., Usmonov, S. N., Kutlimratov, A., Abdiev, J. M., Kalanov, M., ... & Akhmedov, A. M. (2021). Investigation of the Crystallographic Perfection and Photoluminescence Spectrum of the Epitaxial Films of (Si2) 1-x (GaP) x Solid Solution, Grown on Si and GaP Substrates with the Crystallographic Orientation (111). Advances in Condensed Matter Physics, 2021.

3. LEYDERMAN, A., Saidov, A. S., USMONOV, S., Abdiyev, J. M., & Suyarov, Q. T. (2021, February). INFLUENCE OF WEAK GRADED GAP ON INJECTION DIFFUSION REGIMES OF CURRENT TRANSPORT IN SEMICONDUCTOR PN-STRUCTURES. In Congress Dates (p. 40).
4. Raufovich, M. R., & Muzaffar o'g'li, A. J. (2022). THE ROLE OF TEACHER AND STUDENT COOPERATION IN THE MODERN EDUCATIONAL PROCESS OF GOOGLE DOCS. Web of Scientist: International Scientific Research Journal, 3(1), 579-589.
5. Abdiev, J., Safarov, O., & Julanov, H. (2022). Study of the properties of polymer composites–reinforcement based on glass and basalt fibers. Eurasian Scientific Herald, 7, 77-88.
6. Urinov, K. O., Jumanov, K. A., Khidirov, A. M., Urinov, S. K., Abdiyev, J. M., Jumaboyev, T. A., & Eshmirzayev, M. R. (2020). Magnetocaloric effect in polycrystalline cobalt. In Journal of Physics: Conference Series (Vol. 1515, p. 022079).
7. Urinov, K. O., Jumanov, K. A., Khidirov, A. M., Urinov, S. K., Abdiyev, J. M., Jumaboyev, T. A., & Eshmirzayev, M. R. (2020). Magnetocaloric effect in polycrystalline cobalt. In JOP Conference Series: Metrological Support of Innovative Technologies (pp. 22079-22079).
8. Urinov, K. O., Jumanov, K. A., Khidirov, A. M., Urinov, S. K., Abdiyev, J. M., Jumaboyev, T. A., & Eshmirzayev, M. R. (2020, April). Magnetocaloric effect in polycrystalline cobalt. In Journal of Physics: Conference Series (Vol. 1515, No. 2). IOP Publishing.
9. Urinov, K. O., Jumanov, K. A., Khidirov, A. M., Urinov, S. K., Abdiyev, J. M., Jumaboyev, T. A., & Eshmirzayev, M. R. (2020, April). Magnetocaloric effect in polycrystalline cobalt. In Journal of Physics Conference Series (Vol. 1515, No. 2, p. 022079).
10. Saidov, A. S., Saparov, D. V., Usmonov, S. N., Kutlimratov, A., Abdiev, J. M., Kalanov, M., ... & Akhmedov, A. M. (2021). Investigation of the Crystallographic Perfection and Photoluminescence Spectrum of the Epitaxial Films of (Si2) 1-x (GaP) x Solid Solution, Grown on Si and GaP Substrates with the Crystallographic Orientation (111). Advances in Condensed Matter Physics, 2021.
11. Bolkievna, I. A. LEARNING FROM UZBEK AND PERSIAN TRADITIONS: THE ROLE OF ENVIRONMENTAL PERCEPTIONS IN THE CREATION OF SPACE IS A GARDEN-CREATING PARADISE.
12. Saidov, A. S., Saparov, D. V., Usmonov, S., Kutlimratov, A., Abdiev, J. M., Kalanov, M., ... & Akhmedov, A. M. (2021). Investigation of the Crystallographic Perfection and Photoluminescence Spectrum of the Epitaxial Films of (Si 2) 1-x (GaP) x 0≤ x≤ 1 Solid Solution, Grown on Si and GaP Substrates with the Crystallographic Orientation (111). Advances in Condensed Matter Physics, 2021.

13. Saidov, A. S., Saparov, D. V., Usmonov, S. N., Kutlimratov, A., Abdiev, J. M., Kalanov, M., ... & Akhmedov, A. M. (2021). Investigation of the Crystallographic Perfection and Photoluminescence Spectrum of the Epitaxial Films of (Si2)(GaP) x (0≤ x≤ 1) Solid Solution, Grown on Si and GaP Substrates with the Crystallographic Orientation (111).
14. Saidov, A. S., Saparov, D. V., Usmonov, S. N., Kutlimratov, A., Abdiev, J. M., Kalanov, M., & Razzakov, A. S. (2021). Investigation of the Crystallographic Perfection and Photoluminescence Spectrum of the Epitaxial Films of (Si [sub. 2])[sub. 1-x](GaP)[sub. x](0≤ x≤ 1) Solid Solution, Grown on Si and GaP Substrates with the Crystallographic Orientation (111). Advances in Condensed Matter Physics, 2021.
15. Saidov, A. S., Saparov, D. V., Usmonov, S. N., Kutlimratov, A., Abdiev, J. M., Kalanov, M., ... & Akhmedov, A. M. (2021). Investigation of the Crystallographic Perfection and Photoluminescence Spectrum of the Epitaxial Films of (Si 2) 1-x (GaP) x 0≤ x≤ 1 Solid Solution, Grown on Si and GaP Substrates with the Crystallographic Orientation (111). Advances in Condensed Matter Physics.
16. LEYDERMAN, A. Y., SAIDOV, A., USMONOV, S. N., ABDIYEV, J., & SUYAROV, Q. (2021, February). INFLUENCE OF WEAK GRADED GAP ON INJECTION DIFFUSION REGIMES OF CURRENT TRANSPORT IN SEMICONDUCTOR PN-STRUCTURES. In Congress Dates (p. 40).
17. LEYDERMAN, A. Y., SAIDOV, A., USMONOV, S. N., ABDIYEV, J., & SUYAROV, Q. (2021, February). INFLUENCE OF WEAK GRADED GAP ON INJECTION DIFFUSION REGIMES OF CURRENT TRANSPORT IN SEMICONDUCTOR PN-STRUCTURES. In Congress Dates (p. 40).
18. Leiderman, A. Y., Saidov, A. S., & Abdiyev, J. M. (2021). INJECTION DIFFUSION PROCESSES IN THE WEAK LINEAR GRADED-BAND SEMICONDUCTOR pn-STRUCTURES. Euroasian Journal of Semiconductors Science and Engineering, 3(1), 3.
19. Leiderman, A. Y., Saidov, A. S., & Abdiyev, J. M. INJECTION DIFFUSION PROCESSES IN THE WEAK LINEAR GRADED-BAND SEMICONDUCTOR pn-STRUCTURES. Journal of Semiconductors, 3(1), 3.
20. Abdiev, J., Safarov, O., & Julanov, H. (2022). Study of the properties of polymer composites–reinforcement based on glass and basalt fibers. Eurasian Scientific Herald, 7, 77-88.
21. Abdiev, J., Safarov, O., & Julanov, H. Study of the properties of polymer composites–reinforcement based on glass and basalt fibers.
22. Abdiev, J., & Safarov, O. (2022). BASALT FIBER-BASIC (PRIMARY) CONCEPTS. Web of Scientist: International Scientific Research Journal, 3(4), 212-240.
23. Abdiev, J., & Safarov, O. BASALT FIBER-BASIC (PRIMARY) CONCEPTS.KUŞÇULU, N. (2021). Congress Dates.

24. Abdiev, J., Safarov, O., & Julanov, H. (2022). Study of the properties of polymer composites–reinforcement based on glass and basalt fibers. Eurasian Scientific Herald, 7, 77-88.
25. Abdiev, J., Safarov, O., & Julanov, H. (2022). Study of the properties of polymer composites–reinforcement based on glass and basalt fibers. Eurasian Scientific Herald, 7, 77-88.
26. al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM.
27. Abdiev, J., Safarov, O., & Julanov, H. Study of the properties of polymer composites–reinforcement based on glass and basalt fibers.
28. al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM.
29. Bolkievna, I. A., & AbdievMuzaffar o'g'li, J. (2022). LEARNING FROM UZBEK AND PERSIAN TRADITIONS: THE ROLE OF ENVIRONMENTAL PERCEPTIONS IN THE CREATION OF SPACE IS A GARDEN-CREATING PARADISE. Web of Scientist: International Scientific Research Journal, 3(4), 188-211.
30. Bolkievna, I. A. LEARNING FROM UZBEK AND PERSIAN TRADITIONS: THE ROLE OF ENVIRONMENTAL PERCEPTIONS IN THE CREATION OF SPACE IS A GARDEN-CREATING PARADISE.
31. KUŞÇULU, N. (2021). Congress Dates.
32. Abdiev, J., & Safarov, O. BASALT FIBER-BASIC (PRIMARY) CONCEPTS.
33. KUŞÇULU, N. (2021). Congress Dates.
34. Xusanbek Ulug'bek o'g, M., al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. (2022, June). DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM. In E Conference Zone (pp. 83-99).
35. KUŞÇULU, A. P. D. N. Access Date: 22.06. 2021.
36. Xusanbek Ulug'bek o'g, M., al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. (2022, June). DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM. In E Conference Zone (pp. 83-99).
37. Xusanbek Ulug'bek o'g, M., al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. (2022, June). DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM. In E Conference Zone (pp. 83-99).

38. al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM.
39. Xusanbek Ulug'bek o'g, M., al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. (2022, June). DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM. In E Conference Zone (pp. 83-99).
40. al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM.
41. Abdullayeva, D. O. T., Ismoilov, M. M., qizi Abdieva, N. M., & Khasanova, D. Y. (2022). Peach signal control software is an important part of our lives. Eurasian Scientific Herald, 7, 145-152.
42. Abdullayeva, D. O. T., Ismoilov, M. M., qizi Abdieva, N. M., & Khasanova, D. Y. (2022). Peach signal control software is an important part of our lives. Eurasian Scientific Herald, 7, 145-152.
43. Taylanov, N. A. (2022). A Fourth-Order Runge-Kutta Method for Numerical Solution of the Kuramoto-Sivashinsky Equation. Eurasian Scientific Herald, 7, 58-61.
44. Taylanov, N. A. (2022). A Fourth-Order Runge-Kutta Method for Numerical Solution of the Kuramoto-Sivashinsky Equation. Eurasian Scientific Herald, 7, 58-61.
45. Abdurazzokovich, N., Xudoyor o'g'li, S., & Muzaffar o'g'li, J. A Fourth-Order Runge-Kutta Method for Numerical Solution of the Kuramoto-Sivashinsky Equation.
46. Abdiev, J., Abdieva, N. M., & Khasanova, D. Y. (2022). Physical Terms, Problems and Their Solutions. Problems and Their Solutions (March 25, 2022).
47. Ortikovich, U. J., Muzaffar o'g'li, A. J., & Rafikovich, M. R. **Fex V**3− x **O4**MAGNETIC AND ELECTRICAL PROPERTIES OF SPINEL.
48. Ortikovich, U. J., Muzaffar o'g'li, A. J., & Rafikovich, M. R. **Fex V**3− x **O4**MAGNETIC AND ELECTRICAL PROPERTIES OF SPINEL.
49. KUŞÇULU, N. (2021). Congress Dates.
50. KUŞÇULU, N. (2021). Congress Dates.
51. KUŞÇULU, N. (2021). Congress Dates.
52. Safarov, O. (2022). UNDERSTANDING THE MAGNETO CALORIC EFFECT IN SUPER SPIN GLASS COBALT-BASED NANOPARTICLES. Web of Scientist: International Scientific Research Journal, 3(7), 545-563.
53. Qo'Chqarovna, M. U., & Safarali, D. (2022). Theory of dependence of ultrasound absorption (amplification) in semiconductors on electron scattering mechanism. Web of Scientist: International Scientific Research Journal, 3(7), 480-489.

54. Qo'chqarovna, J. H., & Safarov, O. UNDERSTANDING THE MAGNETO CALORIC EFFECT IN SUPER SPIN GLASS COBALT-BASED NANOPARTICLES.
55. Qo'Chqarovna, M. U., Safarali, D., & Ugli, A. J. M. (2022). Theory of dependence of ultrasound absorption (amplification) in semiconductors on electron scattering mechanism.
56. Xusanbek Ulug'bek o'g, M., al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. (2022, June). DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM. In E Conference Zone (pp. 83-99).
57. Qo'chqarovna, M. U., & Safarali, D. THEORY OF DEPENDENCE OF ULTRASOUND ABSORPTION (AMPLIFICATION) IN SEMICONDUCTORS ON ELECTRON SCATTERING MECHANISM.
58. al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM.
59. Xusanbek Ulug'bek o'g, M., al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. (2022, June). DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM. In E Conference Zone (pp. 83-99).
60. Abdullayeva, D. O. T., Ismoilov, M. M., qizi Abdieva, N. M., & Khasanova, D. Y. (2022). Peach signal control software is an important part of our lives. Eurasian Scientific Herald, 7, 145-152.
61. al-Khwarizmi, M., Abdullayeva, D. O. T., qizi Abdieva, N. M., & Khasanova, D. Y. DEVELOPMENT OF ALGORITHMS AND PROGRAMS FOR PROCESSING SPEECH SIGNALS ON VISUAL DSP++ PLATFORM.
62. Abdullayeva, D. O. T., Ismoilov, M. M., qizi Abdieva, N. M., & Khasanova, D. Y. (2022). Peach signal control software is an important part of our lives. Eurasian Scientific Herald, 7, 145-152.