INFORMACION TECHNOLOGY. AUTOMATION

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ. АВТОМАТИЗАЦІЯ

UDC 681.142.2

V. Tigariev, PhD, Assoc. Prof.,
O. Lopakov,
V. Kosmachevskiy,
A. Koliada, PhD, Assoc. Prof.,
I. Lvov,
D. Dmytrenko
Odessa Polytechnic National University, Shevchenko Ave. 1, Odesa, Ukraine, 65044, e-mail: kedrodessa9@gmail.com

SYNTHESIS OF A NEURAL NETWORK REGULATOR AND EXPERIMENTAL STABILITY STUDY FOR AN ADAPTIVE QUADCOPTER CONTROL SYSTEM

В. Тігарєв, О. Лопаков, В. Космачевський, А. Коляда, І. Львов, Д. Дмитренко. **Синтез нейромережевого регулятора та експериментальне дослідження стійкості для адаптивної системи керування квадрокоптером.** При вирішенні низки складних маніпуляційних завдань доцільно брати до уваги нелінійну динаміку об'єкта управління. До таких завдань, зокрема, можна віднести керування великими маніпуляторами космічного базування, а також наземними маніпуляційними системами, що застосовуються у будівництві, у разі ліквідації наслідків аварій та катастроф. Для подібних маніпуляційних систем завдання управління ускладняються за рахунок того, що динаміки керованої конструкції дуже складна і, в більшості випадків, не може бути математично описана. У зв'язку з цим не завжди можуть бути застосовані методи, що ґрунтуються на вирішенні зворотного завдання динаміки. Застосування PID-контролерів, які широко використовують у більшості промислових додатків, також дозволяє взяти до уваги особливості динаміки руху таких систем. Також виникають проблеми із забезпеченням стійкості, у тому числі при дії зовнішніх факторів, які наперед не відомі. Новий напрямок у цій галузі пов'язані з застосуванням нейронних мереж, які можуть оцінити динаміку системи як реального часу. З іншого боку, застосування ковзних режимів у системах управління забезпечує незалежність процесу управління, як від зовнішніх впливів, і від параметричних збурень. Поєднання цих методів дозволяє створити систему, яка може усунути деякі недоліки кожного методу. У цій статті розроблено метод управління, що базується на адаптивному алгоритмі налаштування нейронної мережі. Пропонований метод дозволяє керувати системою без апріорної інформації про структуру та параметри динамічної моделі керованого об'єкта. Для визначення коефіцієнтів нейромережевого регулятора застосовуються адаптивні алгоритми, що дозволяють проводити його налаштування як нормального функціонування системи. За допомогою методу Ляпунова отримано умови стійкості такої системи керування. Ефективність запропонованого способу управління підтверджується результатами моделювання системи управління в середовищі MATLAB, а також експериментальними дослідженнями робототехнічних систем.

Ключові слова: БПЛА, рівняння динаміки, рівняння Ейлера-Лагранжа, ефект високочастотного тремтіння, апроксимація нейронної мережі, сигмоїдальна безперервна активаційна функція, функція Ляпунова, MIMO (Multi Input Multi Output), RBF (Radial basis function), SMC (Sliding mode control)

V. Tigariev, O. Lopakov, V. Kosmachevskiy, A. Koliada, I. Lvov, D. Dmytrenko. **Synthesis of a neural regulator and experimental stability study for an adaptive quadcopter control system.** When solving a number of complex manipulation tasks, it is advisable to take into account the nonlinear dynamics of the control object. Such tasks include, in particular, the control of large space-based manipulators, as well as ground-based manipulation systems used in construction and in the aftermath of accidents and disasters. For such manipulation systems, the control task is complicated by the fact that the dynamics of the controlled structure is very complex and, in most cases, cannot be mathematically described. In this regard, methods based on solving the inverse dynamics problem cannot always be applied. The use of PID controllers, which are widely used in most industrial applications, also allows us to take into account the peculiarities of the motion dynamics of such systems. There are also problems with ensuring stability, including under the influence of external factors that are not known in advance. A new direction in this area is related to the use of neural networks, which can estimate the dynamics of the system in real time. On the other hand, the use of sliding modes in control systems ensures the independence of the control process from both external influences and parametric disturbances. The combination of these methods allows to create a system that can eliminate some of the disadvantages of each method. In this article, we develop a control method based on an adaptive neural network tuning algorithm. The proposed method allows controlling the system without a priori information about the structure and parameters of the dynamic model of the controlled object. Adaptive algorithms are used to determine the coefficients of the neural network controller, which allow its adjustment as a normal functioning of the system. The stability conditions of such a control system are obtained using the Lyapunov method. The

DOI: 10.15276/opu.2.70.2024.12

effectiveness of the proposed control method is confirmed by the results of modeling the control system in MATLAB, as well as by experimental studies of robotic systems.

Keywords: UAV, dynamics equation, Euler-Lagrange equation, high-frequency jitter effect, neural network approximation, sigmoidal continuous activation function, Lyapunov function, MIMO (Multi Input Multi Output), RBF (Radial basis function), SMC (Sliding mode control)

Introduction

The theory of nonlinear control covers a wide class of systems that do not obey the superposition principle. Note that all real control systems are nonlinear. These systems are described by nonlinear differential or difference equations. The mathematical methods developed to handle them are more rigorous and much less general, often applied only to narrow categories of systems.

Backstepping control is a popular method of controlling nonlinear systems represented in so-called strict feedback form, such as in undervoltage systems. For example, some examples of tests for robotics such as Acrobot, Rotating Pendulum and Cart-Pole are defined in a strict feedback form.

Backstepping is a method developed around 1990 by Petrak, Kokotowicz and others to design stabilizing controls for a special class of nonlinear dynamical systems that have a recursive structure to their mathematical description. Because of this recursive structure, an engineer can start the design process in a known stable system and “undo” new controllers that gradually stabilize each external subsystem. The process ends when the final external control is reached. The advantage of this approach is that it can be applied recursively in terms of expanding the dimensionality of the system. The disadvantages are the difficulty in finding the Lyapunov function, and all state variables must be measurable. A nonlinear observer is also needed if not all state variables are measurable. It is also sensitive to parameter changes.

Another approach to controlling a nonlinear system is to first linearize the nonlinear system, and the linearized system is compensated using known methods of linear control theory.

The Lyapunov function can be used to synthesize nonlinear control systems. First, the Lyapunov function V must be found for a closed-loop system, and then a control law must be developed that makes the derivative $\dot{V}$ negative for the desired region of attraction (for all possible initial conditions, perturbations, and other uncertainties). Lyapunov stability theory is a standard tool and one of the most important tools for analyzing nonlinear systems. It can be extended relatively easily to non-autonomous systems and can provide a strategy for constructing stabilizing feedback controllers. In stability studies, the concept of positively defined function plays an important role as it serves as a Lyapunov function in applications. The disadvantages that can be mentioned are the difficulty in finding the Lyapunov function and the need for measurability of all state variables.

Analysis of recent publications

In [1], adaptive neural control is developed for a rehabilitation robot with unknown system dynamics in order to eliminate uncertainties in the system and improve robustness. Adaptive NN (Neural Network) is used to identify the unknown robot model and adapt the interactions between the robot and the patient. In [2], Neural Network based control for robot manipulators is proposed considering uncertain environmental constraints, perturbations and unknown robot dynamics. In [3], a robust neural network output feedback control scheme, which includes a neural network observer, is presented to control the motion of a manipulator robot. In [4], adaptive robust control strategies are presented for controlling coordinated motions of multiple mobile manipulators interacting with a rigid surface in the presence of uncertainties and perturbations. Radial basis function (RBF) network has also found applications for controlling dynamic systems [5] RBF network is a specialized neural network architecture. The adaptation of RBF network can improve the control performance of a system with uncertainty. The adaptation law is obtained using Lyapunov method, so that the stability of the whole system and the convergence of the adaptation of the network weight coefficients are guaranteed. In [6], adaptive control using RBF network is proposed to obtain the elements of inertia matrix, Coriolis matrix and gravity vector that determine the dynamics of manipulation robots. In [7], a modification of RBF-SMC neural network is proposed to eliminate bounce and control the SMA type actuator. The RBF network [7] also utilized adaptive adjustment of the sliding mode factor to eliminate the effects of dynamic uncertainties and ensure asymptotic convergence of the tracking error in the system. In [8], an RBF network for adaptive compensation of tracking error of a continuous-time nonlinear system is presented. The control system provides smooth switching between adaptive and sliding modes, integrating the advantages of robust and intelligent control. Meanwhile, SMC is utilized to compensate the uncertainties and ensure the stability of the nonlinear system. In [8–9], a pre-tuned neural network has been used to solve the sensorimotor coordination formation problems of a robot manipulator. The developed neural structure allows solving the problems of synthesis of coordinated motion control of multilink robot manipulators by combining in a single process both the search for elementary behavioral acts and the reproduction of previously mastered behavioral scenarios. In [9], kinematic control systems for multilink manipulators are considered. A method of synthesizing robot control systems using neural networks is proposed and quality criteria for forming a training sample of the neural network are selected. An algorithm implemented in a software package is developed and the error of training sample formation is established. Yu and Weng (2014) proposed a robust adaptive neural network based ISMC approach for controlling nonlinear interconnected systems with unknown uncertainty. They combined the adaptive neural approach with ISMC method to deal with unknown coupled uncertainties, system uncertainties and external disturbances. The numerical simulation results confirmed the effectiveness of the applied control. Fan and Yang (2016) proposed ISMC for a class of nonlinear systems with unknown nonlinear terms and perturbations using adaptive control. A neural network based observer was applied to estimate the unknown nonlinear terms and perturbations. Heuristic methods were applied to tune the controller. Gholami and Markazi (2012) proposed a new adaptive fuzzy sliding mode (AFSM- Adaptive Fussy Sliding Method) which can be used for a class of nonlinear MIMO type systems. The proposed method was applied to a real manipulator. Tong et al. (2011) proposed an observer-based adaptive fuzzy control method. They developed a nonlinear fuzzy observer to estimate the unmeasured state vector of the system. Zabihifar and Markazi (2013) proposed an adaptive fuzzy sliding mode controller for Autonomous Unmanned Underwater Vehicle. Doostmohammadi and Markazi (2010) proposed a new method for nonlinear systems in strict feedback form using adaptive fuzzy sliding mode approach. Budjedir et al. (2012) presented an adaptive sliding mode control based on two neural networks for quadrocopter stabilization They used the second neural network to eliminate the chattering phenomenon. Bouhali et al. (2011) proposed an adaptive neural control scheme based on a new observer as applied to quadrocopter control. This technique is implemented using two parallel artificial neural networks (ANN) for each subsystem of the quadrocopter. Andropov et al. (2016) presented an adaptive method for adjusting PID controller coefficients based on neural networks. Neural networks with three layers are used to design an adaptive stabilization system for a multirotor drone. Saikalis et al. (2002) presented an approach for adaptive control of a neural network using a novel adaptation algorithm that is based on the principle of adaptive interaction. This approach does not require the creation of a model using a neural network. Adaptation does not require the use of error back propagation method in the neural network of the feedforward network. This important property allows to directly provide adaptation of the control neural network. The inverse problem of manipulator dynamics as applied to the control problem of complex systems has been investigated in many works, including [10–12]. Inverse dynamics models of robot manipulators can be analytically derived from the Newton-Euler equation of motion. However, analytical derivation requires a priori knowledge of the moments of inertia, friction forces, gravity, and centripetal forces, which can be difficult to determine. They can also vary with wear and tear on the robot.

This fact raises the need for approaches that identify the dynamics model based only on current sensor information measurements. In addition, such methods must be able to operate in real time to ensure the adaptability of the system to dynamic changes that may occur during manipulation tasks [12].

Unfortunately, the solution of the multicopter control problem is complicated due to the essential nonlinearity and the action of external perturbations. The most common PID controllers, as well as linear-quadratic controllers, do not cope with this task well enough [13], [14]. There arises the necessity of operative adjustment of PID-coefficients of regulators in the process of operation, or complete readjustment in cases of changes in the parameters of the control object. It is mentioned in [14] that PI (PI) and PID (PID)-regulators realize the two most popular control methods, which have been used in several research works performed in the field of robot manipulator motion studies. The advantage of these methods lies in their simplicity. However, for motion planning of a system with nonlinear dynamic properties, the application of these methods can lead to inaccurate performance. Also, these controllers cannot always provide asymptotic stability in the task of tracking a given trajectory. To overcome these limitations, the use of neural network (NN) can be considered as a suitable method.

Problem statement

In general, the dynamic model of a quadrocopter in Euler-Lagrange form is written as [15]:

ISSN 2076-2429 (print)
ISSN 2223-3814 (online)

$$M(q)\ddot{q}+V_m(q,\dot{q})\dot{q}+G(q)+F(\dot{q})=U\ , \qquad (1)$$

where $M(q)$ is the inertia matrix;

$V_m(q,\dot{q})\dot{q}$ – matrix of Coriolis and centripetal forces;

$G(g)$ – gravity vector;

$F(\dot{q})$ – vector of aerodynamic drag forces;

U – vector of control forces.

Let us consider the mode of motion along the trajectory and orientation in space.

Let's define the current error as $e = q_d - q$, where $q_d = [x_d \ \ y_d \ \ z_d \ \ \varphi_d \ \ \theta_d \ \ \psi_d]^T$ – vector of desired coordinates, and $q = [x\ y\ z\ \varphi\ \theta\ \psi]^T$ – vector of current coordinates.

We set the sliding surface by transforming the error by a linear filter *S(t)*:

$$S = \dot{e} + \lambda e\ , \qquad (2)$$

where $\lambda > 0$ – is a diagonal matrix representing the slope of the slip surface. When the system moves along the sliding surface in the phase plane, the error S and its derivative $\dot{S}$ tend to zero.

The quadrocopter dynamics equation in terms of the transformed error S can be written as:

$$M\dot{S} = -V_m S + f(x) - U\ , \qquad (3)$$

where $f(x)$ – is an unknown function determined by the dynamics of the quadrocopter:

$$f(x) = M(q)(\ddot{q}_d + \lambda\dot{e}) + V_m(q,\dot{q})(\dot{q}_d + \lambda e) + F(\dot{q}) + G(q)\ . \qquad (4)$$

According to the universal approximation theorem, an artificial neural network with a single hidden layer can approximate any nonlinear, continuous, unknown function with any accuracy [16]. Based on this statement, let us introduce a two-layer neural network with a sigmoidal activation function of hidden layer neurons to approximate the function $f(x)$, which is described by the relation:

$$f(x) = W^T \sigma(V^T x) + \varepsilon\ , \qquad (5)$$

where W, V are matrices of unknown weighting factors, under the restriction imposed on the approximation error: $\|\varepsilon\| < \varepsilon_N$ (ε_N – non-negative constraint).

Let us denote the estimation of the function *f(x)* using the introduced neural network:

$$\hat{f}(x) = \hat{W}^T\ \sigma(\hat{V}^T x)\ , \qquad (6)$$

where $\hat{V}, \hat{M}$ – current values of the weighting multipliers, then, we can calculate the errors, $\tilde{V} = V - \hat{V}, \tilde{W} = W - \hat{W}$ with which the neural network is tuned.

Purpose of the research

The purpose of this research is to show that the developed control methods using adaptive neural networks can be used not only for controlling manipulators, but also for robotic systems of other types. In this paper, a solution to the problem of quadrocopter control using a neural network controller is proposed. A control method based on an adaptive neural network tuning algorithm is considered. The proposed method allows controlling the system without a priori information about the parameters of the dynamic model of the controlled object, which in this case is quite complex, as well as about external perturbing influences. To determine the optimal weighting factors of the neural network, as above, adaptive algorithms are used to ensure the tuning of the network in the process of its operation. The Lyapunov method is used to prove the stability of the control system. The proposed control method has shown good results in modeling the control system including neural network regulator and dynamic model of quadrocopter in MATLAB environment.

Currently, there is a steadily growing interest in the use of unmanned multirotor aerial vehicles (UAVs) designed for a wide range of tasks, mainly due to the simplicity of design and high payload capacity, compared to the classical versions of helicopters.

This paper presents a method for controlling a quadrocopter using a neural network controller, which does not assume knowledge of a priori information about the dynamics of the object and about external perturbations. Neural networks approximate the main control signal that allows the system to move near the sliding surface, as well as a corrective control signal that smooths out high-frequency jitter ("chatter").

Presentation of the basic material

Quadcopter dynamics

A quadrocopter is an unmanned aerial vehicle with four motors attached to the ends of a cross-shaped frame (Fig. 1). The opposing motors rotate in different directions to compensate for each other's changing momentum moments. Each motor has a vertical thrust force responsible for lifting and maneuvering the quadcopter.

Using Euler angles and Newton's second law, we can define the equation of dynamics as [17]:

$$\begin{aligned}
\ddot{\varphi} &= \frac{U_\varphi(t)}{I_x} + \frac{(I_y - I_z)}{I_x}\dot{\psi}\dot{\theta} - \frac{I_r\Omega_r(t)}{I_x}\dot{\theta} - \frac{k_{frx}}{I_x}\dot{\varphi}^2 + d_\varphi;\\
\ddot{\theta} &= \frac{U_\varphi(t)}{I_y} + \frac{(I_y - I_x)}{I_y}\dot{\psi}\dot{\varphi} - \frac{I_r\Omega_r(t)}{I_y}\dot{\varphi} - \frac{k_{fry}}{I_y}\dot{\theta}^2 + d_\theta;\\
\ddot{\psi} &= \frac{U_\psi(t)}{I_z} + \frac{(I_x - I_y)}{I_z}\dot{\varphi}\dot{\theta} - \frac{k_{frz}}{I_z}\dot{\psi}^2 + d_\psi;\\
\ddot{x} &= \frac{(C_\psi S_\theta C_\varphi + S_\varphi S_\varphi)U_1(t)}{m} - \frac{k_{ftx}}{m}\dot{x} + d_x;\\
\ddot{y} &= \frac{(S_\psi S_\theta C_\varphi + S_\varphi C_\psi)U_1(t)}{m} - \frac{k_{fty}}{m}\dot{y} + d_y;\\
\ddot{z} &= \frac{C_\theta C_\varphi U_1(t)}{m} - \frac{k_{ftz}}{m}\dot{z} - g + d_z,
\end{aligned} \tag{7}$$

where m – quadcopter weight;

I_r – sum of moments of inertia of rotors taken about z-axis;

I_x, I_y, I_z – moments of inertia with respect to the principal axes;

k_{ftx}, k_{fty}, k_{ftz} – aerodynamic drag coefficients;

k_{frx}, k_{fry}, k_{frz} – aerodynamic rotational drag coefficients;

φ, θ, ψ – roll, pitch and yaw angles;

x, y, z – center of mass coordinates;

$d_x, d_y, d_z, d_\varphi, d_\theta, d_\psi$ – external perturbations in coordinates (e.g., wind);

$\Omega_r = \sum_i^4 (-1)^{i+1}\omega_i$, ω_i – angular velocity of the i-th motor.

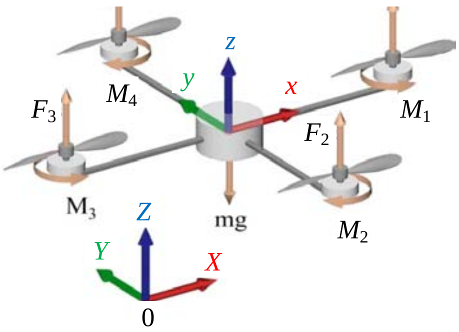

Fig. 1. Computational model of the quadrocopter

There are three types of quadcopter movement:

– lifting-descent is achieved by simultaneously increasing and decreasing the thrust force of the motors;

– pitch as a result of the difference in thrust of the front and rear motors, similarly roll;

– yaw is achieved by using the difference in speed of differently rotating motors, based on the theorem of conservation of kinetic momentum of a mechanical system.

The control signals with respect to the angular velocities of the motors that take into account the three possible types of motion are defined as follows:

$$\begin{bmatrix} U_1(t) \\ U_\varphi(t) \\ U_\theta(t) \\ U_\psi(t) \end{bmatrix} = \begin{bmatrix} k_p & k_p & k_p & k_p \\ 0 & -lk_p & 0 & lk_p \\ -lk_p & 0 & lk_p & 0 \\ k_d & -k_d & k_d & -k_d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}, \tag{8}$$

where k_p – thrust coefficient;

k_d – moment coefficient.

ISSN 2076-2429 (print)
ISSN 2223-3814 (online)

One of the main modes of quadrocopter control is trajectory motion. Considering the equations of dynamics of the system (7), we note that in the translational motion of the system, the control signals for moving the center of mass in the stationary coordinate system *OXYZ* are as follows [18]:

$$U_x(t) = (C_\psi S_\theta C_\varphi + S_\varphi S_\psi) U_1(t);$$
$$U_y(t) = (S_\psi S_\theta C_\varphi + S_\varphi C_\varphi) U_1(t); \quad (9)$$
$$U_z(t) = C_\theta C_\varphi U_1(t),$$

where $S_\varphi = \sin(\varphi), C_\varphi = \cos(\varphi), S_\theta = \sin(\theta), C_\theta = \cos(\theta), S_\psi = \sin(\psi), C_\psi = \cos(\psi).$

The desired trajectory of the transfer motion is determined by the parameters: φ, θ и $U_1(t)$.

Let's express the equations of the desired roll and pitch from formulas (9):

$$\varphi_d = \arcsin\left(\frac{U_x(t) S_{\psi d} - U_y(t) C_{\psi d}}{\sqrt{U_x^2(t) + U_y^2(t) + U_Z^2(t)}}\right);$$
$$\Theta_d = \arctan\left(\frac{U_x(t) C_{\psi d} - U_y(t) S_{\psi d}}{U_z(t)}\right), \quad (10)$$

where $\varphi_d, \Theta_d, \psi_d$ – desired motion parameters;

$S_{\psi d} = \sin(\psi_d), C_{\psi d} = \cos(\psi_d).$

Synthesis of neural network regulator

When the system moves along the sliding surface, the effect of high-frequency shaking occurs, which negatively affects the quality of following the desired motion trajectory. Let us introduce a second neural network to eliminate this effect – a corrective one. The structure of the neural network regulator consisting of the main and corrective neural networks is presented in Fig. 2.

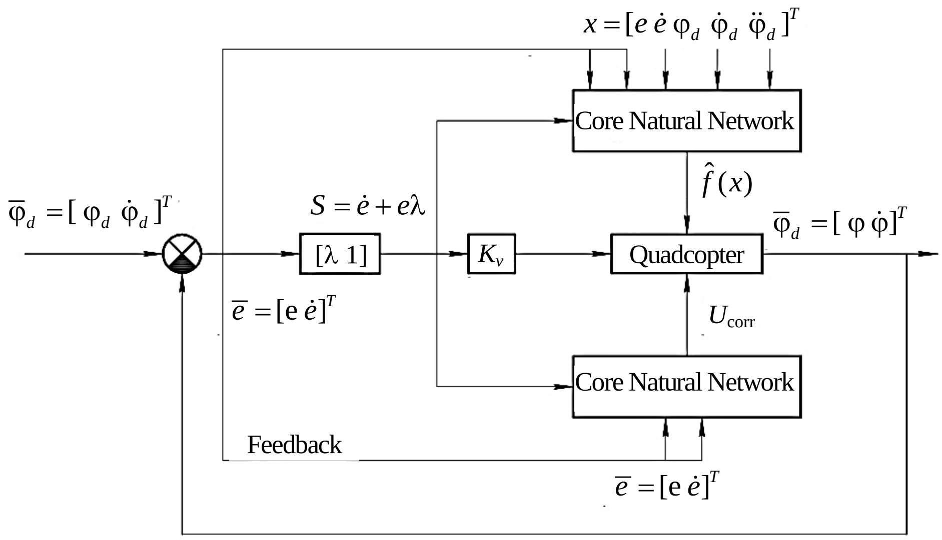

Fig. 2. Structure of neural network regulator

Let us represent the control signal in the form of two summands:

$$U = U^* + U_{corr}, \quad (11)$$

where U^* – is the main control signal for system motion in the vicinity of the sliding surface;

U_{corr} – correction signal.

Let's choose the main control signal as:

$$U^* = \hat{W}^T \sigma(\hat{V}^T x) + K_V S. \quad (12)$$

A corrective control signal is introduced to smooth the effect of high-frequency jitter near the sliding surface, and has at its core a continuous function *P*(.):

$$P(x) = \frac{1-e^{-2x}}{1+e^{-2x}}. \tag{13}$$

Then the ideal corrective control signal:

$$U^*_{\text{corr}} = B^* P(S^*), \tag{14}$$

where S^* – linear combination of e and $\dot{e}$.

A second neural network should be added to the circuit for approximation (Fig. 3).

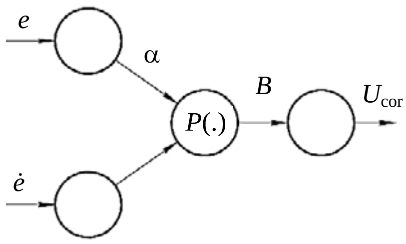

Fig. 3. Structure of the corrective neural network

The output of this network:

$$U^*_{corr} = BP(\alpha^T x^e) + \varepsilon_c, \tag{15}$$

where the weighting factors B and $\alpha = [\lambda_1 \quad \lambda_2]^T$ – ideal weights of the output and hidden layer of the network;

$x^e = [e \quad \dot{e}]^T$ – input vector;

ε_c – approximation and P(.) is the activation function defined in formula (13).

Since the weights B and α are unknown. We need to find their adjustment equations. Let's write down the estimation of the correcting control signal:

$$U^*_{\text{corr}} = \hat{B} P(\hat{\alpha}^T x^e). \tag{16}$$

In this case the estimation error of the correction signal:

$$\tilde{U}_{\text{corr}} = \tilde{B}\hat{P} + \hat{B}\hat{P}'\tilde{\alpha}^T x^e + w_g, \tag{17}$$

where $\hat{B}, \tilde{\alpha}$ – appraisal B, α accordingly, $\tilde{B} = B - \hat{B}, \tilde{\alpha} = \alpha - \hat{\alpha}$ estimation error,

w_g – approximation error:

$$w_g = \varepsilon_c + \tilde{B}\hat{P}\hat{\alpha}x^e + \tilde{B}O(\tilde{\alpha}x^e)^2. \tag{18}$$

Let's assume that: $\|w_g\| \le \overline{w}_g, |B| \le B_m, \|\alpha\| \le \alpha_m$, where $\alpha_m, B_m, \overline{w}_g$ unknown positive constants.

Let us take the tuning laws proposed in [19]:

$$\dot{\hat{B}} = F_B \hat{P}(\alpha^T x^e) S; \quad \dot{\hat{\alpha}} = F_\alpha (x^e S \hat{B} \widehat{P'} + k_\alpha |S| (\overline{\alpha} - \hat{\alpha})), \tag{19}$$

where $F_B = F_B^T > 0, F_\alpha = F_\alpha^T > 0, k_\alpha > 0$,
it's labeled:

$$\overline{\alpha} = \left[\frac{l}{|\varepsilon| + \varepsilon} \right]^T, l, \varepsilon > 0. \tag{20}$$

After substitution into (11) the control signal will take its final form:

$$U = \hat{W}^T \sigma(\hat{V}^T x) + K_V S + \hat{B} P(\hat{\alpha}^T x^e). \tag{21}$$

Using this control the equations of the system dynamics in terms of the transformed error are:

$$M\dot{S} = -(K_V + V_M)S + W^T \sigma(V^T x) - \hat{W}^T \sigma(\hat{V}^T x) + \hat{B}P(\hat{\alpha}^T x^e) + \varepsilon. \tag{22}$$

Let's add and subtract from the right part of equality (22) the expression $W^T \hat{\sigma} + \hat{W}^T \tilde{\sigma}$, where $\hat{\sigma} = \sigma(\hat{V}^T x), \quad \tilde{\sigma} = \sigma - \hat{\sigma}$:

$$M\dot{S} = -(K_V + V_M)S + \tilde{W}^T \hat{\sigma} + \hat{W}^T \tilde{\sigma} + \tilde{W}^T \tilde{\sigma} + \hat{B}\hat{P} + \varepsilon. \tag{23}$$

The key step is the Taylor series expansion of the sigmoidal continuous activation function. Restricting ourselves to the first two terms of the expansion, let us write:

$$M\dot{S} = -(K_V + V_M)S + \tilde{W}^T \hat{\sigma} + \hat{W}^T \hat{\sigma}' \tilde{V}^T x - \tilde{B}\hat{P} - \hat{B}\hat{P}'\tilde{\alpha}x^e + w_1, \tag{24}$$

where the approximation error:

ISSN 2076-2429 (print)
ISSN 2223-3814 (online)

$$w_1(t) = \tilde{W}^T \hat{\sigma}' \tilde{V}^T x + BP(\alpha^T x^e) + w_g + W^T O(\tilde{V}^T x)^2 + \varepsilon . \quad (25)$$

Let $w_1(t)$ in (24) be zero. Then it can be shown that the error *S(t)* tends to zero with time if we put:

$$\dot{\hat{W}} = F(\hat{\sigma} S^T - k\|S\|\hat{W}); \quad \dot{\hat{V}} = G(xS^T \hat{W}^T \hat{\sigma}' - k\|S\|\hat{V}), \quad (26)$$

for any *F, G, k*>0 (“e-modification technique” in [20–23]). In this case, the weight coefficients corresponding to the estimates remain bounded. Let us show that with the chosen algorithm for setting the weight coefficients, the motion of the system in the vicinity of the sliding surface will be stable. Let us define the Lyapunov function as:

$$L = \frac{1}{2} S^T M(q) S + \frac{1}{2}\mathrm{tr}\{\tilde{W}^T F^{-1} \tilde{W}\} + \frac{1}{2} tr\{\tilde{V}^T G^{-1} \tilde{V}\} + \frac{1}{2}\mathrm{tr}\, P\{\tilde{B}^T F_B^{-1} \tilde{B}\} + \frac{1}{2}\mathrm{tr}\{F_\alpha^{-1} \tilde{\alpha}^2\}. \quad (27)$$

Let’s differentiate:

$$\dot{L} = S^T M \dot{S} + \frac{1}{2} S^T \dot{M} S + \mathrm{tr}\{\tilde{W}^T F^{-1} \dot{\tilde{W}}\} + \mathrm{tr}\{\tilde{V}^T G^{-1} \dot{\tilde{V}}\} + \mathrm{tr}\{\tilde{B}^T F_B^{-1} \dot{\tilde{B}}\} + \mathrm{tr}\{F_\alpha^{-1} \tilde{\alpha}\, \dot{\tilde{\alpha}}\} . \quad (28)$$

Substituting into equation (24), at $w_1 = 0$:

$$\dot{L} = -S^T K_V S + \frac{1}{2} S^T (\dot{M} - 2V_m) S + \mathrm{tr}\{\tilde{W}^T (F^{-1} \dot{\tilde{W}} + \hat{\sigma} S^T)\} + \mathrm{tr}\{\tilde{V}^T (G^{-1} \dot{\tilde{V}} + xS^T \hat{W}^T \hat{\sigma}')\} +$$
$$+ \mathrm{tr}\{\tilde{B}^T (F_B^{-1} \dot{\tilde{B}} - S^T \hat{P})\} + \mathrm{tr}\{(F_\alpha^{-1} \tilde{\alpha}\, \dot{\tilde{\alpha}} - \tilde{\alpha} x^e S^T \hat{B} \hat{P}')\}. \quad (29)$$

The matrix $(\dot{M} - 2V_m)$ is cosymmetric, then the second summand in equation (29) is zero. Also, from $\hat{w} = w - \tilde{w}$ considering that *W* is constant, it follows that $\dot{\tilde{W}} = -\dot{\hat{W}}$ (similarly for *V, B, α*).

Then:

$$\dot{L} = -S^T K_V S \le 0 . \quad (30)$$

The first derivative of the Lyapunov function is nonpositive, which guarantees the stability of the system with feedback [24–28].

Experimental research

In the course of research, a quadrocopter model was created in MATLAB environment according to the equations of dynamics (7), as well as a controller for three angles (roll, pitch and yaw). The controller consists of a neural network for approximation of the main control signals input vector:

$$x = \begin{bmatrix} \varphi_d & \dot{\varphi}_d & \ddot{\varphi}_d & e & \dot{e} \end{bmatrix},$$

for each axis (roll, pitch and yaw), 7 neurons in the hidden layer), as well as three neural networks for approximation of corrective control signals.

External disturbances: $d_\varphi, d_\theta, d_\psi = \sin(t) + 3$; simulation time: *T*=20 s.

The system parameters adopted in the modeling are presented in Table 1.

Table 1

Parameter values in modeling

Parameter	Value	Parameter	Value
g, m/s^2	9.81	k_{ftz}, N·s/m	6e^{-4}
m, kg	0.5	$k_{fr\pi}$; $k_{fr\theta}$, kg·m^2/rad	5.5e^{-4}
l, m	0.25	$k_{fr\psi}$, kg·m^2/rad	6e$_{-4}$
k_p, N·s^2	3e^{-5}	F	100
K_d, N·m·s^2	3.5e^{-7}	G	40
I_r, kg·m^2	2.5e^{-5}	K	1
I_x, kg·m^2	3.5e^{-3}	F_B	10
I_y, kg·m^2	3.5e^{-3}	F_α	4
I_z, kg·m^2	8e^{-3}	k_α	1
k_{ftx}, k_{fty},N·s/m	5.5e^{-4}		

Results of the research

Some simulation results are given below. Figure 4 shows the motion of the quadrocopter in roll angle, which is almost identical to that given in the form of a periodic function. Figure 5 shows the self-tuning process of the neural network controller. It can be seen that the tuning time does not increase two seconds.

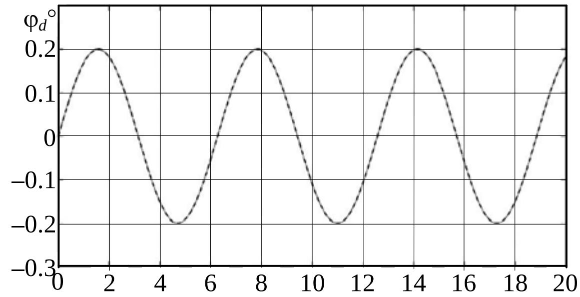

Fig. 4. Roll: dashed line – φ_d , solid line – φ

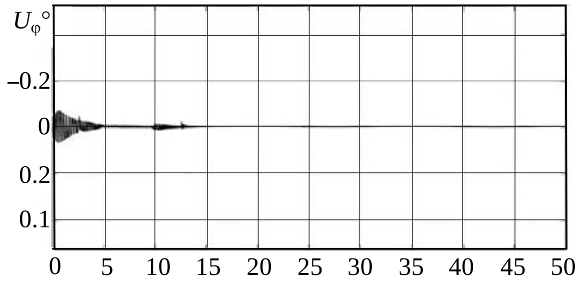

Fig. 5. Roll control signal – U_φ

Figure 6 shows a graph of changes in pitch, and Figure 8 shows a graph of changes in yaw. From the graphs of Figure 6 and Figure 8, it can be seen that the disturbance has significantly deviated the UAV angles from the stable position, but the controller quickly returns the system to a stable position. Thus, the simulation showed that the proposed controller can effectively bring the system to the equilibrium point. As can be seen from Figure 7 and Figure 9, in a very short period of time, the controller can adapt the parameters of a completely unfamiliar model and control the system. The pitch and yaw control signal is within the acceptable range.When the desired angle value changes abruptly, overshoot tends to zero. During an abrupt change of the desired angle value, there are significant control surges - potentially dangerous for the control system.

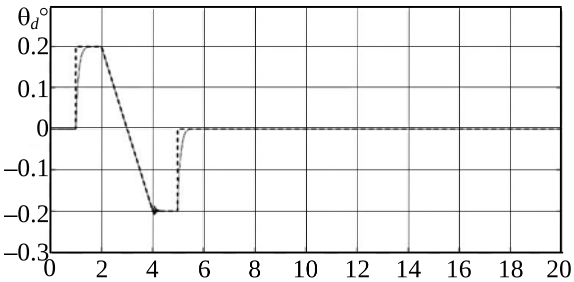

Fig. 6. Tangage: dashed line – θ_d, solid line – θ

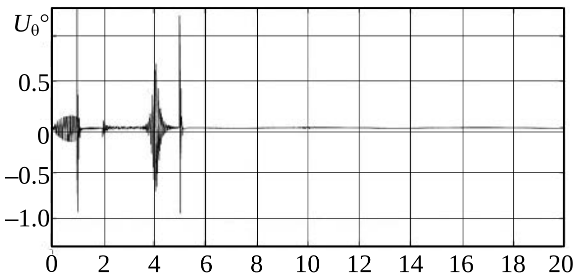

Fig. 7. Pitch control signal – U_θ

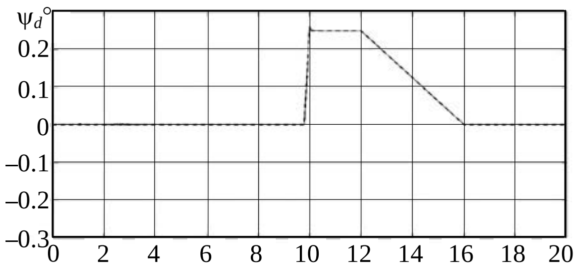

Fig. 8. Yaw: dashed line – ψ_d, solid line – ψ

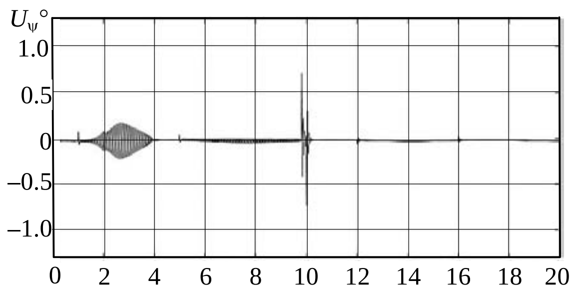

Fig. 9. Yaw control signal – U_ψ

Good quality of following the actual angle value to the desired change law. The trajectory of the center of mass when the three orientation angles are set simultaneously according to the desired change laws is shown in Figure 10.

To test the robustness of the control algorithm, some disturbance such as white noise should be applied to the quadcopter. It can simulate the wind load as an external disturbance acting on the quadrocopter. The stability of the system was confirmed in the experiment. Thus, it has been shown that the proposed approach ensures the stability of the robot control system and compensate the perturbations (from Figure 11 to Figure 16).

 ISSN 2076-2429 (print)
ISSN 2223-3814 (online)

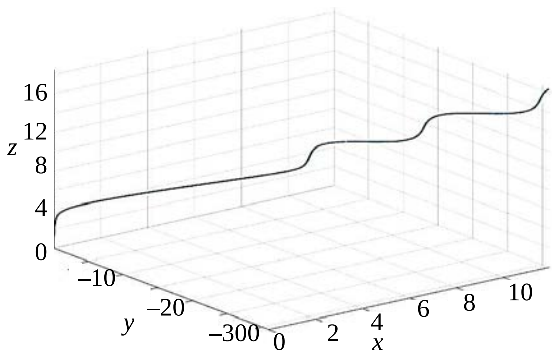

Fig. 10. Coordinates of the center of mass – *x*, *y*, *z* on a given trajectory

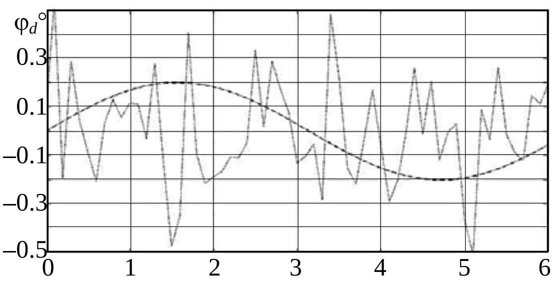

Fig. 11. Dashed line – φ_d, solid line – φ, dotted line – white noise

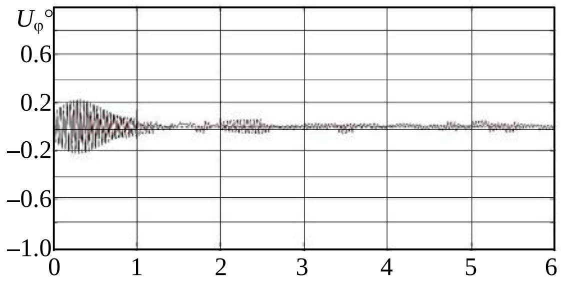

Fig. 12. The red line is with white noise, the black line is without perturbation

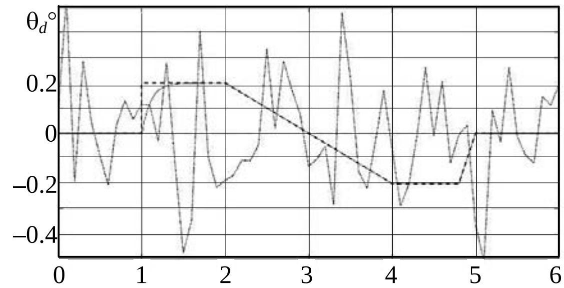

Fig. 13. Dashed line – θ_d, solid line – θ, dotted line – white noise

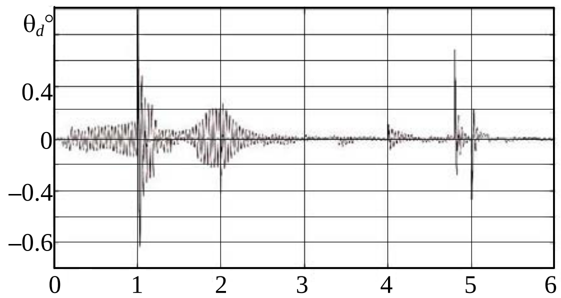

Fig. 14. The red line is with white noise, the black line is without perturbation.

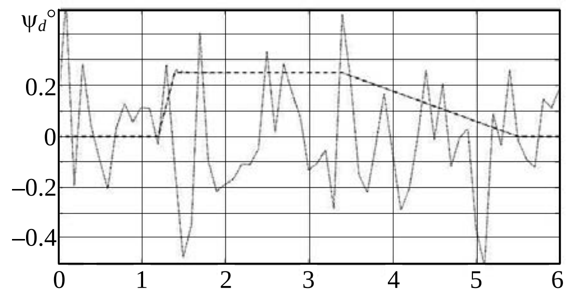

Fig. 15. Dashed line – ψ_d, solid line – ψ dotted line – white noise

Conclusions

The simulation of the system behavior shows the effectiveness of the proposed control method. Each of the orientation angles in space (roll, pitch and yaw) follows the desired trajectory with high accuracy.

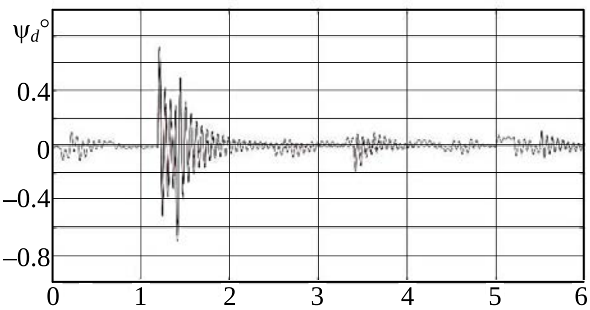

Fig. 16. The red line is with white noise, the black line is without perturbation

In this paper, a sliding mode quadrocopter control system based on a dual neural network controller was considered. The main neural network was a MIMO system that approximates the control signal for the motion of the system in the vicinity of the sliding surface. The auxiliary neural network approximates the corrective control signal needed to smooth the effect of high frequency jitter near the sliding surface. The proposed method allows controlling the system without a priori information about the parameters of the dynamic model of the controlled object. The stability of the system motion in the vicinity of the sliding surface is proved using the Lyapunov method. According to the results of modeling the dynamic model of quadrocopter with neural network regulator and in MATLAB environment, we can conclude that the proposed control method provides stable motion along the specified trajectory despite external disturbing influences. Also, the capabilities of the developed control method have been demonstrated. A new hybrid position-force controller using an adaptive neural network for controlling the motion of a robot manipulator has been proposed. It is shown that the proposed method can be used both for controlling manipulation robots and for controlling other types of robotic systems, including mechanical systems with not fully activated degrees of mobility, manipulators with both force and position control of the working tool, UAVs.

Література

1. Markazi A.H.D., Maadani M., Zabihifar S., Doost-Mohammadi N. Adaptive fuzzy sliding mode control of under-actuated nonlinear systems. *Int J Autom Comput*. 2018. Vol. 15. P. 364–376.
2. Gruber P., Balemi S. Overview of non-linear control methods. *Swiss Society for Automatic Control*. 2010. P. 1–33.
3. He W., Ge S.S., Li Y., Chew E., Ng Y.S. Neural Network Control of a Rehabilitation Robot by State and Output Feedback. *J Intell Robot Syst*. 2015. 80(1). P. 15–31. URL: http://link.springer.com/10.1007/s10846-014-0150-6.
4. Polydoros A.S., Nalpantidis L., Kruger V. Real-time deep learning of robotic manipulator inverse dynamics. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2015. P. 3442–8. URL: http://ieeexplore.ieee.org/document/7353857/.
5. Muller A. Recursive second-order inverse dynamics for serial manipulators. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 2017. P. 2483–2489. URL: http://ieeexplore.ieee.org/document/7989289/.
6. Duc M.N., Trong T.N. Neural network structures for identification of nonlinear dynamic robotic manipulator. In: *2014 IEEE International Conference on Mechatronics and Automation*. 2014. P. 1575–1580. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6885935.
7. Rahmani B., Belkheiri M. Robust Adaptive Control of Robotic Manipulators Using Neural Networks. Application to a Two Link Planar Robot. *ICMIC-2016*. 2016. P. 839–844.
8. He W., Chen Y., Yin Z., Member S. Adaptive Neural Network Control of an Uncertain Robot With Full-State Constraints. *IEEE Transactions on Cybernetics*. 2015. 46(3). P. 620–629. DOI: 10.1109/TCYB.2015.2411285.
9. Yan L., Li C.J. Robot Learning Control Based on Recurrent Neural Network Inverse Model. 1997. 14(3). P. 1–22.
10. Improving control precision and motion adaptiveness for surgical robot with recurrent neural network / Li Y., Li S., Caballero D., Miyasaka M., Lewis A., Hannaford B. *IEEE Int Conf Intell Robot Syst*. 2017. 2017–Septe. P. 3538–3543. DOI: https://doi.org/10.1109/IROS.2017.8206197.
11. He W., Ge S.S., How B.V.E., Choo Y.S. Dynamics and Control of Mechanical Systems in Offshore Engineering. London : Springer London. 2014, 230 p. URL: http://link.springer.com/10.1007/978-1-4471-5337-5.
12. Hana Boudjedir, Fouad Yacef, Omar Bouhali N.R. Dual Neural Network for Adaptive Sliding Mode Control of Quadrotor Helicopter Stabilization. *Int J Inf Sci Tech*. 2012. 2(4). P. 101–115.
13. Moreno-Valenzuela J., Aguilar-Avelar C., Puga-Guzman S.A., Santibanez V. Adaptive Neural Network Control for the Trajectory Tracking of the Furuta Pendulum. *IEEE Trans Cybern*. 2016. P. 1–14.
14. Nelson J., Kraft L.G. Real-time control of an inverted pendulum system using complementary neural network and optimal techniques. In: *Proceedings of 1994 American Control Conference - ACC '94*. IEEE. p. 2553–2554. URL: http://ieeexplore.ieee.org/document/735019/.
15. Hsu C-F. Adaptive backstepping Elman-based neural control for unknown nonlinear systems. *Neurocomputing*. 2014 Jul 20. 136. P. 170–179. URL: https://www.sciencedirect.com/science/article/pii/S0925231214001684.
16. Boudjedir H. Dual Neural Network for Adaptive Sliding Mode Control of Quadrotor Helicopter Stabilization. *Int J Inf Sci Tech*. 2012 Jul 31. 2(4). P. 1–14. URL: http://www.airccse.org/journal/IS/papers/2412ijist01.pdf.
17. Khazaee M., Markazi A.H.D., Rizi S.T., Seyfi B. Adaptive fuzzy sliding mode control of input-delayed uncertain nonlinear systems through output-feedback. *Nonlinear Dyn*. 2017. 87(3). P. 1943–1956. URL: http://link.springer.com/10.1007/s11071-016-3164-8.
18. Li M., Bi D., Xiao Z. Mechanism Simulation and Experiment of 3-DOF Parallel Robot Based on MATLAB. *Ipemec 2015*. 2015. P. 489–494. DOI: 10.2991/ipemec-15.2015.91.

ISSN 2076-2429 (print)
ISSN 2223-3814 (online)

19. Huang Y. Neural Network Based Dynamic Trajectory Tracking of Delta Parallel Robot. *Int Conf Mechatronics Autom*. 2015. P. 1938–1943.
20. Du J., Lou Y. Simplified Dynamic Model for Real-time Control of the Delta Parallel Robot. 2016. (August). P. 1647–1652.
21. Naidoo Y., Stopforth R., Bright G. Helicopter Modelling and Control. *Int J Adv Robot Sy*. 2011. 8(4). P. 139–149.
22. Van Cuong P., Nan W.Y. Adaptive trajectory tracking neural network control with robust compensator for robot manipulators. *Neural Comput Appl*. 2016. 27(2). P. 525–536. URL: http://dx.doi.org/10.1007/s00521-015-1873-4.
23. Yildirim S. Design of Adaptive Robot Control System Using Recurrent Neural Network. *J Intell Robot Syst*. 2005. 44(3). P. 247–261. URL: http://link.springer.com/10.1007/s10846-005-9012-6.
24. Alashqar H.A.H., Prof A., Hussein M.T. Modeling and High Precision Motion Control of 3 DOF Parallel Delta Robot Manipulator. 2013.
25. Tanaka Y., Tsuji T. On-line Learning of Robot Arm Impedance Using Neural Networks. In: 2004 *IEEE International Conference on Robotics and Biomimetics. IEEE*. 2004. P. 941–946. URL: http://ieeexplore.ieee.org/document/1521911/.
26. Brandt R.D., Lin F. Adaptive interaction and its application to neural networks. *Inf Sci (Ny)*. 1999. 121(3–4). P. 201–215. URL: https://www.sciencedirect.com/science/article/pii/S0020025599000900.
27. Olfati-Saber R. Fixed point controllers and stabilization of the cart-pole system and the rotating pendulum. In: *Proceedings of the 38th IEEE Conference on Decision and Control* (Cat No99CH36304). IEEE. 1999. P. 1174–1181. URL: http://ieeexplore.ieee.org/document/830086/.
28. Turker T., Gorgun H., Cansever G. Lyapunov's direct method for stabilization of the Furuta pendulum. *Turkish J Electr Eng Comput Sci*. 2012. 20(1). P. 99–110.

References

1. Markazi, A.H.D., Maadani, M., Zabihifar, S., & Doost-Mohammadi, N. (2018). Adaptive fuzzy sliding mode control of under-actuated nonlinear systems. *Int J Autom Comput*., 15, 364–76.
2. Gruber, P., & Balemi, S. (2010). Overview of non-linear control methods. *Swiss Society for Automatic Control*, 1–33.
3. He, W., Ge, S.S., Li, Y., Chew, E., & Ng, Y.S. (2015). Neural Network Control of a Rehabilitation Robot by State and Output Feedback. *J Intell Robot Syst.*, 80(1), 15–31. Retrieved from: http://link.springer.com/10.1007/s10846-014-0150-6.
4. Polydoros, A.S., Nalpantidis, L., & Kruger, V. (2015). Real-time deep learning of robotic manipulator inverse dynamics. *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3442–3448. Retrieved from: http://ieeexplore.ieee.org/document/7353857/.
5. Muller, A. (2017). Recursive second-order inverse dynamics for serial manipulators. *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2483–2489. Retrieved from: http://ieeexplore.ieee.org/document/7989289/.
6. Duc, M.N., & Trong, T.N. (2014). Neural network structures for identification of nonlinear dynamic robotic manipulator. *2014 IEEE International Conference on Mechatronics and Automation*, 1575–1580. Retrieved from: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6885935.
7. Rahmani, B., & Belkheiri, M. (2016). Robust Adaptive Control of Robotic Manipulators Using Neural Networks : Application to a Two Link Planar Robot. *ICMIC-2016*, 839–844.
8. He, W., Chen, Y., Yin, Z., & Member, S. (2015). Adaptive Neural Network Control of an Uncertain Robot With Full-State Constraints. *IEEE Transactions on Cybernetics*, 46(3), 620–629. DOI: 10.1109/TCYB.2015.2411285.
9. Yan, L., & Li, C.J. (1997). Robot Learning Control Based on Recurrent Neural Network Inverse Model, 14(3), 1–22.
10. Li, Y., Li, S., Caballero, D., Miyasaka, M., & Lewis, A. (2017). Hannaford B. Improving control precision and motion adaptiveness for surgical robot with recurrent neural network. *IEEE Int Conf Intell Robot Syst*, 3538–3543.
11. He, W., Ge, S.S., How, B.V.E., & Choo, Y.S. (2014). *Dynamics and Control of Mechanical Systems in Offshore Engineering*. London: Springer London. Retrieved from: http://link.springer.com/10.1007/978-1-4471-5337-5.
12. Hana Boudjedir, Fouad Yacef, & Omar Bouhali N.R. (2012). Dual Neural Network for Adaptive Sliding Mode Control of Quadrotor Helicopter Stabilization. *Int J Inf Sci Tech*, 2(4),101–15.
13. Moreno-Valenzuela, J., Aguilar-Avelar, C., Puga-Guzman, S.A., & Santibanez, V. (2016). Adaptive Neural Network Control for the Trajectory Tracking of the Furuta Pendulum. *IEEE Trans Cybern.* 1–14.

14. Nelson, J., & Kraft, L.G. (1994). Real-time control of an inverted pendulum system using complementary neural network and optimal techniques. In: Proceedings of 1994 American Control Conference - ACC '94, p. 2553–2554. Retrieved from: http://ieeexplore.ieee.org/document/735019/.
15. Hsu, C-F. (2014). Adaptive backstepping Elman-based neural control for unknown nonlinear systems. *Neurocomputing,* 136, 170–179. Retrieved from: https://www.sciencedirect.com/science/article/pii/S0925231214001684.
16. Boudjedir, H. (2012). Dual Neural Network for Adaptive Sliding Mode Control of Quadrotor Helicopter Stabilization. *Int J Inf Sci Tech*, 2(4), 1–14. Retrieved from: http://www.airccse.org/journal/IS/papers/2412ijist01.pdf.
17. Khazaee, M., Markazi, A.H.D., Rizi, S.T., & Seyfi, B. (2017). Adaptive fuzzy sliding mode control of input-delayed uncertain nonlinear systems through output-feedback. *Nonlinear Dyn,* 87(3), 1943–1956. Retrieved from: http://link.springer.com/10.1007/s11071-016-3164-8.
18. Li, M., Bi, D., & Xiao, Z. (2015). Mechanism Simulation and Experiment of 3-DOF Parallel Robot Based on MATLAB. *2015 Ipemec*, 489–494.
19. Huang, Y. (2015). Neural Network Based Dynamic Trajectory Tracking of Delta Parallel Robot. *Int Conf Mechatronics Autom*., 1938–1943.
20. Du, J., & Lou, Y. (2016). Simplified Dynamic Model for Real-time Control of the Delta Parallel Robot. 1647–1652.
21. Naidoo, Y., Stopforth, R., & Bright, G. (2011). Helicopter Modelling and Control. *Int J Adv Robot Sy*., 8(4), 139–149.
22. Van Cuong, P., & Nan, W.Y. (2016). Adaptive trajectory tracking neural network control with robust compensator for robot manipulators. *Neural Comput Appl*, 27(2), 525–536. Retrieved from: http://dx.doi.org/10.1007/s00521-015-1873-4.
23. Yildirim, S. (2005). Design of Adaptive Robot Control System Using Recurrent Neural Network. *J Intell Robot Syst*, 44(3), 247–261. Retrieved from: http://link.springer.com/10.1007/s10846-005-9012-6.
24. Alashqar, H.A.H., & Hussein, M.T. (2013). Modeling and High Precision Motion Control of 3 DOF Parallel Delta Robot Manipulator.
25. Tanaka, Y., & Tsuji, T. (2004). On-line Learning of Robot Arm Impedance Using Neural Networks. *2004 IEEE International Conference on Robotics and Biomimetics*, 941–946. Retrieved from: http://ieeexplore.ieee.org/document/1521911/.
26. Brandt, R.D., & Lin, F. (1999). Adaptive interaction and its application to neural networks. *Inf Sci (Ny),* 121 (3–4), 201–215. Retrieved from: https://www.sciencedirect.com/science/article/pii/S0020025599000900.
27. Olfati-Saber, R. (1999). Fixed point controllers and stabilization of the cart-pole system and the rotating pendulum. *Proceedings of the 38th IEEE Conference on Decision and Control (Cat No99CH36304),* 1174–1181. Retrieved from: http://ieeexplore.ieee.org/document/830086/.
28. Turker, T., Gorgun, H., & Cansever, G. (2012). Lyapunov's direct method for stabilization of the Furuta pendulum. *Turkish J Electr Eng Comput Sci*., 20(1), 99–110.

Тігарєв Володимир Михайлович; Volodymyr Tigariev, ORCID: https://orcid.org/0000-0001-8492-6633
Лопаков Олексій Сергійович; Oleksii Lopakov, ORCID: https://orcid.org/0000-0001-6307-8946
Космачевський Володимир Володимирович; Volodymir Kosmachevskiy, ORCID: https://orcid.org/0000-0002-3234-2297
Коляда Андрій Сергійович; Andrii Koliada, ORCID: https://orcid.org/0000-0003-3934-8058
Львов Іван Дмитрович; Ivan Lvov
Дмитренко Дмитро Олександрович; Dmytro Dmytrenko, ORCID: https://orcid.org/0009-0004-9948-9682

Received October 03, 2024

Accepted November 21, 2024