

NEURO-FUZZY CONTROL OF A ROBOTIC MANIPULATOR

P. GIERLAK, M. MUSZYŃSKA* and W. ŻYLSKI

Department of Applied Mechanics and Robotics

Faculty of Mechanical Engineering and Aeronautics

Rzeszow University of Technology

al. Powstańców Warszawy 12, 35-959 Rzeszów, POLAND

E-mails: pgierlak@prz.edu.pl; magdaw@prz.edu.pl; wzylski@prz.edu.pl

In this paper, to solve the problem of control of a robotic manipulator's movement with holonomical constraints, an intelligent control system was used. This system is understood as a hybrid controller, being a combination of fuzzy logic and an artificial neural network. The purpose of the neuro-fuzzy system is the approximation of the nonlinearity of the robotic manipulator's dynamic to generate a compensatory control. The control system is designed in such a way as to permit modification of its properties under different operating conditions of the two-link manipulator.

Key words: robotic manipulator, force control, neuro-fuzzy system.

1. Introduction

The issue of the constrained control of manipulator, often finds practical applications in robotised industrial processes (Gierlak, 2014; 2013). If an end-effector of the manipulator comes in contact with an environment, the interacting forces are very important. These forces are generated at the contact point and are normal or tangential to the contact surface. In order to correct the realisation of the control process, interacting forces, normal to the contact surface, must be controlled. Simultaneously, the movement of the end-effector on the contact surface is realised. This leads to the hybrid position-force control method (Gierlak, 2012; 2013).

Robotic manipulators are systems with nonlinear dynamic descriptions, in which the problem of uncertainty modelling occurs. In order to realise the control process with good quality, the control law should take into account the compensation of object nonlinearities (Gierlak, 2012). At present, there are many modern control techniques for nonlinear systems, which base on the theory of adaptive (Chen and Billings, 1996; Gierlak, 2013; Kumar *et al.*, 2011) and robust systems. They are particularly useful in cases of control issues such as the one analyzed in this article.

In this paper, to solve the problem of the constrained motion control of a robot, an intelligent control system understood as hybrid combination of fuzzy logic and artificial neural networks was used. The purpose of the neuro-fuzzy system is to approximate the manipulator's dynamic nonlinearities in order to generate the compensatory control. The control system is designed to keep modifying its properties under different operating conditions of the two-link manipulator. Simulation studies have been carried out for the case when the end-effector of the manipulator moves along a circular path and puts pressure on the contact surface.

* To whom correspondence should be addressed

2. Description of control object

The control object is a two-link manipulator, schematically presented in Fig.1.

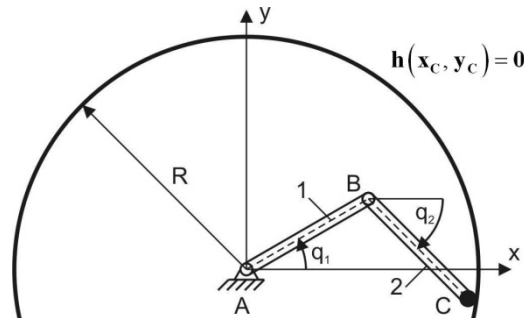


Fig.1. The two-link constrained manipulator.

The kinematics of the manipulator is described by the following equation

$$\mathbf{y} = \begin{bmatrix} x_C \\ y_C \end{bmatrix} = \mathbf{k}(\mathbf{q}) = \begin{bmatrix} l_1 \cos q_1 + l_2 \cos q_2 \\ l_1 \sin q_1 + l_2 \sin q_2 \end{bmatrix} \quad (2.1)$$

where $\mathbf{q} = [q_1, q_2]^T$ – the vector of generalised coordinates, l_1, l_2 – link's length. Assuming, that the point C will remain on the surface, it is described in the following way

$$\mathbf{h}(\mathbf{k}) = x_C^2 + y_C^2 - R^2 = 0. \quad (2.2)$$

It is an equation of holonomical constraints imposed on the manipulator's end-effector, which can be written in terms of generalised coordinates

$$\mathbf{h}(\mathbf{q}) = l_1^2 + l_2^2 + 2l_1l_2 \cos(q_1 - q_2) - R^2 = 0. \quad (2.3)$$

The Jacobian associated with the constraints has the form

$$\mathbf{J}(\mathbf{q}) = \frac{\partial \mathbf{h}(\mathbf{q})}{\partial \mathbf{q}} = \begin{bmatrix} \frac{\partial \mathbf{h}(\mathbf{q})}{\partial q_1} & \frac{\partial \mathbf{h}(\mathbf{q})}{\partial q_2} \end{bmatrix} = \begin{bmatrix} -2l_1l_2 \sin(q_1 - q_2) & 2l_1l_2 \sin(q_1 - q_2) \end{bmatrix}. \quad (2.4)$$

Dynamical equations of motion of the manipulator have the following form

$$\begin{bmatrix} a_1 & a_2 \cos(q_2 - q_1) \\ a_2 \cos(q_2 - q_1) & a_3 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} 0 & -a_2 \sin(q_2 - q_1) \dot{q}_2 \\ a_2 \sin(q_2 - q_1) \dot{q}_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} a_4 \dot{q}_1 \\ a_5 \dot{q}_2 \end{bmatrix} + \begin{bmatrix} a_6 \cos q_1 \\ a_7 \cos q_2 \end{bmatrix} + \begin{bmatrix} \tau_{d1} \\ \tau_{d2} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} -2l_1l_2 \sin(q_1 - q_2) \\ 2l_1l_2 \sin(q_1 - q_2) \end{bmatrix} \lambda \quad (2.5)$$

where a_i – parameters dependent on the geometry and mass distribution of links and friction coefficients, u_1 and u_2 – driving moments, τ_{d1} and τ_{d2} - disturbance, λ - Lagrange multiplier – in this case, the force normal to the constrain surface. In a matrix notation Eq.(2.5) has the form (Lewis, 1999)

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + F(\dot{q}) + G(q) + \tau_d(t) = u + J^T(q)\lambda \tag{2.6}$$

The imposition of the constraints Eq.(2.3) reduced the number of degrees of freedom of the manipulator from 2 to 1, so its movement may be described by using an arbitrarily selected reduced position variable $\theta_1=q_1$. From the assumed constraints the dependent variable takes form

$$\theta_2=q_2 = \gamma(q_1) = q_1 - \arccos \frac{R^2 - l_1^2 - l_2^2}{2l_1l_2}, \quad \dot{q}_2 = \dot{q}_1, \quad \ddot{q}_2 = \ddot{q}_1. \tag{2.7}$$

Some extended Jacobian L, that defines the relations between velocities of the unconstrained and constrained system, has the form

$$\dot{q} = L\dot{\theta}_1 = \begin{bmatrix} 1 \\ \partial\gamma(q_1)/\partial\theta_1 \end{bmatrix} \dot{\theta}_1. \tag{2.8}$$

In the case presented here $L=[1, l]^T$. Substituting $\gamma(q_1)$ and Eq.(2.8) to Eq.(2.5), yielded

$$\begin{aligned} & \begin{bmatrix} a_1 + a_2h \\ a_2h + a_3 \end{bmatrix} \ddot{\theta}_1 + \begin{bmatrix} a_2\sqrt{1-h^2}\dot{\theta}_1 \\ -a_2\sqrt{1-h^2}\dot{\theta}_1 \end{bmatrix} \dot{\theta}_1 + \begin{bmatrix} a_4\dot{\theta}_1 \\ a_5\dot{\theta}_1 \end{bmatrix} + \begin{bmatrix} a_6\cos\theta_1 \\ a_7\cos(\theta_1 - \text{arccosh}) \end{bmatrix} + \begin{bmatrix} \tau_{d1} \\ \tau_{d2} \end{bmatrix} = \\ & = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} -2l_1l_2\sqrt{1-h^2} \\ 2l_1l_2\sqrt{1-h^2} \end{bmatrix} \lambda \end{aligned} \tag{2.9}$$

where $h = \frac{R^2 - l_1^2 - l_2^2}{2l_1l_2}$. Multiplying Eq.(2.9) by L^T and taking into account that $JL=0$, we can write

$$L^T \left(\begin{bmatrix} a_1 + a_2h \\ a_2h + a_3 \end{bmatrix} \ddot{\theta}_1 + \begin{bmatrix} a_2\sqrt{1-h^2}\dot{\theta}_1 \\ -a_2\sqrt{1-h^2}\dot{\theta}_1 \end{bmatrix} \dot{\theta}_1 + \begin{bmatrix} a_4\dot{\theta}_1 \\ a_5\dot{\theta}_1 \end{bmatrix} + \begin{bmatrix} a_6\cos\theta_1 \\ a_7\cos(\theta_1 - \text{arccosh}) \end{bmatrix} + \begin{bmatrix} \tau_{d1} \\ \tau_{d2} \end{bmatrix} \right) = L^T \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{2.10}$$

and in the matrix notation

$$\bar{M}\ddot{\theta}_1 + \bar{V}_1\dot{\theta}_1 + \bar{F} + \bar{G} + \bar{\tau}_d = L^T u \tag{2.11}$$

$$\text{where } \bar{\mathbf{M}} = \mathbf{L}^T \begin{bmatrix} a_1 + a_2 h \\ a_2 h + a_3 \end{bmatrix}, \quad \bar{\mathbf{V}}_1 = \mathbf{L}^T \begin{bmatrix} a_2 \sqrt{1-h^2} \dot{\theta}_1 \\ -a_2 \sqrt{1-h^2} \dot{\theta}_1 \end{bmatrix}, \quad \bar{\mathbf{F}} = \mathbf{L}^T \begin{bmatrix} a_4 \dot{\theta}_1 \\ a_5 \dot{\theta}_1 \end{bmatrix}, \quad \bar{\mathbf{G}} = \mathbf{L}^T \begin{bmatrix} a_6 \cos \theta_1 \\ a_7 \cos(\theta_1 - \text{arccosh}) \end{bmatrix},$$

$\bar{\boldsymbol{\tau}}_d = \mathbf{L}^T \boldsymbol{\tau}_d = \mathbf{L}^T \begin{bmatrix} \tau_{d1} \\ \tau_{d2} \end{bmatrix}$, $\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$. The above equation describes the motion on the contact surface, but it must be noted that it has no information about the downforce.

3. Position-force control

The purpose of the hybrid position-force control is the realisation of the desired positional trajectory $\theta_{1d}(t)$ and the desired force trajectory $\lambda_d(t)$. Defining the motion error e_θ , a filtered motion error s , a force error $\tilde{\lambda}$ and an auxiliary variable v_1 (Levis *et al.*, 1999)

$$e_\theta = \theta_{1d} - \theta_1, \quad s = \dot{e}_\theta + \Lambda e_\theta, \quad \tilde{\lambda} = \lambda_d - \lambda, \quad v_1 = \dot{\theta}_{1d} + \Lambda e_\theta \quad (3.1)$$

where Λ – is a positive constant. Equation (2.10) expressed in these variables has the form

$$\mathbf{L}^T \left(\begin{bmatrix} a_1 + a_2 h \\ a_2 h + a_3 \end{bmatrix} (\dot{v}_1 - \dot{s}) + \begin{bmatrix} a_2 \sqrt{1-h^2} \dot{\theta}_1 \\ -a_2 \sqrt{1-h^2} \dot{\theta}_1 \end{bmatrix} (v_1 - s) + \begin{bmatrix} a_4 \dot{\theta}_1 \\ a_5 \dot{\theta}_1 \end{bmatrix} + \begin{bmatrix} a_6 \cos \theta_1 \\ a_7 \cos(\theta_1 - \text{acosh}) \end{bmatrix} + \begin{bmatrix} \tau_{d1} \\ \tau_{d2} \end{bmatrix} \right) = \mathbf{L}^T \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}. \quad (3.2)$$

The reduced dynamics, in terms of the filtered motion error, was converted to the form

$$\mathbf{L}^T \begin{bmatrix} a_1 + a_2 h \\ a_2 h + a_3 \end{bmatrix} \dot{s} = -\mathbf{L}^T \begin{bmatrix} a_2 \sqrt{1-h^2} \dot{\theta}_1 \\ -a_2 \sqrt{1-h^2} \dot{\theta}_1 \end{bmatrix} s + \mathbf{L}^T \mathbf{f} + \mathbf{L}^T \begin{bmatrix} \tau_{d1} \\ \tau_{d2} \end{bmatrix} - \mathbf{L}^T \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad (3.3)$$

$$\bar{\mathbf{M}} \dot{s} = -\bar{\mathbf{V}}_1 s + \mathbf{L}^T \mathbf{f} + \bar{\boldsymbol{\tau}}_d - \mathbf{L}^T \mathbf{u}$$

where

$$\mathbf{f} = \begin{bmatrix} a_1 + a_2 h \\ a_2 h + a_3 \end{bmatrix} \dot{v}_1 + \begin{bmatrix} a_2 \sqrt{1-h^2} \dot{\theta}_1 \\ -a_2 \sqrt{1-h^2} \dot{\theta}_1 \end{bmatrix} v_1 + \begin{bmatrix} a_4 \dot{\theta}_1 \\ a_5 \dot{\theta}_1 \end{bmatrix} + \begin{bmatrix} a_6 \cos \theta_1 \\ a_7 \cos(\theta_1 - \text{arccosh}) \end{bmatrix}. \quad (3.4)$$

In order to realise the position-force control, the control was assumed in the form

$$\mathbf{u} = \mathbf{u}_C + \mathbf{u}_{PD} - \mathbf{u}_F - \boldsymbol{\zeta} = \hat{\mathbf{f}} + \mathbf{K}_D \mathbf{L} s - \mathbf{J}^T [\lambda_d + K_F \tilde{\lambda}] - \boldsymbol{\zeta} \quad (3.5)$$

where $\boldsymbol{\zeta}$ – the robust term, $\mathbf{u}_{PD} = \mathbf{K}_D \mathbf{L} s$ – the form of PD control, where \mathbf{K}_D – the positive defined matrix of position gain, $\mathbf{u}_F = -\mathbf{J}^T [\lambda_d + K_F \tilde{\lambda}]$ – the force control, where K_F – the positive force gain, $\mathbf{u}_C = \hat{\mathbf{f}}$ – an approximation of the nonlinear function Eq.(3.4), seen as the compensatory control. The scheme of the closed-loop control system is shown in Fig.2.

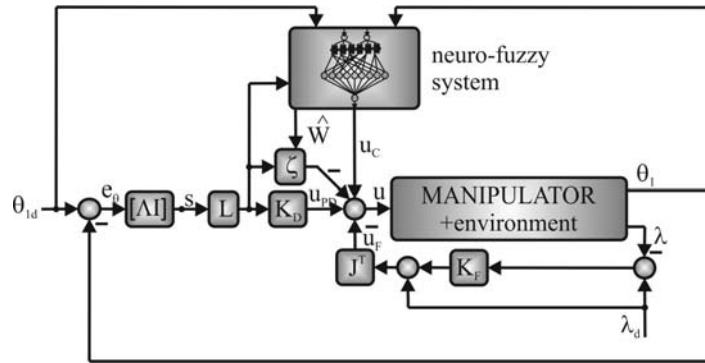


Fig.2. The scheme of the closed-loop position-force control system.

Nonlinearity in the form of Eq.(3.4) is approximated with the neuro-fuzzy system. Because of the explosion of solutions resulting from the large number of input variables, this function was broken up into 6 components of the function given in the equation

$$f_{nf1} = g_1 + g_2 + g_3, \quad f_{nf2} = g_4 + g_5 + g_6 \tag{3.6}$$

where

$$\begin{aligned} g_1 &= (a_1 + a_2h)\dot{v}_1, & g_2 &= a_2\sqrt{1-h^2}\dot{\theta}_1v_1, & g_3 &= a_4\dot{\theta}_1 + a_6\cos\theta_1, \\ g_4 &= (a_2h + a_3)\dot{v}_1, & g_5 &= -a_2\sqrt{1-h^2}\dot{\theta}_1v_1, & g_6 &= a_5\dot{\theta}_1 + a_7\cos(\theta_1 - \text{arccosh}). \end{aligned} \tag{3.7}$$

Each of these functions has two input signals which allowed the use of this structure in real time.

4. Neuro-fuzzy nonlinearity compensator

The neuro-fuzzy system was used for approximation of nonlinear functions. In this system the parameters of the conclusion and the premise of the rule base (Hendzel and Muszyńska, 2012; Muszyńska, 2012) of the Sugeno model must be learned. Rules have the form

$$R_j: \text{IF}(x_1 = A_{j1}) \text{AND}(x_2 = B_{j1}) \text{THEN}(g = w_j), \quad j = 1, 2, \dots, N \tag{4.1}$$

where x_1, x_2 - input signals, A_{j1} and B_{j1} - fuzzy sets represented by the Gaussian functions; parameters of these sets are the width and the center, w_j - conclusion of the rule base. Using a singleton defuzzification, assuming the Gaussian membership function, and the degree of compliance with the premise, we can write the fuzzy model in the form

$$g = \sum_{j=1}^N w_j \phi_j = \mathbf{W}^T \Phi, \quad \mathbf{W}^T = [w_1, \dots, w_N], \quad \Phi = [\phi_1, \dots, \phi_N]^T. \tag{4.2}$$

The degree to meet the premise of the given rule is adopted in the form

$$\phi_j = \mu_{A_j}(x_1) \cdot \mu_{B_j}(x_2) \tag{4.3}$$

where $\mu_{A_f}(x_1)$ and $\mu_{B_f}(x_2)$ - fuzzy sets described by the Gaussian function

$$\mu_{(\cdot)ji}(x_i) = \exp\left(-r_{ji}^2(x_i - c_{ji})^2\right) \quad (4.4)$$

where r_{ji} , c_{ji} - width and center of this function. Considering Eq.(4.4), the degree to meet the premise of the rule (4.1) is written as

$$\phi_j = \exp\left(-r_{j1}^2(x_1 - c_{j1})^2 - r_{j2}^2(x_2 - c_{j2})^2\right). \quad (4.5)$$

This type of assumption makes it possible to describe the nonlinearity of the robot in the form of Eq.(4.6) interpreted as the neuro-fuzzy model, which will undergo learning in real time. Each of g_i functions defined by Eqs (3.7) can be expressed as a product of weight with an index and the degree to meet the premise

$$\hat{\mathbf{f}} = \begin{bmatrix} f_{nf1} \\ f_{nf1} \end{bmatrix} = \begin{bmatrix} \mathbf{W}^{(1)T} \Phi^{(1)} + \mathbf{W}^{(2)T} \Phi^{(2)} + \mathbf{W}^{(3)T} \Phi^{(3)} \\ \mathbf{W}^{(4)T} \Phi^{(4)} + \mathbf{W}^{(5)T} \Phi^{(5)} + \mathbf{W}^{(6)T} \Phi^{(6)} \end{bmatrix} \quad (4.6)$$

where $\mathbf{W}^{(k)T} = [w_1^{(k)}, \dots, w_9^{(k)}]$ - the vector of conclusions of the rule base, $\Phi^{(k)} = [\phi_1^{(k)}, \dots, \phi_9^{(k)}]^T$ - the vector of premise evaluation of the rule base. As a result of the approximation of nonlinearities of the robot, and taking into account the aspect of the linearization of functions describing the fuzzy sets, the description of the closed-loop system is obtained as

$$\begin{aligned} \bar{\mathbf{M}}\dot{\mathbf{s}} = & -\bar{\mathbf{V}}_1\mathbf{s} - \mathbf{L}^T \mathbf{K}_D \mathbf{L}\mathbf{s} + \mathbf{L}^T \tilde{\mathbf{W}}_{nf}^T (\hat{\Phi}_{nf} - \mathbf{A}^T \hat{\mathbf{r}}_{ji} - \mathbf{B}^T \hat{\mathbf{c}}_{ji}) + \\ & + \mathbf{L}^T \hat{\mathbf{W}}_{nf} (\mathbf{A}^T \tilde{\mathbf{r}}_{ji} + \mathbf{B}^T \tilde{\mathbf{c}}_{ji}) + \mathbf{L}^T \mathbf{d}_f + \mathbf{L}^T \boldsymbol{\varsigma}. \end{aligned} \quad (4.7)$$

Parameters of the neuro-fuzzy sets are tuned on-line based on the correlation of Eqs (4.8)-(4.9) with a robust control signal (4.10)

$$\dot{\hat{\mathbf{W}}}_{nf} = \mathbf{F}_W \hat{\Phi}_{nf} (\mathbf{L}\mathbf{s})^T - \mathbf{F}_W (\mathbf{A}^T \hat{\mathbf{r}}_{ji} + \mathbf{B}^T \hat{\mathbf{c}}_{ji}) (\mathbf{L}\mathbf{s})^T - k\mathbf{F}_W \|\mathbf{L}\mathbf{s}\| \hat{\mathbf{W}}_{nf} \quad (4.8)$$

$$\dot{\tilde{\mathbf{r}}}_{ji}^k = \mathbf{F}_r \mathbf{A} \hat{\mathbf{W}}_{nf} \mathbf{L}\mathbf{s} - k\mathbf{F}_r \|\mathbf{L}\mathbf{s}\| \tilde{\mathbf{r}}_{ji}, \quad \dot{\tilde{\mathbf{c}}}_{ji}^k = \mathbf{F}_c \mathbf{B} \hat{\mathbf{W}}_{nf} \mathbf{L}\mathbf{s} - k\mathbf{F}_c \|\mathbf{L}\mathbf{s}\| \tilde{\mathbf{c}}_{ji}, \quad (4.9)$$

$$\boldsymbol{\varsigma} = -\frac{\mathbf{L}\mathbf{s}}{\|\mathbf{L}\mathbf{s}\|} \mathbf{D}^T \mathbf{Y} \quad (4.10)$$

where $\hat{\mathbf{W}}_{nf}$, $\tilde{\mathbf{W}}_{nf}$ - the estimate of the conclusion and its error, $\hat{\Phi}_{nf}$, $\tilde{\Phi}_{nf}$ - the estimate of the premise and its error, $\mathbf{Y}(\mathbf{d}_f, \hat{\mathbf{W}}, \hat{\mathbf{r}}_{ji}, \hat{\mathbf{c}}_{ji})$ - a matrix of measurable signals. All of the signals described by Eqs (4.8)-(4.10) arise from the analysis of the stability of the closed-loop control system.

An exemplary implementation of the presented adaptive approach to determine the parameters of the conclusions and premises is symbolically shown in Fig.3. This figure demonstrates only the implementation of nonlinearities included in the function f_{nf1} . This structure results from the fuzzy system.

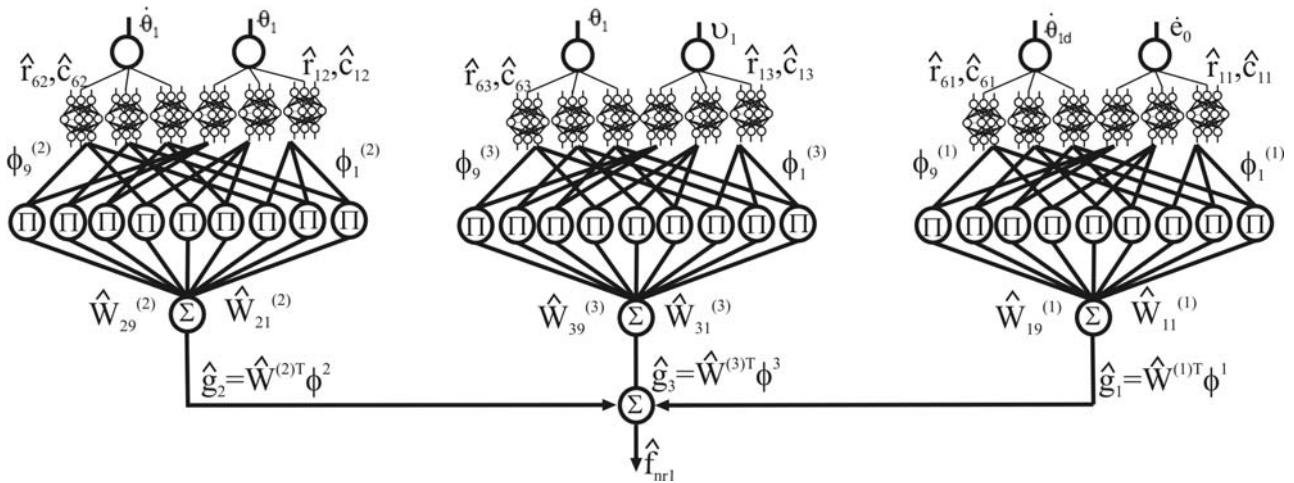


Fig.3. Diagram of neuro-fuzzy system.

5. Numerical test results

In this work, the Matlab/Simulink package is employed for the simulation of the proposed solutions of the neuro-fuzzy control of a robotic manipulator. A simulation study is carried out for the case when the tip of the manipulator moves along a path in the shape of a circle and exerts pressure on the contact surface. The simulation involved the adaptation of the conclusions and premises of the rule base. The distribution of fuzzy sets in the space considerations and parameters of the conclusions were adopted in accordance with Eqs (4.8)-(4.9).

The assumed values of control gains are: $K_D = \text{diag}\{1, 1\}$, $\Lambda = 1$, $K_F = 1$, gain of adaptation of conclusions $F_w = 0.05$, gain of adaptation of premise's centers $F_c = 1.5$ and width $F_r = 0.08$. The desired velocity and path of motion of the end-effector and the desired contact force are presented in Fig.4.

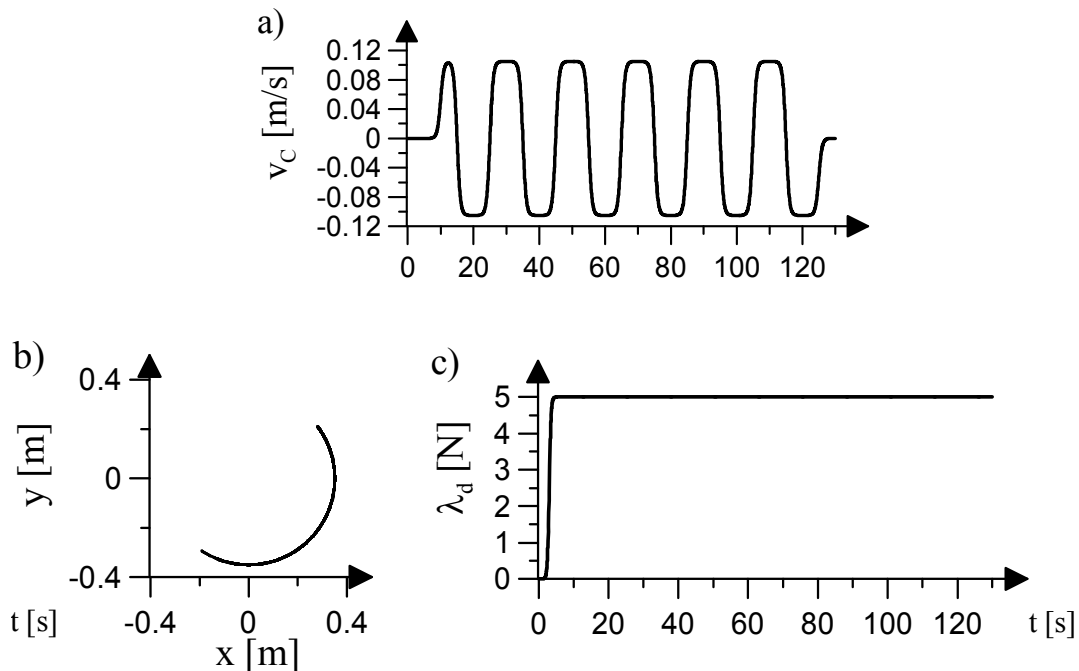


Fig.4. a) desired velocity of end-effector motion, b) the path of the end-effector, c) the desired force.

From the velocity and path the desired trajectory in the joint space arises, as a solution of inverse kinematics problem. This trajectory should be realised by the control system. The resulting control signals in the simulation are shown in Fig.5. Total control is generated in such a way as to realize the prescribed trajectory and provide the desired pressure of the manipulator's tip on the contact surface. The analysis of the results presented in Fig.6 indicates that both the rule base conclusions, the widths and the centers of the premises of the rule base adapt during the robot's movement. Oscillations of error $\tilde{\lambda}$ at the level of 5% of the reference value are due to the inaccuracies of force and position control, which caused pulsation of the downforce.

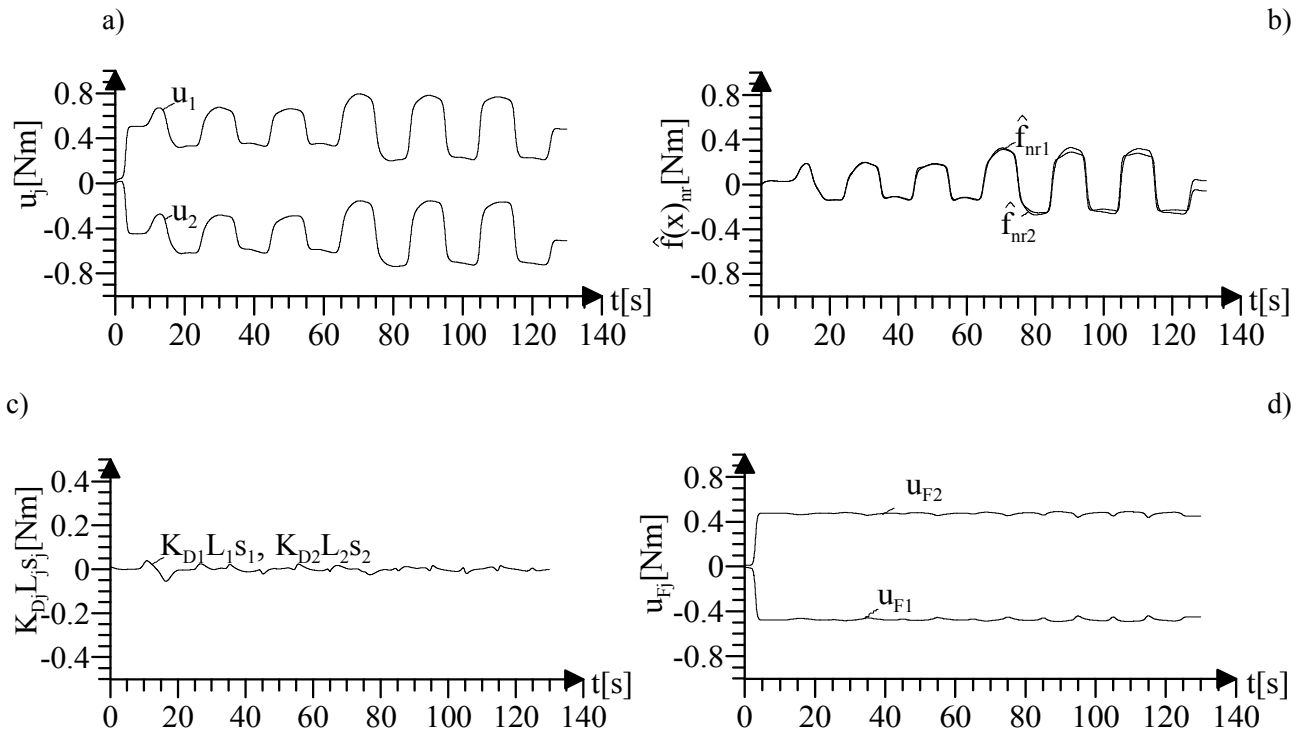
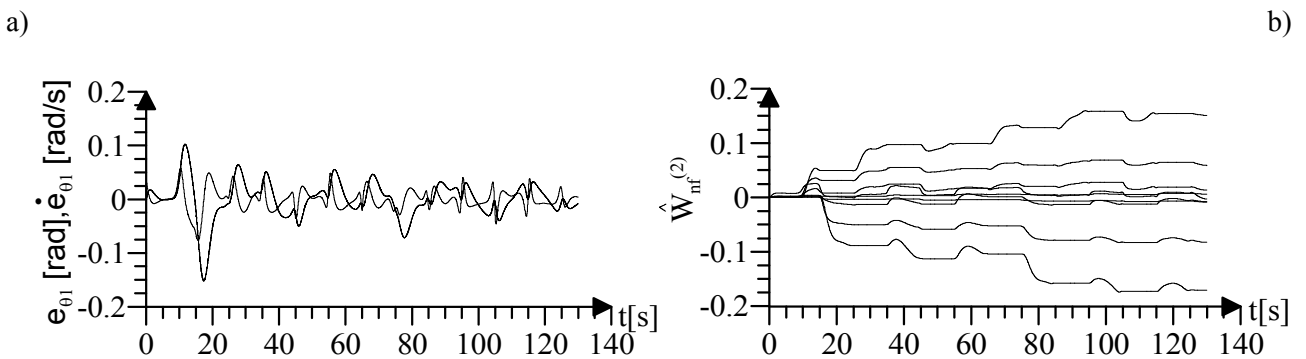


Fig.5. a) Total control, b) compensatory control signals received during the adaptation of conclusions and premises of the neuro-fuzzy rule base, c) PD control signals, d) force control signals.



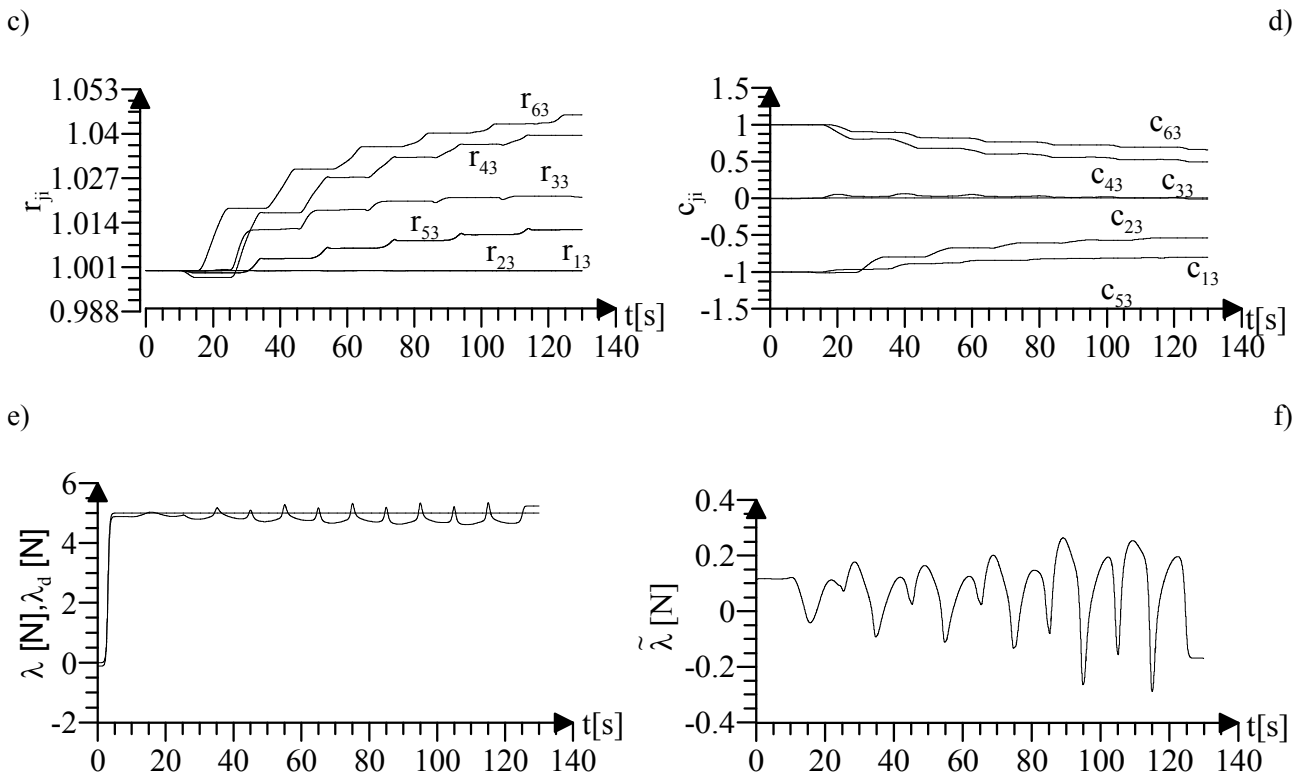


Fig.6. a) Motion errors, b) weights of the rule base conclusions, c) widths, d) centers of the Gaussian functions, e) the desired and realised contact force, f) force error.

6. Conclusion

The effectiveness of the neuro-fuzzy control algorithm designed is confirmed by the tests. They confirm the validity of the control method. The results obtained show that the use of neuro-fuzzy compensation for nonlinearity of the robotic manipulator, in the task of tracking, is very beneficial. The analysis of the results of the numerical verification shows convergence of movement errors to zero. It is caused by taking into account the compensatory control signal from the neuro-fuzzy system. In addition, global stability of the closed-loop control system is ensured in the sense that the signals are limited.

The proposed control method of a nonlinear object, which is the robotic manipulator, is a tool which uses neuro-fuzzy information in a very efficient manner.

Nomenclature

- A_{jl}, B_{jl} – fuzzy sets represented by the Gaussian functions
- a_i – parameters dependent on geometry and mass distribution of
- e_{θ} – motion error
- $F^{(k)}$ – vector of premises evaluation of the rule base
- K_D – positive defined matrix of position gain
- K_F – positive force gain
- q – vector of generalized coordinates
- r_{ji}, c_{ji} – width and center Gaussian function
- u_1, u_2 – drive moments
- u_C – approximation of the nonlinear function
- u_F – force control

- u_{PD} – PD control
 $W^{(k)T}$ – vector of conclusions of the rule base
 $\widehat{W}_{nf}, \widetilde{W}_{nf}$ – estimate of conclusion and its error
 w_j – conclusion of the rule base
 x_1, x_2 – input signals
 Y – matrix of measurable signals
 ζ – robust term
 $\theta_{1d}(t)$ – desired positional trajectory
 λ – Lagrange multiplier (force normal to the constrain surface)
 $\lambda_d(t)$ – desired force trajectory
 $\hat{\lambda}$ – force error
 $\mu_{A_j}(x_1)$ – fuzzy set described by the Gaussian function
 τ_{d1}, τ_{d2} – disturbance
 v_1 – auxiliary variable
 $\sim \hat{\phi}_{nf}, \tilde{\phi}_{nf}$ – estimate of premise and its error

References

- Chen S. and Billings A. (1996): *Neural networks for nonlinear dynamic system modeling and identification*. – Int. J. Control, vol.56, No.2, pp.319-346.
- Giergiel J., Hendzel Z. and Żylski W. (2002): *Modelling and Control of Wheeled Mobile Robots*. – Warsaw: PWN.
- Gierlak P. (2012): *Hybrid Position/Force Control of the SCORBOT-ER 4pc Manipulator with Neural Compensation of Nonlinearities*. – Springer-Verlag Berlin Heidelberg, In: L. Rutkowski *et al.* (Eds.): ICAISC 2012, Part II, LNCS 7268, pp.433–441.
- Gierlak P. (2013): *Application of adaptive hybrid position - force control of manipulator in robotised machining*. – Modelowanie Inżynierskie, vol.46, No.15, pp.28–34.
- Gierlak P. (2014): *Hybrid position/force control in robotised machining*. – Solid State Phenomena, Trans. Tech. Publications, vol.210, pp.192-199.
- Hendzel Z. and Muszyńska M. (2012): *Adaptive fuzzy control of a wheeled mobile robot*. – Int. J. of Applied Mechanics and Engineering, vol.17, No.3, pp.827-835.
- Hornik K., Stinchcombe M. and White H. (1989): *Multilayer feedforward networks are universal approximators*. – Neural Networks, vol.2, pp.359-366.
- Kumar N., Panwar V., Sukavanam N., Sharma S.P. and Borm J.-H. (2011): *Neural network based hybrid force/position control for robot manipulators*. – Int. J. Precis. Eng. Manuf., vol.12, No.3, pp.419-426.
- Lewis F.L., Jagannathan S. and Yesildirek A. (1999): *Control of Robot Manipulators and Nonlinear Systems*. – London: Taylor & Francis.
- Muszyńska M. (2012): *Neuro-fuzzy control of a wheeled mobile robot*. – Doctoral Dissertation.

Received: May 12, 2014

Revised: June 2, 2014