Neural network implementation of controllers for multi-channel objects synthesized by polynomial method

A Voevoda¹* and V I Shipagin¹

¹Novosibirsk State Technical University (NSTU), Prospekt K. Marksa, 20, Novosibirsk, 630073, Russia
E-mail: *shipagin@mail.ru

Abstract. The implementation of neural network multichannel controllers synthesized by polynomial matrix decomposition is analysed. Objects and controllers are assumed to be linear; these allow them to be described by matrix transfer functions. The transfer function of the object is converted to the right polynomial matrix inter-simple decomposition. The transfer function of the controller is sought in the form of a left polynomial matrix of inter-simple decomposition that allows leading the characteristic matrix to the form of a linear matrix polynomial equation with two matrix indeterminates. This equation is solved by leading to a matrix equation with numeric matrix indeterminates. Then the controller equation is converted to a discrete equation. The discrete sampling step is chosen small enough to allow the systems with continuous and discrete controllers have sufficiently close transient processes. The discrete controller is converted to a structure including delay elements, adder units and amplification coefficients. Then this structure is presented in the form of a set of neurons. The operation of the algorithm is illustrated by the example of the synthesis of an unstable inverted pendulum control, which includes two PID controllers. Possible increases in neuro controller performance are demonstrated. Two PID controllers are combined into one neural network in order to further optimization.

1. Introduction
The modern development of technical systems, and, therefore, their complication, sets us the task to synthesize new, more complex controllers, including multi-channel ones. These features lead to the use of multi-channel controllers, but the calculation of their parameters is a difficult task [1, 2, 3]. One of the solutions of this situation can be considered the use of a linearized model of the object; however, this approach may not always provide the necessary level of performance of the controller on the real object.

Thus, the relevance of the research was dictated by the absence of a formalized approach to account additional features of the mathematical description of the object (namely, non-linear parameters) when designing controllers.

One of the solutions to synthesize the controllers can be considered the use of a neural network approach. The use of multilayer neural networks to synthesize the controllers has recently gained wide popularity [4, 5, 6]. The issue to choose their structure is solved by selecting from a standard set of architectures. One of the main problems in application of these methods is to determinate the sufficient complexity of neural network architecture [7, 8, 9, 10]. Redundant architecture contains redundant neural connections and leads to increased requirements for computing power of the hardware. The
number of weight coefficients increases, and therefore the problem of their search arises. Insufficient complexity does not provide the required system performance.

This article presents an approach that allows combining the advantages of classical and neural network approaches. Namely, the algorithm of the neural network controller architecture and its parameters is demonstrated using the transfer function of the discrete controller. This allows overcoming the excessive complexity of the neural network controller architecture and expanding its capabilities compared to a discrete controller.

The system of the “inverted pendulum on a trolley” is considered as an educational example in this article. Section 2 introduces the task and initializes the parameters. A modal method of controllers synthesis in a continuous form is considered (although for the operation of the algorithm for transferring to a neural network controller there is no fundamental difference by which algorithm the classical controller is formed) and then the transition to a discrete form. Using adder units, amplification and delay elements, the structural form of the discrete controller is formed. The transition from a discrete controller to a neural network is carried out in Section 3. The structure of the neural network controller is taken by information on the structure of the discrete controller, and then two methods of adjustment of neural network weight coefficients are considered. An approach is demonstrated that allows improving the operation of the neural network controller compared to the discrete one. Two discrete controllers are combined into one.

2. Materials and methods
This section describes an unstable object, the inverted pendulum. Continuous and discrete controllers are synthesized. Structural diagram is drawn up for the calculated controllers.

2.1. Statement of the task
This work deals with the task to stabilize the inverted pendulum on the trolley and bring the carriage to a given position. This task is a good educational example to test various methods for the synthesis of controllers. One of the features of this object is its multidimensional nature, where the number of inputs is not equal to the number of outputs. As part of this work, an object with one control signal and two adjustable values is considered.

Article [12] uses the following equations of motion of the “inverted pendulum”.

\[
\left(1 - \frac{ml}{M_1 L} \cos^2 \theta \right) \ddot{s} + \frac{ml}{M_1 L} g \sin \theta \cos \theta - \frac{ml}{M_1} \sin \theta \dot{\theta}^2 = \frac{1}{M_1} u ,
\]

(1)

\[
\left(1 - \frac{ml}{M_1 L} \cos^2 \theta \right) \ddot{\theta} - \frac{g}{L} \sin \theta + \frac{ml}{M_1 L} \sin \theta \cos \theta \dot{\theta}^2 = -\frac{1}{M_1 L} \cos \theta u .
\]

(2)

where \( L = (I + ml^2) / ml \), \( M_1 = M + m \); \( M \) – trolley mass; \( m \) – pendulum mass, \( s \) – the center of gravity coordinate of the trolley (horizontal axis – the distance from the center of gravity); \( \theta \) – deviation of the pendulum from the vertical; \( l \) – inertia moment of the pendulum relative to the center of gravity; \( l \) - the length of the pendulum rod; \( u \) - control effect.

For the synthesis of controllers by the polynomial method, articles [11, 12] consider the linearized model of the inverted pendulum on a trolley for the case when \( M \neq 0 \), \( l \neq 0 \). The linearization is carried out in the vicinity of the point \( \theta = 0 \) and \( \dot{\theta} = 0 \). This model of inverted one is described by a system of equations

\[
(1 - \frac{ml}{M_1 L}) \ddot{s} + \frac{ml}{M_1 L} g \theta = \frac{1}{M_1} u ,
\]

(3)

\[
(1 - \frac{ml}{M_1 L}) \ddot{\theta} - \frac{g}{L} \theta = -\frac{1}{M_1 L} u .
\]

(4)
The structural diagram of the linearized object model is as follows:

**Figure 1.** Structural diagram of the linearized model object [12].

where \( a = \frac{\alpha mlg}{M_1 L} \), \( b = \frac{\alpha g}{L} \). Considering the linearized model, we consider the parameters of the object for the case, when \( m = 70 \text{kg}, M = 30 \text{kg}, l = 1 \text{m}, g = 10 \text{m/sec}^2 \).

2.2 Synthesis of controllers

The authors solve the following problems in the article [13]: stabilization of the pendulum position in the vertical one (\( \theta = 0 \)); the trolley transfer to the specified position \( S = S_1 \). This assumes accessibility for measurement \( \theta, \ \dot{\theta}, S, \ \dot{S} \). Two PID controllers are used to solve the tasks. One of them is in feedback chain and provides vertical position of the pendulum. The second PID controller located in a straight chain is used to control the trolley position. The structural diagram of the system is given in Figure 2.

Note. When using two PID-controllers there is a problem of mutual reduction of operators of Laplace \( s \) which are in a denominator of transfer functions. It is not very good when they are combined and leads to unstable states of the system. The solution to this problem will be described in details in Subsection 3.3.

**Figure 2.** Structural diagram of the system “object + two PID controllers” [13].
The following values of the controller parameters for closed system roots were obtained \{-5, -5, -5, -1, -1\}:

\[
\alpha = 9774.7, \quad \beta = 5206.1, \quad \gamma = 454.5, \\
\delta = -378.8, \quad \varepsilon = -984.9, \quad \nu = 60.6.
\]

To obtain the transfer function of the controllers, it is necessary to make several structural transformations. Figure 2 shows that PID controller is implemented in the form of PI controller summation and its differentiating part. To obtain the general transfer function of the PID controller, we add a real-differentiating link to its differentiating part \( s/ (0.01s + 1) \). Then the transfer function of the PID controller by angle and by the trolley position takes the form:

\[
W_{\theta}(s) = \frac{(506.6s^2 + 5303.8s + 9774.7)}{(0.01s^2 + s)}, \\
W_{S}(s) = \frac{(-70.5s^2 - 988.6s - 378.8)}{(0.01s^2 + s)}.
\]

Turn to a discrete form with a sufficiently small sampling step of 0.01 seconds, so that the systems with continuous and discrete controllers have sufficiently close transient processes. To do this, use the built-in operator Matlab – “c2d”.

Transfer functions of discrete controllers take the form: \((az^2 + bz + c) / (dz^2 + ez + f)\). Where for the controller by angle \( a = 50660, \quad b = -97930, \quad c = 47330, \quad d = 1, \quad e = 1.4, \quad f = -0.4 \). For the controller by trolley position \( a' = -7045, \quad b' = 13460, \quad c' = -6421, \quad d' = 1, \quad e' = 1.4, \quad f' = -0.4 \).

Recall that the transfer function expresses the ratio of the output of the controller to its input \( W_{\theta}(z) = u_{\theta} / \Theta, \quad W_{S}(z) = u / S \). The discrete controller is converted to the structure including delay elements, adder units and amplification coefficients. Express the output of the controllers through the product of its input to the transfer function and reduce the equation by \( z^2 \):

\[
\begin{align*}
     u_{\theta} &= (1/z)(b \cdot \Theta - e \cdot u_{\theta} + (1/z)(c \cdot \Theta - f \cdot u_{\theta})) + a \cdot \Theta, \\
     u_{S} &= (1/z)(b' \cdot S - e' \cdot u_{S} + (1/z)(c' \cdot S - f' \cdot u_{S})) + a' \cdot S.
\end{align*}
\]

The above equations are represented in a structural form.

![Figure 3. Structural diagrams of the discrete controllers (from top to bottom): by pendulum angle, by trolley position.](image-url)
This structure is converted into neural network in the following section. In addition, another problem is solved that is the choice of weight coefficients.

### 3. Neural controller

This section shows the transition algorithm from the discrete controller to the neural one. We try to improve the operation of the neural network controller by increasing the complexity of the network architecture and adding non-linear activation functions. Unlike the controller obtained by the modal synthesis method, neural network implementation will improve control characteristics, such as re-regulation and transition time. We demonstrate the combination of two discrete controllers into one neural network.

#### 3.1. Synthesis of the neural controller

Figure 3 shows that we need the neural network with two inputs and one output. Based on the number of adder units and their location, the structure of the neuro controller consists of three layers, one neuron in each layer. All layers are input layers. Since there are no functional transformations of the signal, the linear function can be used as the activation one. Displacements are not required in neurons. The neural network, as well as the discrete controller, contains 5 tuning parameters - weight coefficients.

![Figure 3. Neuron controller](image)

**Figure 3.** Neural network structure with weight coefficients for controllers (from top to bottom): by pendulum angle, by trolley position.

The weights of the neural network controllers are chosen in two ways. The first one is the calculation by a given discrete controller (5 parameters). Where for the controller by angle $a = 50660$, $b = -97930$, $c = 47330$, $d = 1$, $e = 1.4$, $f = -0.4$. For the controller by trolley position $a' = -7045$, $b' = 13460$, $c' = -6421$, $d' = 1$, $e' = 1.4$, $f' = -0.4$. The obtained transient processes of the neuro controller completely coincide with the discrete one.

The second one is to adjust the scales using Matlab. The training sample was formed from the transient data of the discrete controller with a step of 0.01 seconds. Considering that the transient process lasts 10 seconds, the sample will consist of 1001 specimens.

Random values in the range [-1: 1] were used as initial conditions for weight coefficients. Levenberg – Marquardt algorithm was used to optimize the neural network. The obtained results were estimated by calculating the mean square error of the difference with the transient process of the discrete controller.
The mean square error between transient processes graphic charts by the neural network controllers synthesized by the first and second methods is significantly small and about $1 \times 10^{-20}$.

The transient processes for the neural network controllers are as follows:

![Figure 5. System response to step impact 5.](image)

Figure 5 shows that neural networks with weight coefficients, taken from the discrete controllers and tuned using Levenberg – Marquardt algorithm, show the same performance at the discrete level. The weight coefficient adjustment method for the structure (figure 4) does not improve the performance compared to the discrete controller.

To improve further the performance of the synthesized controller we complicate the structure of the neural network and change the educational sample. As initial values of weight coefficients we use ones obtained from the discrete controller.

### 3.2. Improvement of the neural controller performance

Try to improve the controller operation by the trolley position. To do this, enter the architecture of the neural network into the first and second layers of nonlinearity of the type ‘tansig’ (hyperbolic tangent). In addition, increase the complexity of the network architecture. In the first layer, increase the number of neurons to 2.

![Figure 6. Neural network structure with four neurons in the first layer.](image)

The training is performed at the discrete controller operation by the trolley position. As initial conditions for weight coefficients, we take random values within $[-1:+1]$. In addition, as it is shown in article [11], it is necessary to revise the training sample in the direction of data increase on the static
mode of the controller operation to increase the controller performance. To do this, increase the operating time of the controller to form the training sample from 10 to 30 seconds.

The results of the trained neuro controller is shown in figure 7.

Figure 7. The comparison of transient processes for discrete and neural network controllers for step exposure 5.

Figure 7 shows there is an improvement in the controller operation compared to the discrete one. Namely, the re-adjustment decreased from 60% to 10% and the departure time to static mode decreased from 6 to 2 seconds.

The reasons for this improvement include:

- training sample with predominance of information on static operation mode of the system;
- increased complexity of the neural network controller compared to architecture.

In addition, the operation of the improved neurocontroller by the trolley position effected the controller operation by pendulum angle. Its re-regulation decreased slightly.

We combine two controllers into one neural network to improve simultaneously by two controllers (by the trolley position and the pendulum angle).

3.3. Transition to one neural network controller

Figure 2 of this article shows two PID controllers were used to control the inverted pendulum system. The controller that provides the vertical position of the pendulum ($R_\Theta$) is located in the feedback chain of the system. The second controller used to control the position of the pendulum trolley ($R_S$) is located in the straight chain. Thus, the controller $R_S$ is located with respect to the controller ($R_\Theta$) in the outer chain. It means that the transfer function of the closed system takes the form:

\[
\frac{w_c(s)}{s} = \frac{w_{r_\Theta}(s)\cdot w_{r_S}(s)}{1 + w_{r_\Theta}(s)\cdot w_{r_S}(s)}
\]  

(5)

Where $w_c(s)$ - closed system transfer function; $w_{r_S}(s)$ - transfer function of the open part of the system (object - inverted pendulum); $w_{r_\Theta}(s)$ - transfer function of the controller by angle located in the feedback link; $w_{r_S}(s)$ - transfer function of the controller by the position of the pendulum trolley located in straight chain.

When calculating the transfer function of the closed system $w_c(s)$, taking into account that the transfer functions of the controllers of the form: $w_{r_\Theta}(s) = (\alpha + \beta s) / s + \gamma s$; $w_{r_S}(s) = (\delta + \epsilon s) / s - \nu s$, mutual reduction of Laplace operators $s$ occurs located in the denominator of transfer functions of the controllers. It is not very good when they are combined and leads to unstable states of the system.
To avoid this effect, it is necessary to take the integrating part of the PID controllers behind the adder unit, which is located before entering the object (figure 2). In addition, as in this work we use a really differentiating link of the form \( \frac{s}{0.05s + 1} \) instead of the differentiating link of the PID controllers, we also take it behind the adder unit.

**Figure 8.** Converted structural diagram of the system “object + two PID controllers”.

Convert the integrating and real-differentiating links to the discrete type with the discrete sampling step of 0.05sec.:

\[
0.05/(z-1), \quad (20z - 20)/(z - 0.4).
\]

Compare the nature of transient processes of continuous and discrete controllers to determine how the discrete controller works with the continuous object.

The discrete controller completes the task rather well in regulating the continuous object. When the discrete sampling step increases (greater than 0.05 seconds), the system is “loosened”. Thus we conclude that the selected discrete sampling step is sufficient.

As it is shown in the second section of this article, the obtained discrete controller is converted into the structure including delay elements, adder units and amplification coefficients.

**Figure 9.** Structural diagram of the generalized controller.

According to the obtained scheme, we build the neural network controller with three inputs and one output. The neural network consists of 5 layers. The input layers are the first, second, and last ones. In addition, it contains of two hidden layers and the last layer is the output layer. We take linear as the activation function, since functional transformations after adder units are not required. The neural network contains delay elements in the fourth layer and feedback links. Thus, the neural network structure of the generalized object looks as follows.
Input layers  Hidden layers  Output layer

Figure 10. Neural network structure of the generalized controller.

Figure 10 shows that the neural network architecture has feedbacks, that means we deal with a recurring neural network. There is a number of articles that describe the problem of adjusting the weight coefficients of recurring networks using the algorithms based on the method of back propagation through time [12, 13]. It consists of a suddenly “disappearing” or “exploding” gradient. It was confirmed when trying to adjust the weights using Matlab methods. In order to solve this problem, it is planned to apply the approach described in these articles in the following researches.

In the framework of this research, weight coefficients obtained from the structure of the generalized discrete controller were used (figure 9).

4. Discussion
To avoid the excessive complexity of the neural network, the algorithm is shown to calculate its architecture based on the knowledge of the transfer function of the discrete controller.

Two methods to choose weight coefficients of the neural network are tested: from the discrete controller and the adjustment by Levenberg – Marquardt method. The obtained neural network controllers show the performance at the discrete level.

It is found out that the neural network controller possesses the ability to increase the performance compared to the discrete one. It requires a slight increase in the complexity of the neural network architecture and the introduction of non-linear activation functions, as well as the adjustment of the training sample towards the increase in the data of the static mode of the controller operation. The property of training neural network controllers will be used in following works when transitioning to non-linear objects.

In addition, two discrete controllers were combined into one neural network. The obtained neural network architecture possesses feedbacks, i.e. presents the recurring network type. When trying to adjust it, the problem of “exploding” and “disappearing” gradients was obtained, described in details in works [14, 15]. It is planned to try to overcome this problem using the BPTT (back propagation through time) approach in the further study.

5. Conclusion
The advantages of the applied approach in comparison with the approach using the standard neural network architecture include: the problem solution in choosing the necessary network structure and reducing the number of adjustable weight coefficients, and therefore reducing the time and hardware resources necessary for the synthesis of such controllers. The required training sample volume can also be reduced.

We carry out some complications of the neural network architecture, that is, we increase the number of neurons and add activation of non-linear functions. To train the neural network, we take
amplification elements of the discrete controller as initial values of weight coefficients. The next step is to check the working efficiency of this structure and compare its performance with the classic controller.

The objective of the research is to formalize the procedure for obtaining a neuro controller according to the calculated classical one. The proposed method of neuro structure synthesis of controlled controllers makes it possible to expand possibilities to control the unstable object compared to the controller obtained by the method of polynomial matrix decomposition. The presented structure of the neural network is not redundant compared to typical ones, which means it does not require additional computing power and resources to configure it. It is proposed to solve the problem of selection of initial values of weight coefficients for adjustment of a neural network.

Unlike the classical controllers synthesized using polynomial matrix decomposition, the neural network approach allows expanding the applicability boundaries of the controllers (including non-linear models).

References
[1] Golnaraghi F, Kuo B C 2017 Automatic control systems, 10th ed., New York: McGraw-Hill, 1160 p
[2] Isidori A 2016 Lectures in Feedback Design for Multivariable Systems, Advanced Textbooks in Control and Signal Processing, London: Springer, 414 p
[3] Bobobekov K M, Troshina G V, Voevoda A A 2019 The parameters identification of the automatic control system with the controller, Journal of Physics: Conference series, 1210, Art. 012021
[4] Yang X C, Yung M H, Wang X 2018 Neural-network-designed pulse sequences for robust control of singlet-triplet qubits, Phys. Rev. A 2018, 97, 042324
[5] Ping Z 2013 Tracking problems of a spherical inverted pendulum via neural network enhanced design, Neurocomputing, 106, pp 137-147
[6] Nizami T K, Chakravarty A, Mahanta C 2017, Design and implementation of a neuro-adaptive backstepping controller for buck converter fed PMDC-motor, Control Eng. Pract., 58, pp 78-87
[7] Manuel L, Belen C, Antonio S 2019 Neural network architecture based on gradient boosting for IoT traffic prediction, Future Generation Computer Systems, 100, pp 656-673
[8] J. da Silva Adenilton J, R. de Oliveira Wilson R, B. Ludermir Teresa B 2016 Weightless neural net-work parameters and architecture selection in a quantum computer, Neurocomputing, 183, pp 13-22
[9] Zoph B and Le Q V 2016 Neural architecture search with reinforcement learning, arXiv pre-print arXiv:1611.01578
[10] Elsken T, Metzen J H, and Hutter F 2018 Neural architecture search: A survey, arXiv pre-print arXiv:1808.05377
[11] Shipagin V I 2019 Neural network implementation of a controller for a stable object, Collection of scientific papers of Novosibirsk state technical university, No. 3-4 (96), pp 53-63. DOI: 10.17212/2307-6879-2019-3-4-53-63
[12] Voevoda A A, Shoba E V 2012 About the inverted pendulum model, Collection of scientific papers of Novosibirsk state technical university, No. 1 (67), pp 3-14
[13] Voevoda A A, Shoba E V 2012 Control of the inverted pendulum, Collection of scientific papers of Novosibirsk state technical university, No. 2 (68), pp 3-14
[14] Razvan Pascanu, Tomas Mikolov, Yoshua Bengio, On the difficulty of training Recurrent Neural Networks, arXiv:1211.5063 [cs.LG]
[15] Arjovsky, M Shah A, Bengio Y, Unitary Evolution Recurrent Neural Net-works, arXiv:1511.06464