Technical Sciences

UDC 681.5:519.876.5

Allakhveranov Rauf

Candidate of Technical Sciences, Associate Professor

Kharkiv National University of Radio Electronics

Hubskiy Ihor

Master Student of the

Kharkiv National University of Radio Electronics

DOI: 10.25313/2520-2057-2026-4-12112

ANALYSIS OF METHODS FOR MATHEMATICAL MODELING OF DISCRETE AUTOMATION OBJECTS

Sumarry. *This article analyzes methods for the mathematical modeling of discrete automation objects. It examines approaches to constructing mathematical models based on differential equations, state-space models, transfer functions, and numerical methods. The article analyzes the specifics of using the Laplace transform and software tools for mathematical modeling. A comparative analysis of existing methods is performed, and their advantages and limitations in the study of automatic control systems are identified. It is shown that the integrated use of analytical and numerical methods improves the accuracy of mathematical models and the efficiency of automation system development.*

Key words: *mathematical modeling, discrete automation object, state space, differential equations, Laplace transform, numerical methods, automatic control.*

Introduction. Mathematical modeling is one of the fundamental tools for researching, analyzing, and designing automation systems. Its application makes it possible to investigate the dynamic properties of controlled systems, evaluate their behavior under various operating conditions, synthesize control algorithms, and reduce the costs of conducting full-scale experiments. Mathematical modeling becomes particularly relevant in the development of robotic systems, cyber-physical systems, and digital manufacturing, where high demands are placed on the accuracy and adequacy of models.

Discrete automation objects are characterized by complex structures, the presence of nonlinearities, time delays, and the influence of external disturbances, which necessitates the use of effective mathematical modeling methods. In this regard, it is important to analyze approaches to constructing mathematical models, evaluate their capabilities, and identify areas of practical application. The purpose of this article is to analyze methods of mathematical modeling for discrete automation objects, compare them, and identify the specific features of their use in the development of automatic control systems.

Analysis of Recent Research and Publications. The issue of mathematical modeling of discrete automation objects is one of the priority areas in the development of automatic control theory. The construction of adequate mathematical models enables the analysis of dynamic processes, the study of system stability, the synthesis of control algorithms, and the optimization of technical object parameters. The foundation of modeling methods consists of differential equations, state-space models, transfer functions, and numerical methods for solving mathematical models.

In the research conducted, significant attention is paid to the application of computer technologies for mathematical modeling, which allow for the automation of the process of model construction, the performance of numerical calculations, and the analysis of results. The use of software packages such as Scilab, Maxima, MATLAB, and Python significantly expands the possibilities for studying complex dynamic systems and increases the efficiency of automation system development.

At the same time, an analysis of scientific publications shows that most studies focus on individual methods of mathematical modeling, while the issues of their comprehensive comparison, identification of advantages, limitations, and areas of practical application remain insufficiently addressed. This underscores the relevance of conducting a comprehensive analysis of mathematical modeling methods for discrete automation objects and evaluating their potential for use in the development of automatic control systems.

Statement of the Research Problem. The purpose of this article is to analyze methods of mathematical modeling of discrete automation objects, compare their main characteristics, and identify the features of their practical application in the development of automatic control systems.

To achieve this goal, the following tasks must be addressed:

– analyze existing approaches to the mathematical modeling of discrete automation objects;

– to investigate the characteristics of using models based on differential equations, state spaces, and transfer functions;

– to perform a comparative analysis of numerical modeling methods;

– to investigate the capabilities of mathematical modeling software tools;

– to identify promising directions for the development of mathematical modeling methods in automation systems.

Methods of Mathematical Modeling of Discrete Automation Objects. Mathematical modeling is one of the most important stages in the development of automatic control systems, as it allows us to describe the behavior of a real-world object using mathematical relationships, investigate its dynamic properties, and evaluate its response to external influences. The use of mathematical models makes it possible to analyze a system as early as the design stage, which significantly reduces the time and resources required to develop new technical solutions. The main requirements for mathematical models are adequacy, accuracy, versatility, and the ability to be efficiently implemented on a computer.

Existing discrete automation systems are characterized by a large number of interconnected elements, digital information processing, and the discrete nature of changes in control signals. Various methods of mathematical modeling are used to study them; the choice of method depends on the complexity of the system, the required accuracy of calculations, and the conditions of practical application. The most widely used models are those based on differential equations, transfer functions, state-space models, and numerical integration methods.

Figure 1 shows a classification of methods for mathematical modeling of discrete automation systems.

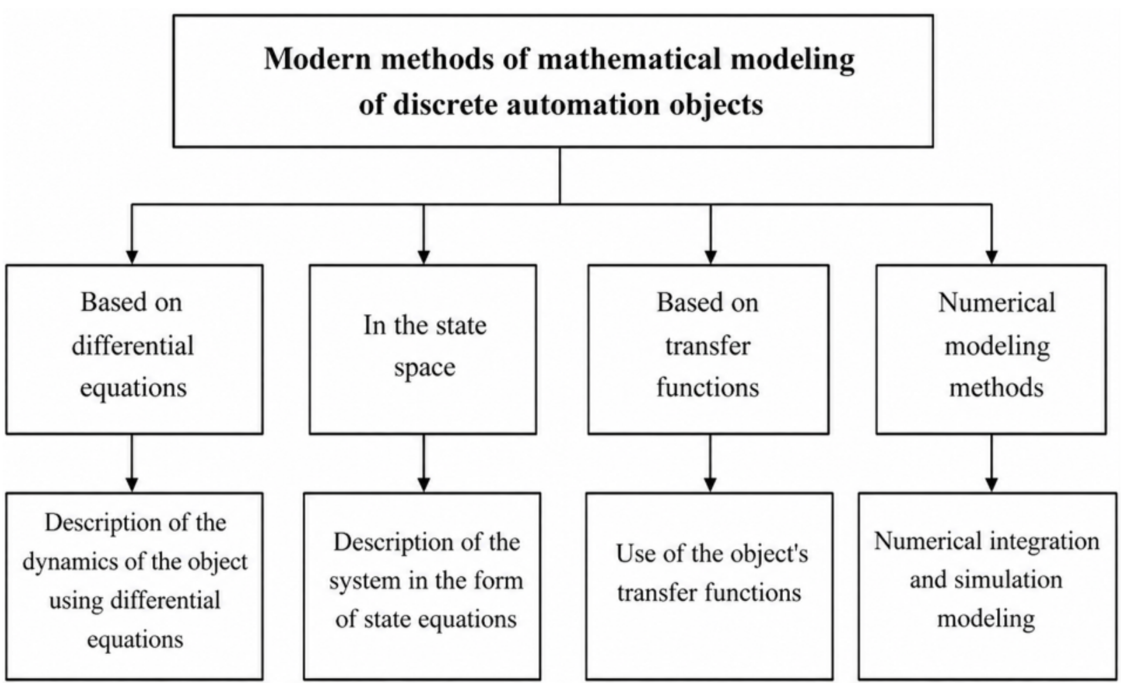

Fig. 1. Classification of Methods for Mathematical Modeling of Discrete Automation Objects

When mathematically describing dynamic automation systems, a system of ordinary differential equations is most commonly used to establish the relationship between state variables, input control signals, and output variables. The mathematical description of a dynamic system can be expressed as:

$$\frac{dx(t)}{dt} = f(x(t), u(t), t), \qquad (1)$$

where $x(t)$ – vector of an object's state variables;

$u(t)$ – vector of control inputs;

$f(x(t), u(t), t)$ – a nonlinear function that describes the dynamics of a system.

Presenting a mathematical model in this form is universal and allows for the description of a broad class of technical systems, regardless of their physical nature. For linear automation systems, it is advisable to express the mathematical model in state space, which significantly simplifies the analysis of multidimensional systems and is used in the synthesis of automatic control algorithms.

For linear systems, the model in state space takes the form:

$$\dot{x} = Ax + Bu, \qquad (2)$$

where A – system state matrix;

B – control matrix;

x – state vector;

u – control signal vector.

State-space models enable the analysis of a system's controllability, observability, and stability, and are widely used in the development of digital automatic control systems, robotic systems, and cyber-physical systems.

Use of the Laplace Transform and Numerical Modeling Methods. One of the most common methods for analyzing linear dynamic systems is the use of the Laplace transform, which allows one to transform differential equations into algebraic relationships. This approach significantly simplifies the study of the dynamic characteristics of automation systems, the analysis of their stability, the determination of time characteristics, and the construction of transfer functions. The use of operational calculus is particularly effective in the analysis of linear systems, where it is necessary to determine the system's response to standard input signals and to evaluate the parameters of transient processes.

The Laplace transform of a function of time is defined by the following equation:

$$F(s) = \int_0^\infty f(t)e^{st}dt\,, \tag{3}$$

where $f(t)$ – time function;

s – a complex variable in the operational domain.

After transitioning to the operational domain, mathematical models can be represented as transfer functions, which makes it possible to study the dynamic characteristics of systems without directly integrating differential equations. This approach is widely used in the analysis of electromechanical systems, robotic systems, and other automation systems.

Figure 2 shows the sequence of mathematical modeling using the Laplace transform.

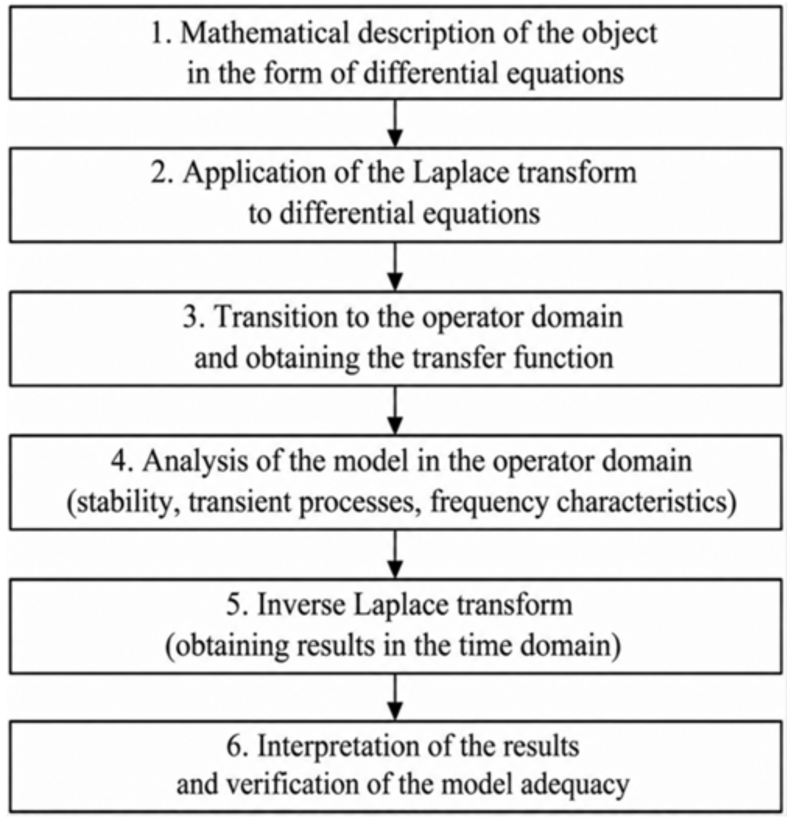

Fig. 2. Sequence of Mathematical Modeling Using the Laplace Transform

In many practical problems, mathematical models have a complex structure or contain nonlinearities, making it impossible to obtain an analytical solution. In such cases, numerical integration methods are used, which allow for obtaining an approximate solution to the mathematical model with the required accuracy. Among the most common are the Euler method, the Runge-Kutta methods, and their modifications, which ensure high modeling accuracy at an acceptable cost in terms of computational resources.

One of the simplest numerical methods is the Euler method, whose mathematical description is as follows:

$$x_{k+1} = x_k + hf(x_k t_k), \tag{4}$$

where h – integration step;

x_k – the value of the state variable at the k step;

$f(x_k t_k)$ – the value of the right-hand side of the differential equation.

Despite its simplicity of implementation, the Euler method has limited accuracy; therefore, when modeling complex automation systems, fourth-order Runge-Kutta methods are more commonly used, as they provide a significantly lower error in numerical integration and high stability of the computational process.

Table 1 presents a comparative analysis of mathematical modeling methods.

Table 1

Comparative Analysis of Mathematical Modeling Methods

Modeling Method	Advantages	Disadvantages	Main Area of Application
Differential Equations	high accuracy of the mathematical description	computational complexity for nonlinear models	analysis of dynamic systems
State Space	convenient analysis of multidimensional systems	requires a matrix representation	digital control systems
Laplace Transform	simplifies the analysis of linear systems	limited use for nonlinear models	analysis of transient processes
Numerical Methods	the ability to create complex models	the presence of a numerical error	computer modeling

A comparative analysis shows that none of the methods considered is universal. The choice of a specific approach is determined by the characteristics of the object under study, the requirements for modeling accuracy, and the available computational resources. In practice, the most effective approach is the integrated use of analytical and numerical methods, which ensures the development of adequate mathematical models for a broad class of discrete automation objects.

Existing software tools for mathematical modeling. Advances in computer technology and software have significantly changed approaches to the mathematical modeling of automation systems. Whereas previously most mathematical calculations were performed using analytical methods, today a significant portion of research is conducted using specialized software packages. This makes it possible to significantly reduce the time required for calculations, improve modeling accuracy, and study complex multidimensional systems for which an analytical solution is practically impossible.

The most widely used software packages in the field of automation are Scilab, Maxima, MATLAB, and Python. They support a wide range of tools for mathematical computations, numerical integration, building mathematical models, analyzing dynamic characteristics, and graphically visualizing results. Of particular interest are the open-source software packages Scilab and Maxima, which provide sufficient functionality for most scientific research and educational purposes.

One promising area is the use of Python software libraries (NumPy, SciPy, Matplotlib), which allow for the integration of mathematical modeling with modern technologies for data analysis, machine learning, and artificial intelligence. This opens up new opportunities for creating adaptive models, digital twins, and systems for predicting the technical condition of equipment.

Figure 3 shows a diagram illustrating the use of software tools for the mathematical modeling of discrete automation objects.

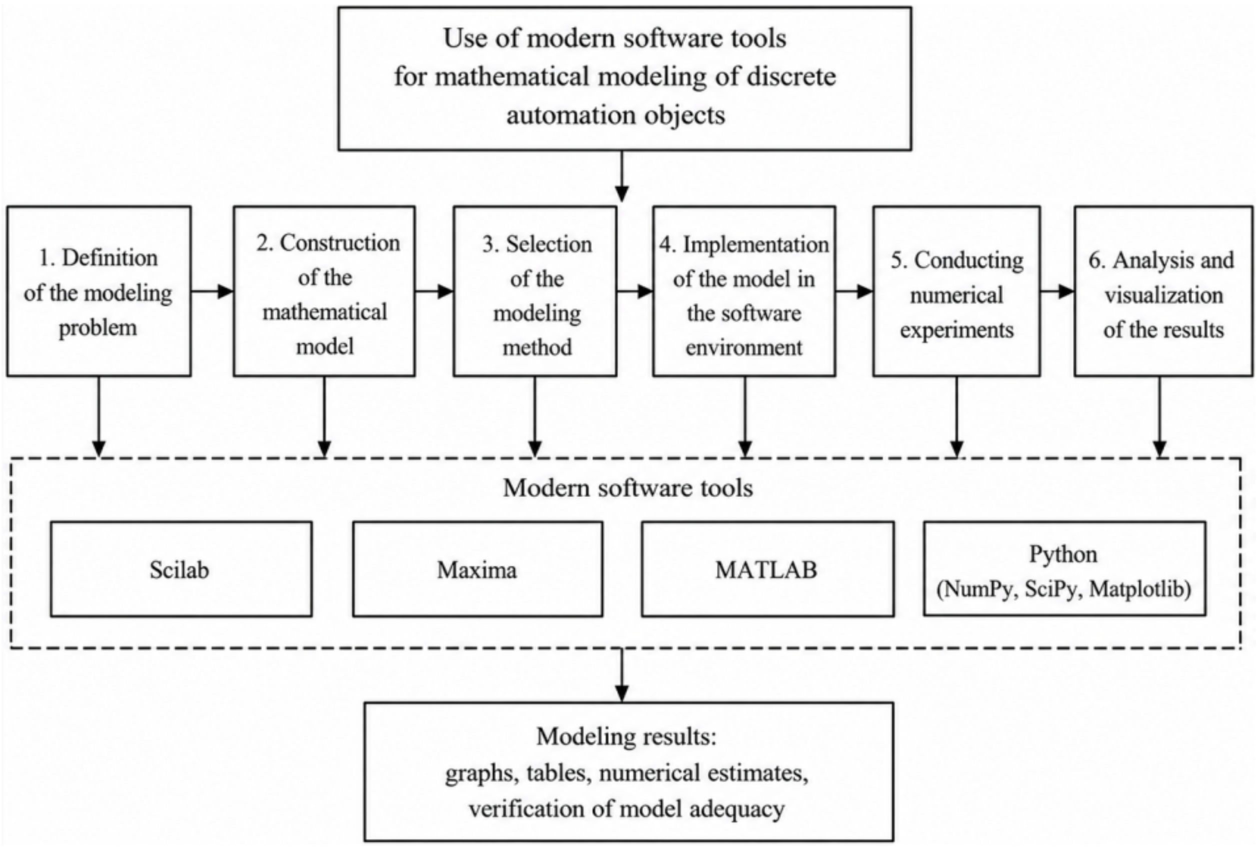

Fig. 3. Diagram Illustrating the Use of Software Tools for Mathematical Modeling of Discrete Automation Objects

In numerical simulation, the accuracy of calculations is often evaluated based on the magnitude of the absolute error:

$$\Delta = \left| x_e - x_m \right|, \tag{5}$$

where x_e – experimental value of the parameter;

x_m – a value obtained as a result of mathematical modeling.

Minimizing the absolute error is one of the main criteria for evaluating the adequacy of a mathematical model. The smaller the error value, the more accurately the model reflects the behavior of the actual automation object, and the more effectively it can be used in the design of automatic control systems.

Table 2 presents a comparison of mathematical modeling software tools.

Table 2

Comparison of Mathematical Modeling Software Tools

Software Tool	Open Source	Numerical Methods	Symbolic Computation	Main Area of Application
Scilab	yes	yes	partially	automation, control systems
Maxima	yes	limited	yes	symbolic modeling
MATLAB	no	yes	yes	scientific research, industry
Python (NumPy, SciPy)	yes	yes	partially	scientific computing, artificial intelligence

This comparison shows that each software tool has its own advantages and areas of application. For educational and scientific research, it is advisable to use the open-source software packages Scilab, Maxima, and Python, while MATLAB remains one of the most powerful tools for industrial tasks and complex mathematical modeling.

Conclusions. An analysis of methods for mathematical modeling of discrete automation objects was conducted, and their role in the study and design of automatic control systems was determined. The paper examines the specifics of using mathematical models based on differential equations, state spaces, the Laplace transform, and numerical integration methods. A comparative analysis of modeling methods and the software tools used to implement them was performed, and their advantages, limitations, and areas of practical application were identified. It is shown that the most effective approach is the comprehensive use of analytical and numerical methods in combination with modern software packages for mathematical modeling. This approach ensures greater accuracy of mathematical models, reduces the time required to construct them, and lays the foundation for the development of automation systems, robotic systems, and cyber-physical systems.

References

1. Boutalis Y., Theodoridis D., Kottas T., Christodoulou M.A. System Identification and Adaptive Control: Theory and Applications of the Neurofuzzy and Fuzzy Cognitive Network Models. ChamHeidelber-New York-Dordrecht-London: Springer, 2014. 315 p.

2. Heij C., Ran A., van Schagen F. Introduction to Mathematical Systems Theory Linear Systems, Identification and Control. BaselBoston-Berlin: Birkh¨auser Verlag, 2007. 174 p.

3. Ljung L. System Identification. Theory for the User. Upper Saddie River: Prentice Hall PTR, 1999. 628 p.

4. Graupe D. Identification of Systems. Malabar: Krieger Pub Co, 1976. 276 p.

5. System Identification, Environmental Modelling, and Control System Design / L. Wang, H. Garnier (Eds.). London-Dordrecht-Heidelberg-New York: Springer, 2012. 673 p.

6. Eykhoff P. System identification: Parameter and State Estimation. London: John Wiley & Sons, 1974. 555 p.

7. Speedy C. B., Brown R. F., Goodwin G. C. Control Theory: Identification and Optimal Control. Edinburgh: Oliver & Boyd, 1970. 293 p.