EPJ Web of Conferences **318**, 04005 (2025) https://doi.org/10.1051/epjconf/202531804005
AAPM-III 2025

Differential equations and their applications for mathematical modeling of systems and processes

Sergey Masaev*, *Irina* Ilina, *Sergey* Alimasov, and *Alexander* Korolev

Reshetnev Siberian State University of Science and Technology, 31, Krasnoyarsky rabochy prospect, Krasnoyarsk, 660037, Russian Federation

Abstract. This article provides a comprehensive comparison of various mathematical modeling software programs. It discusses the features, capabilities, and limitations of popular programs such as Wolfram, Maple, MatLab and MathCad. The article also evaluates the ease of use, performance, and flexibility of the software in different modeling scenarios. The aim is to help researchers and professionals make informed decisions when selecting a mathematical modeling tool for their specific needs. Additionally, the article offers recommendations for the best software based on different criteria and applications.

1 Introduction

A mathematical model is an abstract representation of a real system that uses mathematical formulas and methods to describe and predict its properties, behavior and interactions. Mathematical models help to understand and explain complex phenomena, conduct experiments, and predict results.

Mathematical modeling of systems is one of the key tools in science and engineering. It allows one to research the behavior of complex systems such as physical, biological, economic and social systems under various conditions. The purpose of mathematical modeling is to obtain quantitative and qualitative characteristics of the system, as well as predict its behavior in the future. Systems modeling tasks involve creating mathematical models that describe the behavior of complex systems over time. These models are used to analyze, predict and control systems in various fields, including physics, chemistry, biology, economics, engineering and others [1-2]. The use of mathematical modeling programs is most effective together with the mathematical formalization of the problem, at least in general terms.

2 Methods

The control quality criterion is a mathematical expression that allows you to quantify how successfully the requirements for the management method are met. The optimal control

* Corresponding author: faberi@list.ru

method is considered to be one in which the value of the control criterion reaches a minimum (in some cases, a maximum).

When solving control problems, it is necessary to take into account that the movement of the system is limited by various conditions. There are two main types of constraints that influence the choice of optimal control methods. Constraints of the first type are determined by the fundamental laws of nature that describe the dynamics of the system. These constraints are usually represented in the form of algebraic, differential, or difference equations called constraint equations. The second type of constraint is associated with limited resources or physical constraints that can affect the variables that describe the state of the system. Such constraints are usually expressed as a system of algebraic equations or inequalities [3].

The mathematical formulation of the control problem includes:

- Formulating management goals through control quality criteria;
- Determination of restrictions of the first type, represented by a system of differential or difference equations that limit the possible trajectories of the system's movement;
- Definition of restrictions of the second type, represented by a system of algebraic equations or inequalities expressing restrictions on resources or other parameters of the system;
- Selecting a control method that satisfies all constraints and minimizes (maximizes) the control quality criterion (optimal control method) [4].

The research of phenomena and systems, in the article, is considered in general with a simplified formalization of the mathematical model, using the variational method as an example, since it is the most common.

In scientific research, phenomena have a set of observable data, which $\{x_1, x_2, x_3, \dots, x_n\}$ is characterized as a number or vector with dimension n. those $x \in R^n$. These

$$f: x \to y \tag{1}$$

are the rules obtained in general form:

$$y = f(x) \tag{2}$$

This is the object of research through functionality, in optimization problems of finding the maximum or minimum of a function

$$F: x \to y. \tag{3}$$

In optimization problems, a definite integral

$$F(x) = \int_{t_0}^{t_1} f(x, \dot{x}, t)dt \tag{4}$$

on continuous functions is usually specified $C^1[t_0, t_1]$.

In general, problems of the dynamics of the phenomenon under research are studied through derivatives

$$f' = \frac{df}{dx} = \frac{f'dx}{dx} \tag{5}$$

$$df = f(x + h) - f(x) \tag{6}$$

where h is f

The change in functionality can be expressed similarly

$$F(x + h) - F(x) = \int_{t_0}^{t_1} f(x + h, \dot{x} + \dot{h}, t) - f(x, \dot{x}, t)dt \tag{7}$$

If it is linear, then the phenomenon is characterized by simple dependencies that are simply calculated. It is difficult to calculate integrals in complex, inexplicable phenomena. In complex cases, they try to solve the Taylor problem, since the representation of the process being studied as a power series simplifies the calculations themselves with a given error.

$$f(x, y) = f(a, b) + \frac{1}{1!}(f_x'(a, b)(x - a) + f_y'(a, b)(y - b)) + o(\rho) \tag{8}$$

where $o(\rho)$ - distance between two points a, b.

For $\int_{t_0}^{t_1} f(x + h, \dot{x} + \dot{h}, t)$

$$f(x + h, \dot{x} + \dot{h}, t) = f(x, \dot{x}, t) + f_x'h + f_{\dot{x}}'\dot{h} + o(h) \tag{9}$$

$$F(x+h)-F(x)=\int_{t_0}^{t_1}(f_x'h+f_{\dot{x}}'\dot{h})\,dt+o(h) \tag{10}$$

For h =0 we get

$$dF=\int_{t_0}^{t_1}f_x'h+f_{\dot{x}}'\dot{h}\,dt=\int_{t_0}^{t_1}f_x'h+\int_{(t_0)}^{h(t_1)}f_{\dot{x}}'\,dh=\int_{t_0}^{t_1}f_x'\,h+f_{\dot{x}}'h\Big|_{h(t_0)}^{h(t_1)}-\int_{h(t_0)}^{h(t_1)}hdf_{\dot{x}}' \tag{11}$$

or

$$dF=\int_{t_0}^{t_1}f_x'h+f_{\dot{x}}'\dot{h}\,dt=\int_{t_0}^{t_1}f_x'h+\int_{t_0}^{t_1}f_{\dot{x}}'\,dh=\int_{t_0}^{t_1}f_x'\,h+f_{\dot{x}}'h\Big|_{h(t_0)}^{h(t_1)}-\int_{t_0}^{t_1}hdf_{\dot{x}}'. \tag{12}$$

If $h(t_0)=h(t_1)$, $f_{\dot{x}}'=\frac{d}{dt}$ so

$$\int_{t_0}^{t_1}h\left(f_x'-\frac{d}{dt}f_{\dot{x}}'\right)dt \tag{13}$$

To use functional (4) to characterize the observed phenomenon, the equality is necessary

$$f_x'-\frac{d}{dt}f_{\dot{x}}'=0 \tag{14}$$

It is now possible to solve the optimization problem by minimizing $F(x)\to min$.

Next, you can analyze different options for the state of the phenomenon being studied through its variations $F'(x)$ depending on different rules in the function f.

The resulting power series (8) can be used for the task of extrapolation (forecast) of future function values.

You can also extend the Taylor series to the second variation, so you can obtain sufficient conditions for the minimum of the functional $F(x)$.

3 Results

Having a general mathematical formalization of the phenomenon under research and having determined the values of the functions, you can begin to research them in specialized programs.

The class of problems solved by mathematical systems modeling programs is broad and varied, and these tools play an important role in the analysis, prediction and control of various systems in various application areas of Wolfram, Maple, MatLab and MathCad, etc. They have broad capabilities for working with various types of models, including linear and nonlinear systems, statistical models, multidimensional systems and others.

It is important to note that all of these programs have wide functionality and can be used to model many types of systems. However, in cases where special specialization or functionality is required, the use of specialized software tools or programming languages may be necessary.

A comparison of these software solutions is shown Table 1.

Table 1. Comparison of programs for mathematical modeling.

Functionality	Wolfram	Maple	MatLab	MathCad
Symbolic computation	Yes	Yes	Limited support	Yes
Analytical Modeling	Yes	Yes	Limited support	No
Numerical modeling	Yes	Yes	Yes	Yes
Graphical representation of results	Yes	Yes	Yes	Yes
Solving differential equations	Yes	Yes	Yes	Yes
System Modeling	Yes	Yes	Yes	Limited support
Large set of libraries and packages	Yes	Yes	Yes	No
Interactive interface	Yes	Yes	Yes	Yes

Programming language support	Yes	Yes	Yes	No
Support for symbolic expressions	Yes	Yes	Limited support	Yes
Built-in support for developing flowcharts for system modeling	Yes	No	Yes	No
Built-in Functions and Operators	Yes	Yes	Yes	Yes
Support for working with matrices	Yes	Yes	Yes	Yes
Function optimization	Yes	Yes	Yes	No
Built-in tools for system stability and convergence analysis	Yes	No	Yes	No
Total	15 «Yes»	13 «Yes», 2 «No»	12 «Yes», 3 «Limited support»	8 «Yes», 6 «No», 1 «Limited support»

Table 1 gives a general idea of the functionality of each program [7-10]. However, it is worth noting that each program has its own characteristics and is intended for different applications.

The class of problems solved by programs for mathematical modeling of systems includes the following types of problems:

1. Forecasting the behavior of a system is the task of predicting the future behavior of a system based on its mathematical model. Mathematical modeling programs allow you to evaluate how a system will change over time and what factors or parameters may influence its further development.
2. Optimization of system parameters is the task of finding optimal values of system parameters in order to achieve specified goals or optimal results. Mathematical modeling programs allow you to analyze the influence of various parameters on the behavior of the system and optimize their values to achieve certain criteria.
3. System stability analysis is the task of determining the stability of a system and its behavior under various conditions. Mathematical modeling programs can perform numerical analysis of the stability of a system, examining its ability to maintain its state or return to it after deviations.
4. System management is the task of developing strategies to manage systems to achieve desired goals. Mathematical modeling programs can help optimize the control signal, such as setting regulators or determining the optimal timing to turn them on, so that the system operates most efficiently.
5. Interaction research is the task of researching the interactions between different components or subsystems in a system. Mathematical modeling programs can help evaluate the impact of one part of a system on another and determine how changes in one element can affect the entire system [7].

Common functionality of these programs includes:

1. Ability to work with various types of mathematical equations and differential equations, including ordinary differential equations (ODEs) and partial differential equations (PDEs).
2. Graphical presentation of simulation results, including the construction of graphs of functions, vector fields, phase portraits and three-dimensional models.
3. The ability to analyze the stability of systems, including finding equilibrium points, calculating eigenvalues and Jacobian vectors, analyzing stability and bifurcations.

4. Support for numerical methods for solving differential equations, such as Euler's method, Runge-Kutta method or finite difference method.
5. Ability to work with symbolic expressions and symbolic mathematical calculations, allowing an analytical approach to modeling and analysis of systems [11].

However, in addition to these capabilities, these programs may have some limitations regarding system modeling. Below are some functionalities that may not be available in these mathematical system modeling programs:

1. Lack of support for convenient specialized libraries for system modeling. Some other software environments, such as Simulink (part of the MatLab package), have advanced libraries containing many models for simulating systems such as electrical circuits, mechanical systems, fluid dynamics systems, etc.
2. Limited support for symbolic calculations for systems. Maple, for example, can perform analytical calculations, but support for symbolic work with differential equations and systems may be limited.
3. Limited support for numerical integration. Although Wolfram, Maple, MatLab, and MathCad provide tools for numerically solving differential equations, complex systems or systems with special conditions may require the use of specialized techniques that may not be available in these programs.
4. Lack of specific functions for modeling certain types of systems. In some cases, you may need to create your own functions or extensions for Wolfram, Maple, MatLab, or MathCad to model certain types of systems, such as distributed or nonlinear systems.

When choosing a program for mathematical modeling of systems, it is recommended to take into account your own requirements and preferences, as well as familiarize yourself with the additional capabilities and tools offered by each of them.

All programs have similar functionality for mathematical modeling of systems. However, Wolfram and Maple have a greater focus on symbolic computing and symbolic modeling, while MatLab and MathCad provide extensive computational capabilities and tools for numerical analysis [10-12].

Based on the results of comparative Table 1, Wolfram has more advantages and is therefore the most convenient software tool for mathematical modeling.

4 Discussion

This research is of great relevance and special importance for various fields of science and technology. Here are a few reasons why learning such programs is important:

1. Simplify Math: Symbolic mathematics allows you to work with symbols and variables instead of numbers, making it easier to solve complex math problems. Symbolic mathematics programs allow you to perform analytical calculations, carry out algebraic manipulations, solve equations, find derivatives, and much more. This helps reduce the time and effort required to perform such calculations manually.
2. Support for scientific research: Mathematics is the fundamental basis for many scientific fields such as physics, engineering, economics, artificial intelligence and others. Using symbolic mathematics programs allows researchers and scientists to perform complex mathematical calculations, find analytical solutions, and analyze large amounts of data. Such programs help speed up and simplify the research process, and also increase the accuracy and reliability of the results obtained.
3. Development of engineering solutions: In modern engineering practice, it is increasingly necessary to carry out complex mathematical modeling and analysis of systems. Symbolic mathematics programs enable engineers to perform analytical analysis and optimization of systems, solve equations and systems of equations, and

perform symbolic algebra to create and optimize algorithms. This helps reduce the time it takes to develop and test engineering solutions and increase their efficiency.

4. Training and Education: Using symbolic mathematics programs in the classroom can greatly facilitate the understanding of mathematical concepts and develop skills in applying mathematical methods to solve problems. Such programs allow you to visualize mathematical concepts, perform numerical and symbolic calculations, and teach students algorithmic thinking and the application of symbolic mathematics in practical situations.

Overall, the research of symbolic mathematics programs is of great relevance and can bring significant benefits in areas such as scientific research, engineering, and education. These programs allow you to simplify and speed up mathematical calculations, increase the accuracy of results, and develop skills in applying mathematical methods to solve complex problems.

5 Conclusion

As a result of the research, we can conclude that the use of software in solving problems of mathematical modeling of systems is an effective approach. Such tools allow you to automate the modeling process, simplify calculations and data processing, and also increase the accuracy and reliability of the results.

Software tools for mathematical modeling of systems offer a wide range of tools and functions that allow you to create and analyze various models, solve equations and systems of equations, conduct numerical experiments and simulations. They have a graphical interface that simplifies working with models and allows you to visualize the results.

One of the main advantages of using software is the ability to quickly and flexibly change the conditions and parameters of the simulated system. This allows you to explore different scenarios and behavior of the system, as well as take into account the influence of various factors. In addition, software tools have high computing power, which makes it possible to solve complex equations and systems, select optimal model parameters, and analyze large volumes of data. This is especially important when working with large and complex system models, where high accuracy and reliability of the results are required.

Thus, the use of software in solving problems of mathematical modeling of systems is an effective and appropriate approach. They allow you to increase the efficiency and accuracy of modeling, speed up the process of finding solutions and analyzing results.

References

1. N.V. Golubeva, *Mathematical Modeling of Systems and Processes* (Lan, St. Petersburg 2013).
2. M. Dorrer, A. Dorrer, *Analysis of the University's Supporting Business Processes Using Process Mining Methods Digital Transformation and New Challenges* in: Lecture Notes in Information Systems and Organisation **45**, ed E Zaramenskikh and A Fedorova (2021). https://doi.org/10.1007/978-3-030-71397-3_4
3. A.A. Samarsky, A.P. Mikhailov, *Mathematical Modeling: Ideas. Methods. Examples* (Moscow, Fizmatlit, 2002).
4. A.B. Kisabo, N.C. Uchenna, F.A. Adebimpe, American Journal of Mathematical and Computer Modelling **2(4),** 117-131 (2017).
5. B. Barnes, G.R. Fulford, *Mathematical Modelling with Case Studies: A Differential Equation Approach Using Maple* (CRC Press, 2003).

6. D. Kurasov, J. Phys.: Conf. Ser. **1691,** 012123 (2020).
7. A.A. Czajkowski, K. Sidorkiewicz, Problems of Applied Sciences **3,** 015-032 (2015).
8. V.R. Kristalinskii, S.N. Chernyi, International Journal of Open Information Technologies **7(2),** 42-48 (2019).
9. P. Mokhasi, J. Adduci, D. Kapadia, CODEE Journal **9(1),** 1-20 (2012).
10. M.M. Aripov, F. Tashpulatov, Research in Mathematical Education **9,** 269-73 (2005).
11. D.K. Chaturvedi, *Modeling and Simulation of Systems Using MATLAB and Simulink* (CRC Press, 2010).
12. J. G. S. Leon, *Mathematica Beyond Mathematics: The Wolfram Language in the Real World* (Chapman and Hall/CRC, 2017).