ISSN: 1231-4005
e-ISSN: 2354-0133
DOI: 10.5604/01.3001.0012.2513

RESEARCH OF AN OMNIDIRECTIONAL MECANUM-WHEELED PLATFORM WITH A FUZZY LOGIC CONTROLLER

Andrzej Typiak, Łukasz Rykała

Military University of Technology
Faculty of Mechanical Engineering, Institute of Machine Design
Gen. Witolda Urbanowicza Street 2, 00-908 Warsaw, Poland
tel.: +48 261 839388, +48 261 837306, fax: +48 261 837 366
e-mail: andrzej.typiak@wat.edu.pl, lukasz.rykala@wat.edu.pl

Abstract

This article presents the process of designing a fuzzy logic controller (FLC) to be applied in the control system for the tracking movement of an omnidirectional Mecanum-wheeled platform, where the tracking movement is defined as the execution of a specific movement of the omnidirectional Mecanum-wheeled platform along a preset path. The conventional PID controller, which is a popular choice for control systems, was replaced with a control algorithm featuring fuzzy logic elements. The findings from numerical testing of the control system with the applied FLC were compared with the results of numerical testing using a variant of a conventional PID controller. The comparison led to a feasibility study of the FLC for the kinematic control of the omnidirectional Mecanum-wheeled platform.

The article compares the results of applying an FLC and a variant of a conventional PID controller for the tracking control of an omnidirectional-wheeled platform with mecanum wheels. It was assumed in this work that kinematic equations would be sufficient in this case to plan the trajectory (path) of a characteristic point for the omnidirectional-wheeled platform.

Keywords: *FLC, PID controller, DC motor, kinematics, wheeled platforms, mecanum wheels, fuzzy logic*

1. Introduction

The automatic control systems, which are often applied in engineering still, feature traditional systems based on PID controllers or their variants, namely P, PI and PD controllers. These feature algorithms, which can provide satisfactory results in most conventional control applications. However, the needs of applications characterized by complex mathematical models may prove beyond the capability of conventional controllers to provide the required quality of control, forcing their operators to continuously find and implement sufficient set point parameters [7-9].

An approach to motion control based on fuzzy set theory, however, facilitates the formulation of control algorithms based on rules of logic, the inputs of which involve the operator's (an expert, if you will) knowledge, describing complex systems with variables formulated in an intuitive and qualitative (verbal) manner [7-9].

The article compares the results of applying an FLC and a variant of a conventional PID controller for the tracking control of an omnidirectional-wheeled platform with mecanum wheels. It was assumed in this work that kinematic equations would be sufficient in this case to plan the trajectory (path) of a characteristic point for the omnidirectional-wheeled platform.

2. Kinematics of the omnidirectional mecanum-wheeled platform

The platform comprised a chassis frame, four mecanum wheels and four permanent magnet DC motors (Fig. 1). The mecanum wheels, which featured identically oriented rollers, were installed at the locations shown in Fig. 1.

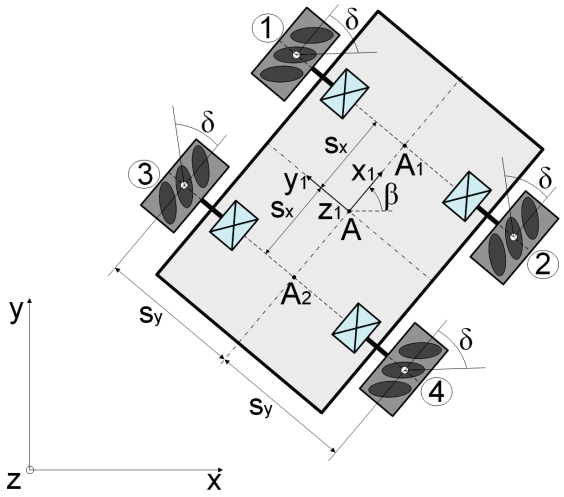

Fig. 1. Numerical model of the omnidirectional mecanum-wheeled platform

The kinematics of the platform, positioned within an immobile coordinate system, was described with the following equations:

$$-\dot{x}_A(\cos\beta + \sin\beta) + \dot{y}_A(\cos\beta - \sin\beta) + \dot{\beta}(s_x + s_y) + R_t\dot{\varphi}_1 = 0, \quad (1)$$

$$\dot{x}_A(\cos\beta - \sin\beta) + \dot{y}_A(\cos\beta + \sin\beta) + \dot{\beta}(s_x + s_y) - R_t\dot{\varphi}_2 = 0, \quad (2)$$

$$\dot{x}_A(\cos\beta - \sin\beta) + \dot{y}_A(\cos\beta + \sin\beta) - \dot{\beta}(s_x + s_y) - R_t\dot{\varphi}_3 = 0, \quad (3)$$

$$-\dot{x}_A(\cos\beta + \sin\beta) + \dot{y}_A(\cos\beta - \sin\beta) - \dot{\beta}(s_x + s_y) + R_t\dot{\varphi}_4 = 0, \quad (4)$$

with: $\dot{x}_A, \dot{y}_A$ – projections for the velocity of a specific characteristic point of the omnidirectional mecanum-wheeled platform, i.e. point A on the x-y axis of the coordinate system, $\dot{\beta}, \beta$ – angular parameters of the chassis frame, $\dot{\varphi}_i$ (i = 1, 2, 3, 4) – angular velocities of Mecanum wheel i, R_t – radius of the mecanum wheels [2, 3].

To solve the simple kinematic problem, the platform was described with coordinate system $x_1y_1z_1$ fixed to the chassis of the platform. One of used methods is to apply the Moore-Penrose theorem of inverse square matrices to the Jacobian point J present in the kinematic relations discussed here [2, 10].

Ultimately, the solution to the simple kinematic problem was presented as the following relationships:

$$v_{Ax_1} = \left(\frac{R_t}{4}\right)[\dot{\varphi}_1 + \dot{\varphi}_2 + \dot{\varphi}_3 + \dot{\varphi}_4], \quad (5)$$

$$v_{Ay_1} = \left(\frac{R_t}{4}\right)[-\dot{\varphi}_1 + \dot{\varphi}_2 + \dot{\varphi}_3 - \dot{\varphi}_4], \quad (6)$$

$$\dot{\beta} = \left(\frac{R_t}{4[s_x+s_y]}\right)[-\dot{\varphi}_1 + \dot{\varphi}_2 - \dot{\varphi}_3 + \dot{\varphi}_4]. \quad (7)$$

Equations (5-7) describe the solution to the simple kinematic problem, which served here as a mathematical model for the kinematic system [2].

3. Permanent-magnet DC motor

The four actuators for the omnidirectional mecanum-wheeled platform were permanent-magnet DC motors. Given that the angular velocity of the DC motor's rotor ($\dot{\varphi}_r$) was an output quantity and the supply voltage (U_z) was an input value, the transmittance of the DC motor could be expressed as a second-order element.

$$G(s) = \frac{\Omega_r(s)}{U_z(s)} = \frac{K}{T_eT_ms^2 + T_ms + 1}, \quad (8)$$

with: T_e – electric time constant, T_m – mechanical time constant, K – velocity gain factor [6].

The electric time constant had values much lower than the mechanical time constant for the same kinematic system, given the very low inductance rotors and very low resistance windings found in most DC motors.

4. PID controller

To synthesize a PID controller, the following law of conventional control was assumed:

$$u_{PID}(t) = k_P e_{PID}(t) + k_I \int_0^t e_{PID}(t)dt + k_D \frac{de_{PID}(t)}{dt}, \tag{9}$$

with: k_P, k_I, k_D – gains of the proportional element, the integrating element and the differential element, respectively [5, 6].

The PID controller bias was described as follows:

$$e_{PID}(t) = y_d(t) - y(t), \tag{10}$$

with: $y_d(t)$ – set point, $y(t)$ – response of the system [5, 6].

For this example, a PI controller was used, as a characteristic variant of the PID controller.

5. FLC

The conventional PID controller (or PI controller here) was replaced with a control algorithm featuring fuzzy logic (FL) elements. Fuzzy logic control is one of the main applications of fuzzy set theory. An FLC is characterized by a control law, which can be described with a rule-based knowledge base and a fuzzy control mechanism. The FLC featured the following elements: a fuzzification block, an inference block and a defuzzification block. In the fuzzification process, the data at the FLC input were fuzzified. Next, their degree of membership with a specific fuzzy set was determined. The inference process calculated the resultant membership function from a set of rules and fuzzed input data. Finally, the defuzzification process calculated the acute FLC output value from the resultant membership function [1, 4, 8, 11].

The inputs of the Mamdani PI FLC were bias e and change of bias, Δe (here: bias integral), the output of the FLC model was the control signal u (Fig. 2).

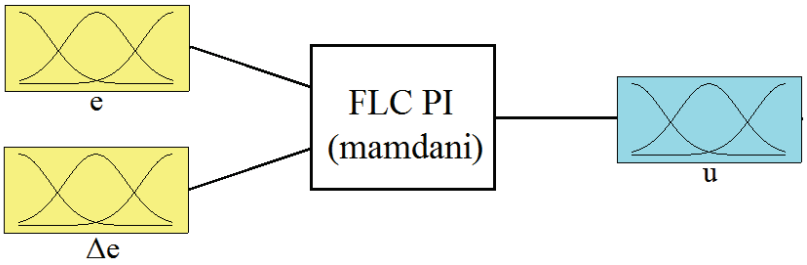

Fig. 2. General diagram of the FLC model

The FLC bias was described as follows:

$$e(t) = y(t) - y_d(t), \tag{11}$$

with: $y_d(t)$ – set point, $y(t)$ – response of the system [4, 8].

The input and output signals were divided into three fuzzy Gaussian sets: NE (negative), ZE (zero) and PO (positive), as shown in Figs. 3a, 3b and 4a, respectively. The inputs and outputs were normalized to the interval [-1, 1]. The resulting factors k_e, k_i and k_u were the parameters applied to tune the FLC at a later stage of the study.

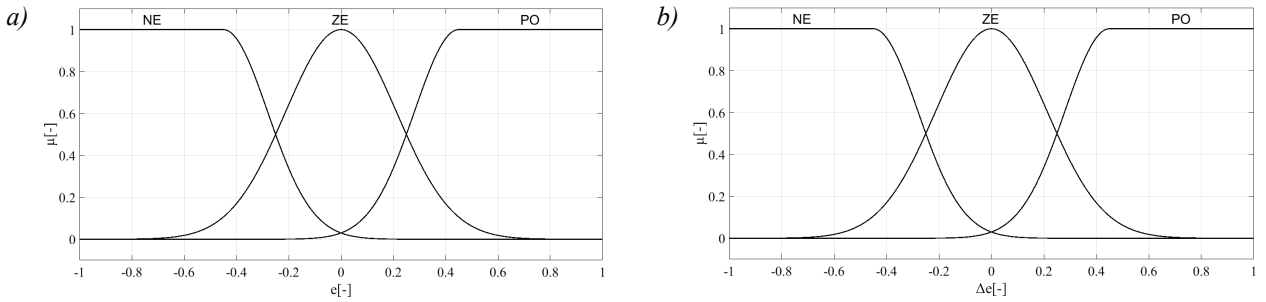

Fig. 3 Trends in the fuzzy sets of memberships of the following signals: a) bias; b) change of bias

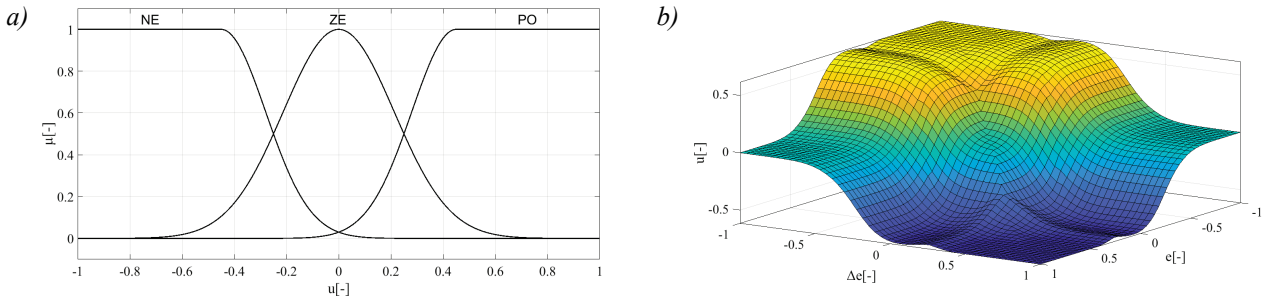

Fig. 4. Trends in the fuzzy sets of control signal membership (a). Surface of the fuzzy model (b)

The knowledge base of the FLC was rule-based, as shown in Tab. 1 and comprised 9 rules.

The small number of rules and the application of the membership functions described above gave the surface of the fuzzy model (Fig. 4b) a smooth and transparent spatial form.

Tab. 1. Rule-based knowledge base of the FLC

Δe \ e	NE	ZE	PO
NE	ZE	NE	NE
ZE	PO	ZE	NE
PO	PO	PO	ZE

6. Numerical simulations

A simplified diagram of the numerical simulations is shown in Fig. 5. The execution of two motion paths was evaluated for the characteristic point of the omnidirectional Mecanum-wheeled platform with a constant configuration of the chassis frame. The first motion path described a square with sides of length 20 m (Fig. 6a), the second motion path described a circle with a radius of 5 m and a geometric centre at point (0, 5) (Fig. 6b).

Fig. 5. Diagram of the numerical simulation

It was assumed that each permanent-magnet DC motor had a separate controller, which corresponded to an implementation of a decentralized control system for the platform.

The square motion path was generated with linear intervals (Fig. 7a). The circular motion path was generated with continuous, parametric equations of a circle (Fig. 7b).

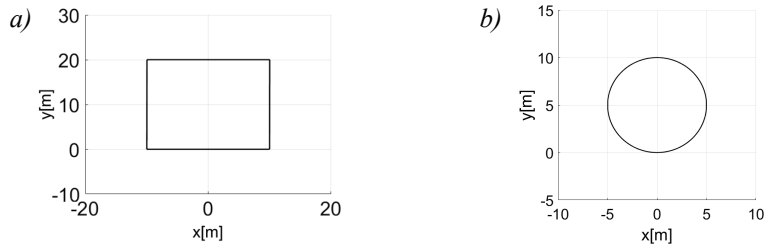

Fig. 6. Charts of the motion paths of the characteristic point of the omnidirectional Mecanum-wheeled platform: (a) square motion path; b) circular motion path

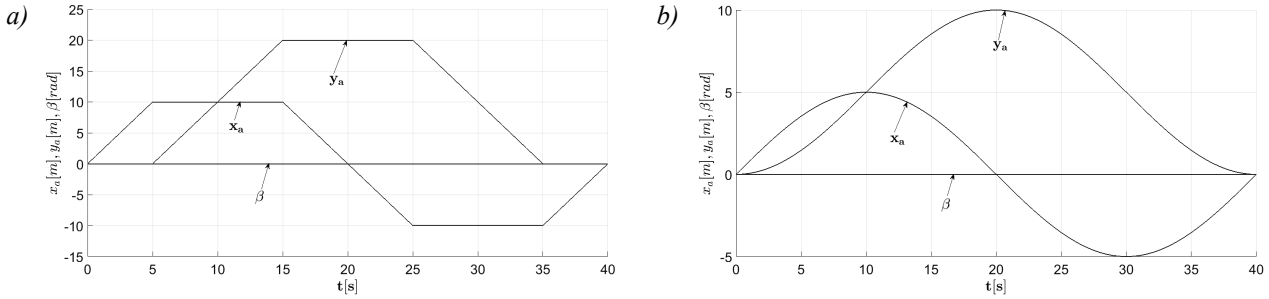

Fig. 7. Trends in the preset kinematic parameters of the characteristic point of the omnidirectional Mecanum-wheeled platform for: a) square motion path; b) circular motion path

All numerical simulations considered here were carried out in the Matlab/Simulink software, method ode4 and discretisation step h = 0.001. The following geometrical parameters of the system were assumed in the numerical simulations: s_x = 0.5 m, s_y = 0.75 m, $\delta = \pi/4$ rad, R_t = 0.1 m. The parameters of the permanent-magnet DC motors are shown cumulatively in Table 2.

Tab. 2. Identification of the parameters of the i-th permanent-magnet DC motor (i = 1, 2, 3, 4)

i / K,T	1	2	3	4
K_i	108.15	109.06	104.27	107.13
T_{mi}	0.141	0.145	0.136	0.148
T_{ei}	0.001	0.001	0.001	0.001

Another assumption considered was: throughout each numerical simulation, each of the permanent-magnet DC motors was affected by a variable interference, which was proportional to the angular velocity of the rotor.

To compare the effects of operation of the FLC and the PI controller, the following quality index was defined:

$$J_q = \int_{t_k}^{t_0} e(t)^2 dt, \tag{12}$$

with: t_0, t_k – start and end time of the numerical simulation, $e(t)$ – control signal bias.

6.1. Numerical simulation 1

Numerical simulation 1 represented the operation of the PI controller with the square motion path of characteristic point A. The proportional and integral element gains for the controller were chosen by experimental trial and error. The results for numerical simulation 1 are shown in Figs. 8-10.

The distinctive and sudden spikes in the values shown in Figs. 8a, 8b, 9a and 10 at 5, 15, 25 and 35 s were caused by sudden changes in the preset kinematic parameters of velocity: $\dot{x}_A, \dot{y}_A$. The quality index calculated for numerical simulation 1 from relation (12) was $J_q = 9.859 \cdot 10^4$ rad^2/s^2.

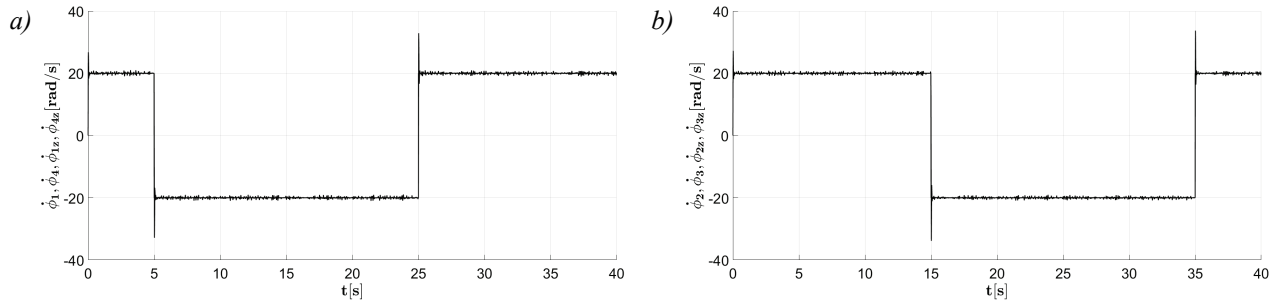

Fig. 8. Angular velocity trends: a) preset angular velocities $(\dot{\varphi}_1, \dot{\varphi}_4)$ and actual angular velocities $(\dot{\varphi}_{1z}, \dot{\varphi}_{4z})$ for wheels 1 and 4; b) preset angular velocities $(\dot{\varphi}_2, \dot{\varphi}_3)$ and actual angular velocities $(\dot{\varphi}_{2z}, \dot{\varphi}_{3z})$ for wheels 2 and 3 in numerical simulation 1

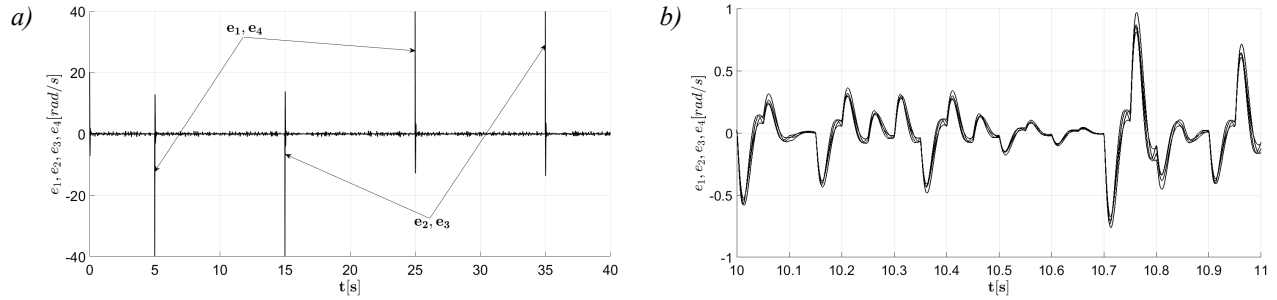

Fig. 9. Bias signal trends: $e_1(t)$, $e_2(t)$, $e_3(t)$, $e_4(t)$: a) during the numerical simulation; b) at time interval t = (10, 11) s of numerical simulation 1

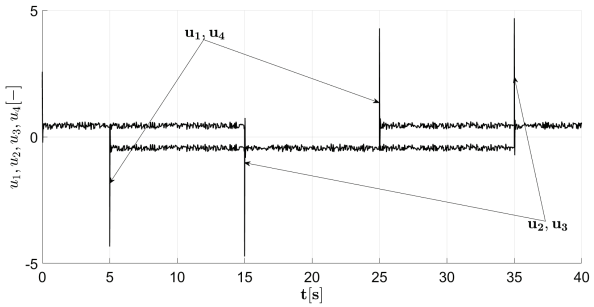

Fig. 10. Chart of the control signals: $u_1(t)$, $u_2(t)$, $u_3(t)$, $u_4(t)$ in numerical simulation 1

6.2. Numerical simulation 2

Numerical simulation 2 represented the operation of the PI FLC with the square motion path of characteristic point A. The following gain factors for the individual controller signals were assumed as resulting from the normalization: $k_e = 1/80$, $k_i = 1$, $k_u = 200$. The results for numerical simulation 2 are shown in Figs. 11-13.

Not unlike numerical simulation 1, the distinctive and sudden spikes in the values shown in Figs. 11a, 11b, 12a and 13 at 5, 15, 25 and 35 s were also caused by sudden changes in the preset kinematic parameters of velocity: $\dot{x}_A, \dot{y}_A$. The quality factor J_q in this numerical simulation was $J_q = 1.981 \cdot 10^4$ rad^2/s^2.

6.3. Numerical simulation 3

Numerical simulation 3 represented the operation of the PI controller with the circular motion path of characteristic point A. The proportional and integral element gains for the controller were chosen by experimental trial and error. The results for numerical simulation 3 are shown in Figs. 14-16.

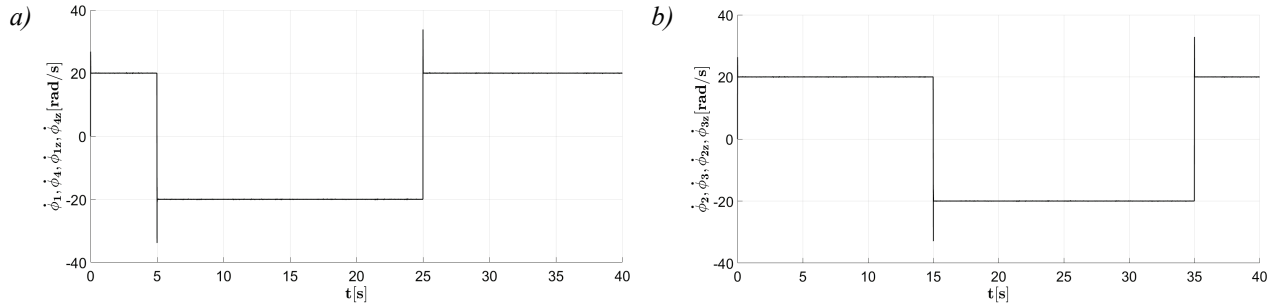

Fig. 11. Angular velocity trends: a) preset angular velocities $(\dot{\varphi}_1, \dot{\varphi}_4)$ and actual angular velocities $(\dot{\varphi}_{1z}, \dot{\varphi}_{4z})$ for wheels 1 and 4; b) preset angular velocities $(\dot{\varphi}_2, \dot{\varphi}_3)$ and actual angular velocities $(\dot{\varphi}_{2z}, \dot{\varphi}_{3z})$ for wheels 2 and 3 in numerical simulation 2

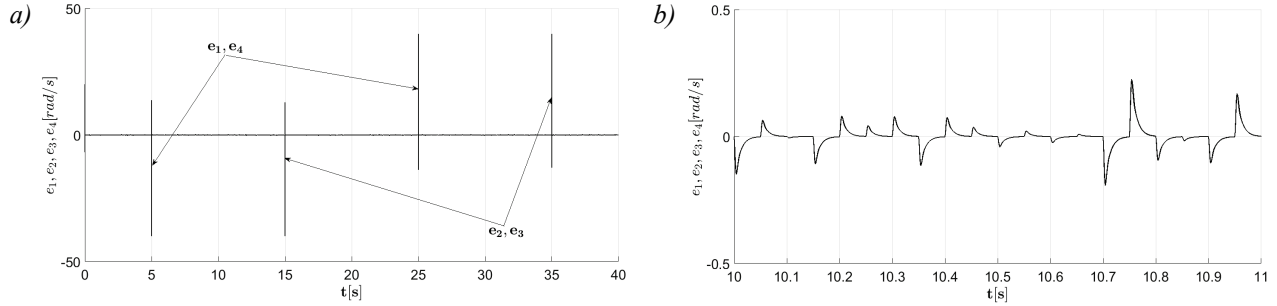

Fig. 12. Bias signal trends: $e_1(t)$, $e_2(t)$, $e_3(t)$, $e_4(t)$: a) during the numerical simulation; b) at time interval t = (10, 11) s of numerical simulation 2

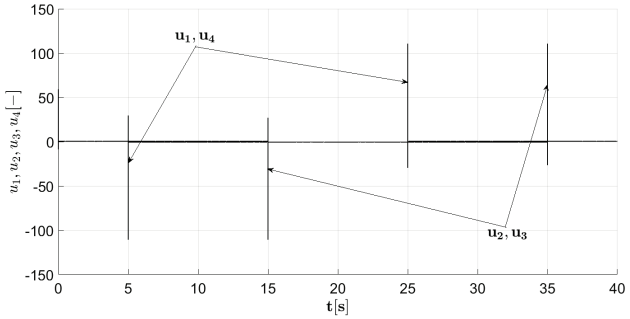

Fig. 13. Chart of the control signals: $u_1(t)$, $u_2(t)$, $u_3(t)$, $u_4(t)$ in numerical simulation 2

Figures 14a, 14b, 15a, 15b and 16 show the distinctive effect of interference on individual signals, most prominently control signal u. The quality index calculated for numerical simulation 3 from relation (12) was $J_q = 3.137 \cdot 10^3$ rad^2/s^2.

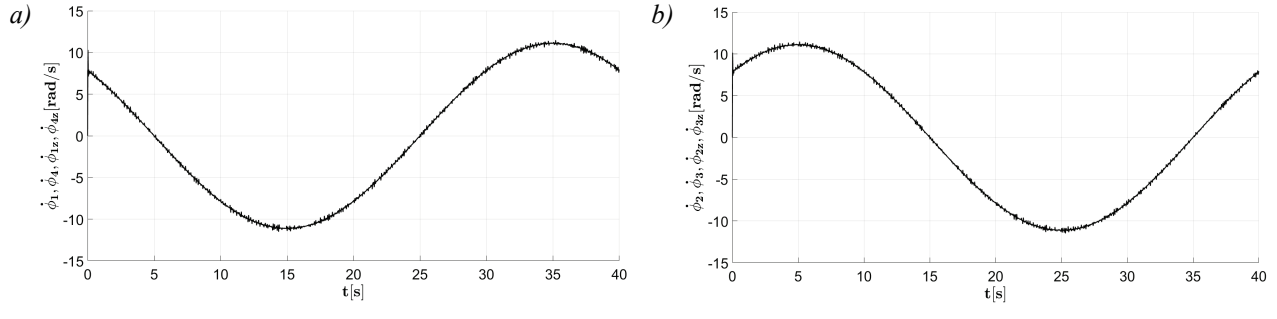

Fig. 14. Angular velocity trends: a) preset angular velocities (φ_1, φ_4) and actual angular velocities $(\varphi_{1z}, \varphi_{4z})$ for wheels 1 and 4; b) preset angular velocities (φ_2, φ_3) and actual angular velocities $(\varphi_{2z}, \varphi_{3z})$ for wheels 2 and 3 in numerical simulation 3

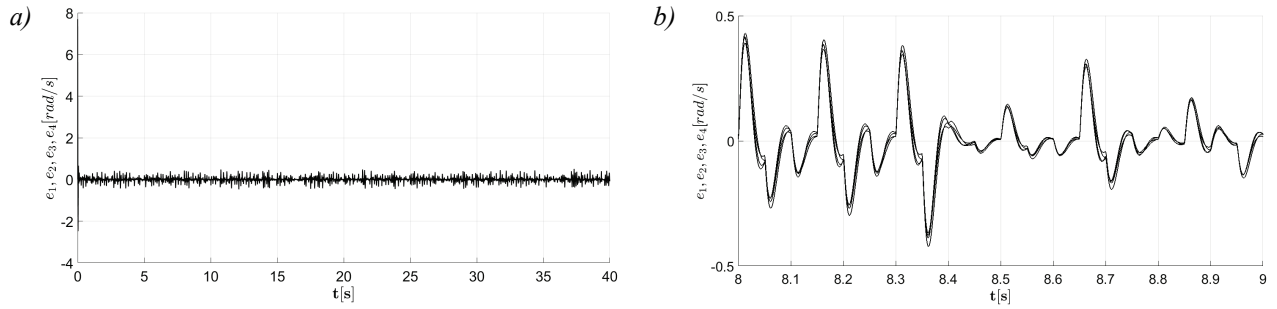

Fig. 15. Bias signal trends: $e_1(t)$, $e_2(t)$, $e_3(t)$, $e_4(t)$: a) during the numerical simulation; b) at time interval t = (10, 11) s of numerical simulation 3

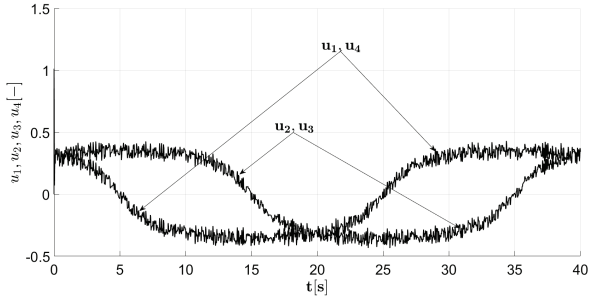

Fig. 16. Chart of the control signals: $u_1(t)$, $u_2(t)$, $u_3(t)$, $u_4(t)$ in numerical simulation 3

6.4. Numerical simulation 4

Numerical simulation 4 represented the operation of the PI FLC with the circular motion path of characteristic point A. The gain factors for the individual controller signals were identical to those in numerical simulation 2. The results for numerical simulation 3 are shown in Figs. 17-19.

Figures 17a, 17b, 18a, 18b and 19 show that the interference was much less effective on the individual signals than in numerical simulation 3. The quality index calculated from relation (12) for numerical situation 3 was $J_q = 202.713$ rad^2/s^2.

The values of quality index J_q determined in the numerical simulations and the comparison of the charts shown in Figs. 9b and 12b to those in Figs. 15b and 18b, show that the PI FLC was much better at executing tracking control of the omnidirectional mecanum-wheeled platform than its conventional PI counterpart.

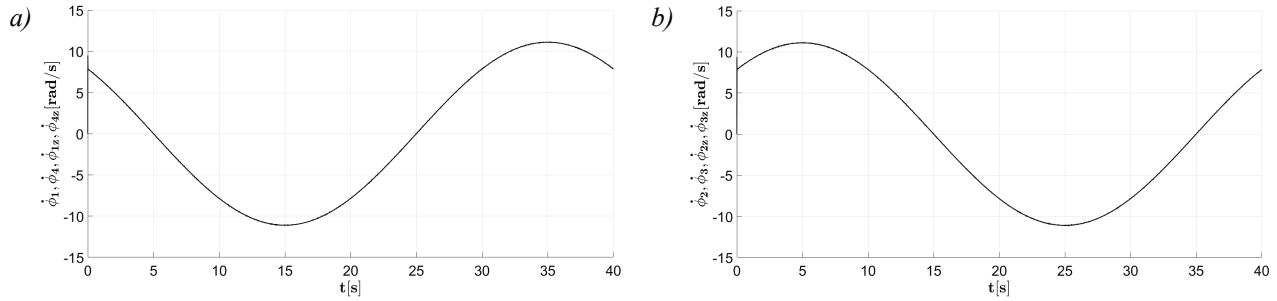

Fig. 17. Angular velocity trends: a) preset angular velocities $(\dot{\varphi}_1, \dot{\varphi}_4)$ and actual angular velocities $(\dot{\varphi}_{1z}, \dot{\varphi}_{4z})$ for wheels 1 and 4; b) preset angular velocities $(\dot{\varphi}_2, \dot{\varphi}_3)$ and actual angular velocities $(\dot{\varphi}_{2z}, \dot{\varphi}_{3z})$ for wheels 2 and 3 in numerical simulation 4

7. Conclusions

The procedure for tuning an FLC is usually much more difficult than for tuning a conventional PID controller. This is due to the high degree of flexibility of the FLC, described by the large

number of parameters, which define the fuzzy sets, the inference mechanism and the defuzzification mechanism, etc. However, once the FLC was tuned properly and supplied with a small number of fuzzy sets and a small rule base, it provided much less bias than the PI controller, which required tuning for every numerical simulation, as clearly evidenced by the individual quality index values.

The application of fuzzy logic in automatic controllers facilitates a different kind of object motion control than achieved with conventional methods. Fuzzy algorithms are much more robust against changes in the preset system parameters and improve the effectiveness of control in comparison to traditional PID controllers.

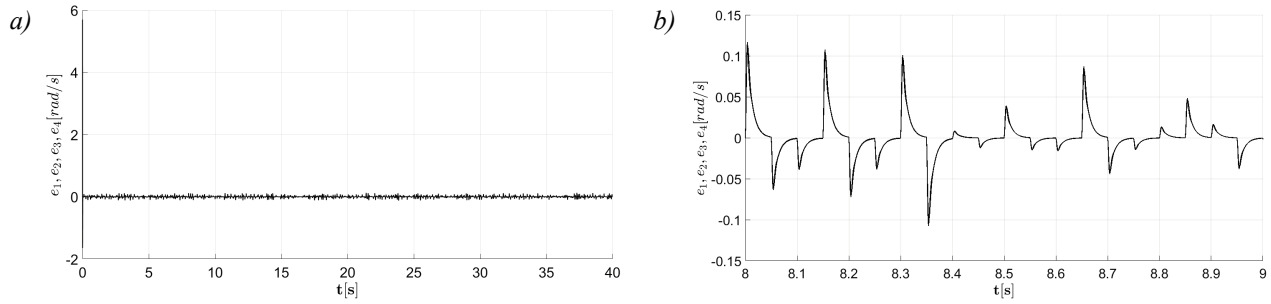

Fig. 18. Bias signal trends: $e_1(t)$, $e_2(t)$, $e_3(t)$, $e_4(t)$: a) during the numerical simulation; b) at time interval t = (10, 11) s of numerical simulation 4

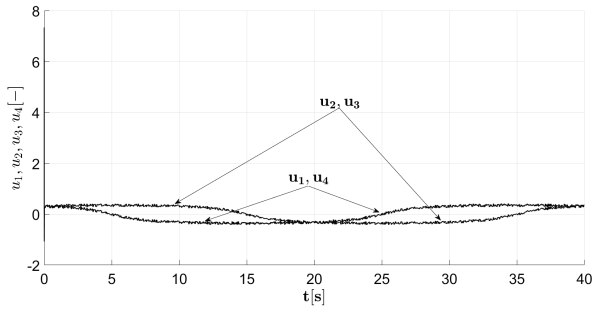

Fig. 19. Chart of the control signals: $u_1(t)$, $u_2(t)$, $u_3(t)$, $u_4(t)$ in numerical simulation 4

References

[1] Giergiel, J., Kurc, K., *Identification of the mathematical model of an inspection mobile robot with fuzzy logic systems and neural networks*, Journal of Theoretical and Applied Mechanics, 49 (1), pp. 209-225, 2011.

[2] Hendzel, Z., Rykała, Ł., *Modelling of dynamics of a wheeled mobile robot with mecanum wheels with the use of Lagrange equations of the second kind*, International Journal of Applied Mechanics and Engineering, 22 (1), pp. 81-99, 2017.

[3] Hendzel Z., Rykała, Ł., *Opis kinematyki mobilnego robota kołowego z kołami typu mecanum*, Modelowanie Inżynierskie, 26 (57), pp. 5-12, 2015.

[4] Hendzel Z., Szuster, M., Gierlak, P., *Sieci neuronowe i systemy rozmyte*, Oficyna Wydawnicza Politechniki Rzeszowskiej, pp. 151-167, Rzeszow 2010.

[5] Kowal, J., *Podstawy automatyki. t. I*, AGH Uczelniane Wydawnictwa Naukowo-Dydaktyczne, pp. 223-229, Krakow 2003.

[6] Kozłowski, K., Dutkiewicz, P., Wróblewski, W., *Modelowanie i sterowanie robotów*, Wydawnictwo Naukowe PWN, pp. 271-285, Warszawa 2012.

[7] Mrozek, B., *Projektowanie regulatorów rozmytych w środowisku MATLAB-Simulink*, Pomiary Automatyka Robotyka, 11, pp. 5-12, 2006.

[8] Piegat, A., *Modelowanie i sterowanie rozmyte*, Akademicka Oficyna Wydawnicza „Exit", pp. 165-285, Warszawa 1999.

[9] Rojek, R., Bartecki, K., *Metody sztucznej inteligencji w zastosowaniach automatyki*, Pomiary Automatyka Kontrola, 52, pp. 29-34, 2006.
[10] Typiak, A., Łopatka, M. J., Rykała, Ł., Kijek, M., *Dynamics of omnidirectional unmanned rescue vehicle with mecanum wheels*, AIP Conference Proceedings, Vol. 1922 (1), p. 120005, AIP Publishing, 2018.
[11] Yager, R., Filev, D., *Podstawy modelowania i sterowania rozmytego*, Wydawnictwa Naukowo-Techniczne, pp. 270-313, Warszawa 1995.

Manuscript received 20 December 2017; approved for printing 30 March 2018