Algorithms for switching technical systems modeling

Olga N. Masina[1][0000-0002-0934-7217], Alexey A. Petrov[1][0000-0002-3011-9278],
Olga V. Druzhinina[2][0000-0002-9242-9730], Sergey V. Chernomordov[1][0000-0001-6773-8737] and
Daniil Y. Openkin[1][0000-0003-1519-1615]

[1] Bunin Yelets State University, 28, Kommunarov st., Yelets, 399770, Russian Federation,
[2] Federal Research Center "Computer Science and Control" of Russian Academy of Science,
44, building 2, Vavilov st., Moscow, 119333, Russian Federation,
olga121@inbox.ru

Abstract. The paper is devoted to the design of algorithms for switching technical systems modeling. These algorithms are based on the application of numerical methods for solving ordinary differential equations, methods of control theory, and methods of global parametric optimization. An algorithm for switching using a PID controller according to a mismatch signal is developed. An algorithm for global parametric optimization based on artificial neural networks with training is proposed. Model examples are considered. For the development of the corresponding software the Python language with a set of mathematical libraries is used. The obtained results can be used in solving various applied problems, in particular, problems of transport dynamics, problems of aircraft control, and problems of energy systems control.

Keywords: neural network modeling, switching technical systems, PID controller, machine learning, numerical optimization methods, Python.

1   Introduction

The switching system can be represented as a multi-mode dynamic system with a mode switching law that determines the activity intervals of each mode [1]. Such systems find numerous applications in applied problems of mechanical systems control, processes in switching power converters [2-3], in controlling the movement of unmanned aerial vehicles, automated transport systems, robotic systems, and in other fields [4-5]. The actual problems in this direction are the analysis of these technical systems models.

Within the framework of the direction related to the study of switching systems, the problems of constructing models and synthesizing controls are of significant interest [6]. Subsequent modeling is associated with the development of appropriate algorithms and with software implementation in high-level languages.

In [7-8], a number of problems of analysis and synthesis of two-dimensional and three-dimensional models of controlled switching systems are considered. These problems are solved on the basis of numerical optimization various methods using neural network modeling. The most important problems of neural network modeling are the problems of developing the effective methods of machine learning of neural networks [9-10]. The algorithms of global parametric optimization used in the problems of constructing optimal trajectories and in machine learning problems are considered in [7-8; 11–13] and in other papers.

The purpose of this paper is to develop algorithmic software for controlled switching systems modeling based on the use of intelligent technologies and numerical optimization methods. Mathematical models of technical systems with switching are considered. The models are described by means of ordinary differential equations considering the given initial conditions and multivalued right-hand sides. A switching algorithm based on the PID controller is proposed, as well as a neural network optimization algorithm. The analysis of model examples based on the results of computational experiments is carried out.

2   Models and methods

The mathematical model of the technical system under study is given by an ordinary differential equation of the form

$$m\dot{x} = QT + mG + D(x, \dot{x}, \ddot{x}, t). \tag{1}$$

In (1) the following notations are accepted: $x \in R^n$ is the phase vector, t is the time, $D(x, \dot{x}, \ddot{x}, t)$ is the perturbation in the system, G is the vector-column of the potential gravitational field, m is the mass. Through $T = (t^0, t^1, ... t^{n-1})^*$ we denote the column vector corresponding to the time. The matrix Q in (1) is a matrix of coefficients the search for which is carried out by means of solving the initial boundary value problem for the following equation:

$$m\ddot{x} = QT + mG. \tag{2}$$

We note that model (2) is a special case of the model (1) and is characterized by the absence of a perturbation vector. The switches in the model (1) are determined by the program control algorithm. We develop program control algorithms considering the need to reduce the number of switches.

The generalized model (1) allows us to form trajectories that can satisfy a set of phase constraints, considering a given quality criterion. For a model of an aircraft, we can set a quality criterion at which the time or energy costs will be minimal. In this case, the phase restrictions depend on the need to be in the specified air corridor and change the destination during the motion. In particular, one of the model examples is a model of the aircraft dynamics with the condition of minimum fuel consumption.

Further we consider the optimal control problem for the model (1). A control quality criterion has the following form:

$$\int_0^{t_1} \|U(t)\| \, dt \to \min. \tag{3}$$

In (3) the following notations are used: U is the control vector, t_1 is the specified time to reach the phase subspace.

We use methods of numerical optimization [14], methods of control theory (the method of constructing PID controllers [15] and the method of sliding modes [16]). In addition, global parametric optimization methods (including methods that simulate biological processes: neural network methods, bee swarm methods, ant colonies, etc.) can be used for searching of optimal trajectories of the model (1) [17].

In this paper we solve the control problem by synthesizing regulators based on artificial neural networks (ANNs). The synthesis of regulators with the use of ANNs makes it possible to form models of weakly formalized systems [9]. It should be noted the effectiveness of ANNs in the problems of finding optimal trajectories for dynamic models of switching systems. In particular, it is possible to change the parameters for the nonlinear approximation of multidimensional functions using the ANNs.

3   Results

We consider the issues of constructing a feedback control algorithm using a PID controller. The use of PID controllers reduces the time for debugging the system and allows to set the necessary parameters. Under the condition that there are no large noises in the control loop with feedback, the design of the PID controller becomes a simple, but quite effective way of control, and for the implementation of such control, a simplified algorithm for tuning the coefficients is sufficient [18].

The scheme of the generalized switching algorithm using the PID controller is shown in Figure 1. At the stage of this algorithm activation it is necessary to set up such a trigger threshold that leads to switching. Namely, instead of a continuous signal we fixate a specific value (0 or 1) which is used to control the switching of the system.
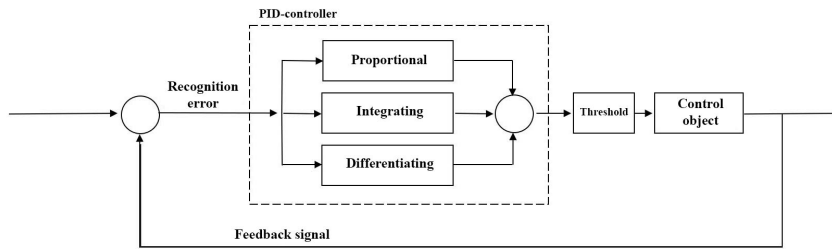

Fig. 1. Scheme of the generalized switching algorithm using the PID controller.

We configure the PID controller using a neural network optimization method. As neural networks for configuring the PID controller, multi-layer networks of direct signal propagation with one hidden layer and with a linear activation function are

most suitable. Note that when tuning the coefficients of the PID controller, there is an analogy with the selection of the weight coefficients of a neural network.

To solve the problem of constructing a switching algorithm, a method is proposed that is based on switching the matrices of the control parameters Q taking into account the feedback [7].

Further we present the algorithm developed for the model (1) "Configuring the PID controller" (Algorithm 1), which consists of the following steps.

— *Step* 1. To initialize the PID controller with random coefficients.
— *Step* 2. To perform the switching algorithm and to find the trajectory.
— *Step* 3. To calculate the value of the quality criterion.
— *Step* 4. If the stop condition is carried out, the algorithm is finished.
— *Step* 5. To optimize the coefficients *P, I, D* and the threshold value λ.
— *Step* 6. To proceed to step 2.

The configured PID controller is then used to find the optimal switching moments in the controlled system (2). In this algorithm, we use optimization methods to obtain the PID controller coefficients. In this case, the principle of applying optimization methods is that the coefficients will be selected taking into account the minimum number of switches.
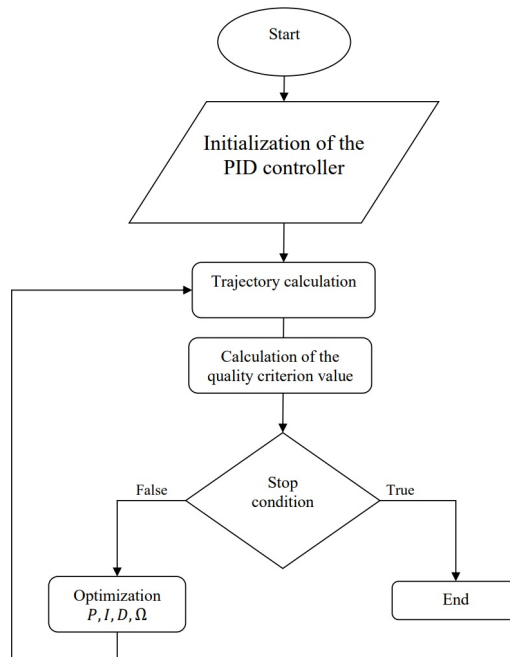
The flow chart of Algorithm 1 is shown in Figure 2.

Fig. 2. Flow chart of the algorithm "Configuring the PID controller".

Algorithm 2 includes Algorithm 1. We present Algorithm 2 "Switching generation based on the PID controller" for the two-dimensional case of the model (1), which consists of the following steps.

— *Step* 1. To configure the PID controller (Algorithm 1).
— *Step* 2. To perform one iteration of the numerical integration algorithm.
— *Step* 3. To check the stop condition. If the stop condition is carried out, the algorithm is finished.
— *Step* 4. To calculate the switching generator outputs. If the activation function is on the output is $x > \Omega$, then the switch is performed, otherwise go to step 2.
— *Step* 5. To calculate new values of matrix Q.
— *Step* 6. To proceed to step 2.

The flow chart of the algorithm "Switching generation based on the PID controller" is shown in Figure 3.
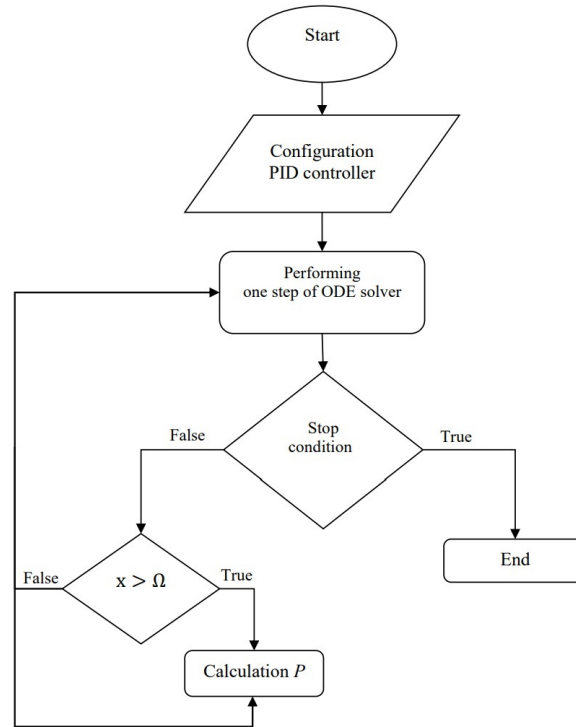
Fig. 3. Flow chart of the algorithm "Generation of switches based on the PID controller".

As a software tool for prototyping software we use the high-level language Python with the libraries NumPy, SciPy, SymPy, and Matplotlib for scientific computing, data processing, and visualization [19].

A prototype of software for interpreting the output value of the PID controller has been developed. The corresponding implementation code is shown in Figure 4.

```
import matplotlib.pyplot as plt
import matplotlib as mpl
mpl.rcParams['font.family'] = 'fantasy'
mpl.rcParams['font.fantasy'] = 'Arial'
from numpy import e,arange
def hp(t,Kp,Ti,Kd,Td):
        z=Kp+Kp*t/Ti+Kp*Kd*e**(-t/Td)
        return z
x=arange(0,10,0.02)
y1=[hp(t,2,2,0,0.5) for t in x]
y2=[hp(t,2,2,4,0.5) for t in x]
plt.plot(x, y2, linewidth=2,)
plt.legend(loc='best')
plt.grid(True)
plt.show()
```

Fig. 4. A program code fragment for the realization of PID controller.

The graphical interpretation for the example implementation of Algorithm 1 is shown in Figure 5, where $\mu(t)$ is the PID controller output, $\varepsilon(t)$ is the mismatch signal.
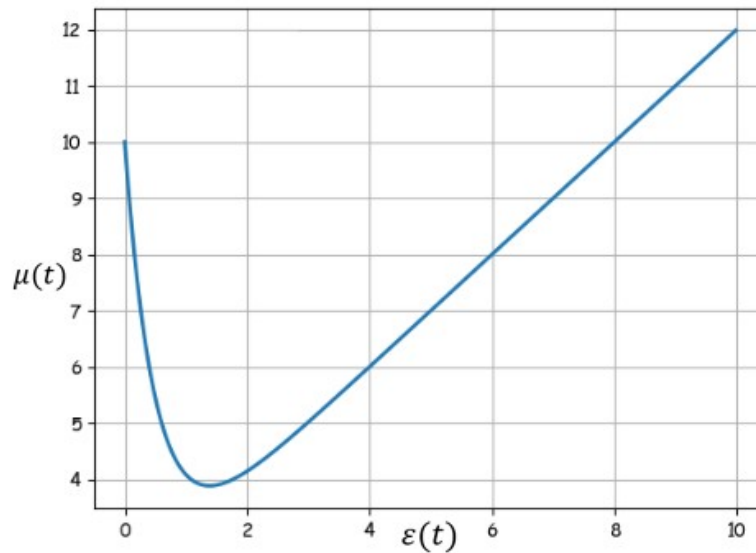

Fig. 5. Graphical interpretation of the PID controller output value.

Before implementing the PID controller the trajectories of the dynamic system of a search are being performed. As an example, we give the result of the trajectory search in the two-dimensional case when moving under the influence of gravity without taking into account the action of other forces. Figure 6 shows a program code fragment for searching of trajectories before further optimization.

```
model_example = switched_model_2d()
dynamic_system = ode(model_example, 0, [0, 5, 0, 50],
25, max_step = 0.1)
                 X = []

for point in dynamic_system:
    if model_example.timer[-1] > 15:
        model_example.P = np.array([[0,0],
                                    [0,0]])
        model_example.D = np.vstack([-1, 10])


    X.append(point)

solution = np.array(X)
plt.plot(solution[:,0], solution[:,2])
plt.grid()
plt.show()
```

Fig. 6. A fragment of the program code for finding a trajectory when moving under the influence of gravity.

The program code shown in Figure 6 can be used for further expansion and implementation of algorithm 1.

An important problem is to develop aggregated algorithms that combine switching algorithms and algorithms for finding optimal trajectories. Such an aggregated algorithm is a new neural network optimization algorithm, the essence of which is as follows. In the space of the optimized function, k neural network automata with a given topology are randomly initialized. A neural network automata is an artificial neural network with certain state parameters (energy and position). The purpose of functioning of a neural network automaton is to find the function optimum. Neural network automata form a closed system, the evolution of which proceeds according to a combined heuristic algorithm [20].

For the two-dimensional case of the model (1), a global parametric optimization algorithm is developed based on a combination of Hooke–Jeeves methods [21], artificial neural networks, and the swarm optimization method.

Further for the two-dimensional case of the model (1), we present the developed Algorithm 3 "Search for the function optimum by the aid of neural network automata swarm".

— *Step* 1. To initialize a k neural network automata ($k \geq 2$) in n-dimensional Euclidean space.

— *Step* 2. To calculate the outputs of neural network automata.
— *Step* 3. To change the position of neural network automata in accordance with the output vectors.
— *Step* 4. To check the stop condition. If the stop condition is carried out, the algorithm terminates. Otherwise, go to step 2.

We note that the proposed Algorithm 3 belongs to algorithms inspired by nature, and is a zero-order search algorithm. Further we present the developed Algorithm 4 "Training of neural network automata".

— *Step* 1. To initialize a network of k neural network automata ($k \geq 2$) in n-dimensional Euclidean space.
— *Step* 2. To perform steps 2-4 of Algorithm 3.
— *Step* 3. To find the value of the loss function.
— *Step* 4. To optimize the weight coefficients of neural network automata.

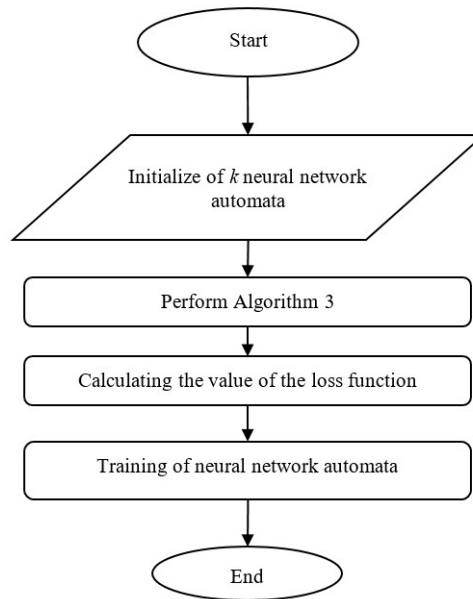The flow chart of Algorithm 4 is shown in Figure 7.

Fig. 7. Flow chart of the algorithm «Training of neural network automata».

The software based on the developed algorithms can be used to expand the optimization module of the SSMC software package [22].

The algorithm using neural network automata is a universal optimization algorithm and can be used in a number of applied problems of parametric optimization.

4 Discussion

For the developed flow chart (Figure 2), an example of the implementation of Algorithm 1 is given. Figure 3 shows the flow chart of Algorithm 2, which includes Algorithm 1. The program code shown in Figure 6 is a prototype of software for modeling switchable systems based on a PID controller. The considered examples demonstrate the effectiveness of the developed algorithms (see Figures 5). The program code shown in Figure 6 requires further extensions and modifications for various cases, in particular, for the case of increasing the system dimension and for the case of the influence of indeterminate perturbations with the corresponding complication of the model.

The implementation of Algorithm 3 and the performing of a series of computational experiments to find the optimal trajectories of switching systems are the prospects for the development of this paper. The importance of the results obtained in the development of optimization algorithms (see Algorithm 3) is that these results can be applied in a number of different subject areas related to the mathematical modeling of controlled systems.

To implement the flowchart of Algorithm 4 (Figure 7), it is possible to use reinforcement learning based on evolutionary algorithms to find the function optimum.

When studying and verifying switching high-dimensional models, along with algorithms 1-4 and algorithms for the numerical solution of multidimensional differential equations, it is advisable to use Data Mining methods [23] for the purpose of preliminary preparation of model parameters arrays.

5 Conclusion

In this paper we study an approach to computer research and development of algorithms and software for technical switching systems.

A switching algorithm based on a PID controller on the basis of a mismatch signal is proposed. The algorithm of global parametric optimization on the basis of artificial neural networks with training is developed. These algorithms allow to analyze the influence of various parameters on the qualitative characteristics of the switching technical system functioning process.

The presented results can be used in solving various optimization problems, in solving problems of mathematical control theory, as well as in designing and improving switching systems taking into account uncertainties. In addition, the results obtained can be used in machine learning problems and transport route optimization problems.

References

1. Liberzon, D.: Switching in systems and control. Birkhäuser, Basel (2003).

2. Ho, C.Y.F., Ling, B.W.K., Liu, Y.Q., Tam, P.K.S., Teo, K.L.: Optimal PWM control of switched-capacitor DC-DC power converters via model transformation and enhancing control techniques. IEEE Transactions on Circuits and Systems I: Regular Papers, 55, 1382–1391 (2008).

3. Loxton, R., Teo, K.L., Rehbock, V., Ling, W.K.: Optimal switching instants for a switched-capacitor DC/DC power converter. Automatica, 45(4), 973–980 (2009).

4. He, W., Chen, Y., Yin, Z.: Adaptive neural network control of an uncertain robot with full-state constraints. IEEE Transactions on Cybernetics, 46(3), 620–629 (2016).

5. He, W., Dong, Y., Sun, C.: Adaptive neural impedance control of a robotic manipulator with input saturation. IEEE Transactions on Cybernetics, 46(3), 334–344 (2016).

6. Zhao, X., Kao, Y., Niu, B., Wu, T.: Control synthesis of switched systems. Studies in systems, decision and control. Springer International Publishing (2016).

7. Druzhinina, O.V., Masina, O.N., Petrov, A.A.: Models for the control of technical systems motion taking into account optimality conditions. Evtushenko, Yu.G., Khachay, M.Yu., Khamisov, O.V., Kochetov, Yu.A., Malkova, V.U., Posypkin, M.A. (eds.) Proceedings of the OPTIMA-2017 Conference, CEUR Workshop Proceedings, 1987, 386–391.

8. Druzhinina, O.V., Masina, O.N., Petrov, A.A.: The synthesis of the switching systems optimal parameters search algorithms. Communications in Computer and Information Science, 974, 306–320 (2019).

9. Haykin, S.: Neural networks: a comprehensive foundation. Prentice Hall, Englewood Cliffs, NJ (1999).

10. Rashid, T.A., Abbas, D.K., Turel, Y.K.: A multi hidden recurrent neural network with a modified grey wolf optimizer. PLoS ONE, 14(3), 1–23 (2019).

11. Plotnikova N.P., Fedosin S.A., Teslya V.V.: Gravitation search training algorithm for asynchronous distributed multilayer perceptron model. Lecture Notes in Electrical Engineering, 312, 417–423 (2015).

12. Sun, S., Cao, Z., Zhu, H., Zhao, J.: A survey of optimization methods from a machine learning perspective. IEEE Transactions on Cybernetics, 50(8), 3668–3681 (2020).

13. Kim, K.S.; Choi, Y.S.: HyAdamC: A new Adam-based hybrid optimization algorithm for convolution neural networks. Sensors, 21(4054) (2021).

14. Nocedal, J., Wright, S.J.: Numerical optimization. Springer, New York (2006).

15. Kiam, H.A., Chong, G., Yun, L.: PID control system analysis, design, and technology. IEEE Transactions on Control Systems Technology, 13(4), 559–576 (2005).

16. Shtessel, Y., Edwards, C., Fridman, L., Levant, A.: Sliding mode control and observation. Series: Control Engineering. Birkhäuser, Basel (2014).

17. Karpenko, A.P.: Modern algorithms of search optimization. Algorithms inspired by nature. BMSTU, Moscow (2014).

18. Heinänen, E.: A Method for automatic tuning of PID controller following Luus-Jaakola optimization (Master's Thesis ed.). Tampere University of Technology, Tampere (2018).

19. Harris, C.R. et al: Array programming with NumPy. Nature, 585(7825), 357–362 (2020).

20. Aggarwal, C.: Neural Networks and Deep Learning. Springer International Publishing (2019).

21. Hooke, R., Jeeves, T.A.: Direct search solution of numerical and statistical problems. Journal of the ACM, 8(2), 212–229 (1961).

22. Druzhinina, O.V., Masina, O.N., Petrov, A.A., Lisovsky, E.V., Lyudagovskaya, M.A.: Neural network optimization algorithms for controlled switching systems. Advances in Intelligent Systems and Computing, 1225, 470–483 (2020).

23. Hastie, T., Tibshirani. R., Friedman J.H.: The elements of statistical learning: data mining, inference, and prediction. Springer, New York (2001).