INFLUENCE OF CONTROL ALGORITHM ON DRIVE CONSUMPTION OF WHEELED MOBILE ROBOT

Dušan Marko*, Vladimír Madola, Dušan Hrubý, Martin Olejár

Department of Electrical Engineering, Automation and Informatics, Faculty of Engineering, Slovak University of Agriculture in Nitra, Tr. Andreja Hlinku 2, 949 01, Nitra, Slovakia

Abstract

In field of autonomous mobile robotics, the quality of control is a crucial aspect. Besides the precision of control, the energy consumption for motion is becoming an increasingly demanding characteristic of a controller due to the increasing costs of fossil fuels and electricity. We created a simulation model of a differential drive mobile robot motion in a vineyard row including robot dynamics for evaluating motion consumption and implemented commonly used PID, fuzzy, and LQ control algorithms. Their task was to navigate the robot through the middle of the vineyard row section by measuring distances from trellises on both sides of the robot. The comparison was carried out using MATLAB software and the best results both in terms of power consumption and control accuracy were achieved by LQI controller. The designed model for navigating the robot through the centre of the vineyard row and optimized controllers will be implemented in a real robot and tested in a real scenario.

Keywords: *robot control, PID control, fuzzy control, optimal control, simulation, energy consumption*

1. INTRODUCTION

Nowadays, with continuously reducing fossil fuels supplies and increasing electricity prices as the population and demand for it increases, it is becoming necessary to consider, in addition to accuracy and speed another condition for choosing the control algorithm which is the energy consumption of control. Paper [1] presented a proposal on how to reduce the energy consumption of components such as processors and sensors, but depending on the size of the robot, the participation of components consumption in total consumption decreases. Therefore, the main objective is to examine the consumption of actuators. Work [2] compared the impact of a mixed dynamic model, including both motor and robot chassis dynamics, on driving characteristics and consumption to the model including only a motor dynamics. This resulted in differences in consumption and more realistic behaviour of the mixed model. Also, the work [3] proved that an accurate power model allows the estimation of energy consumption along different trajectories with over 90 % accuracy. Another work [4] compared both the simulated and experimentally designed controller with its optimized version using the H∞ method on the omnidirectional robot model. The optimized controller achieved not only smaller control errors, but also lower power consumption of the robot. In [5] authors were engaged in optimizing the speed profiles of the wheeled robot for the lowest possible consumption when moving on a flat track. They found that the lowest consumption was achieved with a parabolic speed profile. A direct comparison of mobile robot motion controllers was done by [6]. Controllers with low energy consumption were lacking accuracy due to their smooth control actions and vice versa accurate controllers showed to have higher consumption. A recent work [7] created a prototype of an autonomous robot for precision agriculture potentially capable of various tasks. This type of robot equipped with complex location sensors capable of real-time trajectory planning is becoming a standard in precision agriculture and is a great platform for optimizing control with respect to reducing consumption. We set our goal to compare power consumption and control precision with three commonly used control algorithms in a scenario of differential drive mobile robot movement in a vineyard row. The desired trajectory of the robot will be the middle of the row. Simulations will be done with a mathematical model including both dynamics and kinematics.

2. MATERIALS AND METHODS

2.1. Differential drive mobile robot

The robot with differential chassis represents a robot with two independently driven wheels located on a common axis perpendicular to the direction of rotation of the wheels and often the third wheel which is added due to the stability of the robot [10]. For purpose of robot driving simulations using various control algorithms and examining their quality, including the consumption of the robot, we decided to use a state-space mathematical model published by [11]. This model includes, in addition to the robot's kinematics, a dynamic part describing the dependence of electric currents on the load and wheel velocities from the supplied motor voltages, respecting the dynamics of the engine, construction, geometry and other parameters of the chassis and engines. Parameters for our robot model have been chosen to approximate a real mobile robot described in [9] and are stated in Table 1 where K_t is motor torque constant, K is electric constant, R is motor winding resistance, L is motor inductance, J is rotor moment of inertia, k_r is coefficient of rotation resistance; r is wheel radius, m is robot mass, l is half of track width, J_T is moment of inertia in centre of gravity, k_v is resistance coefficient against linear motion, k_ω is resistance coefficient against rotary motion and p_G is gearbox gear ratio.

K_t [N·m·A^{-1}]	$9 \cdot 10^{-2}$	**m [kg]**	3
K [V·min^{-1}]	$4{,}4 \cdot 10^{-2}$	**l [m]**	0,25
R [Ω]	8	**l_T [m]**	0,05
L [mH]	2	**J_T [kg·m^{-2}]**	0,056
J [kg·m^{-2}]	3.10^{-5}	**k_v [kg·s^{-1}]**	0,1
k_r [kg·m^2·s^{-1}]	$6{,}3 \cdot 10^{-2}$	**k_ω [kg·m^2·s^{-1}]**	1,3
r [m]	0,05	**p_G [-]**	10

Table 1. Parameters of differential drive mobile robot mathematical model

2.2. Control algorithms

As for control algorithms, we have chosen three of the most used - PID (proportional-integral-derivative), fuzzy, and LQI (linear-quadratic-integral) controllers. For most accurate simulation results, we will use a discrete form of PID controller - PSD controller, which is necessary for simulating computer-controlled robot at a certain sampling rate. Equation (1) represents a parallel form of PSD controller.

$$u(k) = K_p e(kT) + K_i T \sum_{k=0}^{n} e(kT) + K_d \frac{1}{T}[e(kT) - e(k_{-1}T)] \quad (1)$$

where $u(k)$ is control action [V],

$e(kT)$ is control error [-],

T is sampling time [s],

K_p is proportional gain [-],

K_i – integral gain [s^{-1}],

K_d – derivative gain [s],

k – sample number [-].

Fuzzy controller is based on fuzzy logic which provides a solution for linguistic terms such as "much", "medium", "little". It enables the realization of processes of human thinking of decision-making in algorithmic form and is based on a qualitative description of the dynamic system. Its main advantage is

that there is no need to know the mathematical description of the system for its implementation [8]. The fuzzy sets theory is based on a membership degree of element to the set. The value of each element in set is between 0 and 1 [9]. In Table 2 are chosen linguistic variables for our fuzzy controller. For shorter computational time, we used Sugeno fuzzy system with constant output membership functions in brackets in Table 2. In total it represents 25 combinations for inference rules.

robot velocity error e_v [m·s^{-1}]	**robot position error θ_e [rad]**	**multiple of motor supply voltage u_P, u_L [-]**
very slow	too left	slow down more (0)
slow	left	slow down (0,25)
desired	desired	optimal (0,5)
fast	right	speed up (0,75)
very fast	too right	speed up more (1)

Table 2. Linguistic variables for input and output quantities v_R, θ_e, u_P, u_L

When dealing with consumption and control efficiency, LQ controller as a representative of optimal control should be considered. A detailed description of LQ controller and the effect of its weighting matrices Q and R was presented by [12]. We opted for a variation of LQ controller with added integration component that improves the response of the controller when additional external forces act on the system – LQI controller. Using the integration component, the Q and K matrices are extended by the number of output vector states and the impact of the integration component can be penalised by adjusting the related elements of the Q matrix. A feedback law for LQI controller is stated in (2).

$$u = -K \begin{bmatrix} x \\ x_i \end{bmatrix} \tag{2}$$

where u is control variable [V],

K is optimal gain matrix,

x – state vector,

x_i - integrator output.

2.3. Navigation algorithm

Inputs to controllers represent measured distances from a vineyard wall on each side of the robot and the linear velocity of the robot. The basic equation for robot position error e is in (3).

$$e = d_R - d_L \tag{3}$$

where e is robot position error [m],

d_R is distance from right vineyard row wall [m],

d_L is distance from right vineyard row wall [m].

Because we want not only the position error to be minimized, but also control the robot orientation, we used a similar technique to one presented by [12] to calculate desired robot orientation angle θ_d in (4).

$$\theta_d = arctan\left(\frac{-e}{\Delta s}\right) = arctan\left(\frac{-e}{v_R T}\right) \tag{4}$$

where θ_d is desired robot orientation angle [rad],

T is sample time [s],

Δs is distance travelled over time T [m].

The second input to a controller is velocity error e_v in (5). The desired velocity v_w was in our simulations constant.

$$e_v = v_w - v_R \tag{5}$$

where e_v is robot velocity error [m·s^{-1}],

v_w is desired speed [m·s^{-1}],

v_R – actual robot speed [m·s^{-1}].

Finally, robot orientation angle error θ_e entering the controller is calculated by (6).

$$\theta_e = \theta_d - \theta_R \tag{6}$$

where θ_R is actual robot orientation angle [rad].

For the LQI controller, as its integrational matrix x_i requires to have same quantities as the robot's state space model output vector y, it was required to calculate robot's angular velocity error ω_e as stated in (7).

$$\omega_e = \omega_d - \omega_R = \frac{\Delta\theta_d}{T} - \omega_R = \frac{\theta_d - \theta_{d-1}}{T} - \omega_R \tag{7}$$

where ω_d is desired angular velocity [rad·s^{-1}],

ω_R is actual angular velocity of robot [rad·s^{-1}],

θ_{d-1} is previous desired orientation angle of robot [rad].

2.4. Control quality criterions

Absolute control error area I_{AE} (8) gives basic information on controller's ability to track the desired trajectory. This criterion is applicable for both oscillating and non-oscillating control process.

$$I_{AE} = \int_0^\infty |e(t)|dt \qquad [\text{m}] \tag{8}$$

The average control deviation is calculated as the proportion of the absolute control area and the distance travelled by the robot according to (9).

$$e_{avg} = \frac{I_{AE}}{d} \qquad [-] \tag{9}$$

where *d is* distance travelled by robot [m].

To quantify how smooth is the robot movement, we used Smoothness of Curvature (Smoo) metric presented by [13] stated in (10).

$$Smoo = \frac{\int_0^l \left(\frac{d\kappa}{dt}\right)^2 ds}{t} \qquad [\text{m}^{-1}\text{s}^{-3}] \tag{10}$$

where κ is trajectory curvature [m^{-1}],

s – trajectory length [m],

t – simulation time [s].

Due to operation in a discrete environment, integration was approximated by trapezoidal numerical method. For trajectory curvature calculation we use open source Matlab code created by [14]. The mobile robot's own consumption W was calculated from instantaneous power P for each motor. Consumption was evaluated as the sum of all instantaneous power values as stated in (11).

$$W = \sum_0^t P(kT)T = \sum_0^t [u(kT) \cdot i(kT)]T \qquad [\text{W·s}] \tag{11}$$

where $P(kT)$ is instantaneous power at current sample k [W],

u is motor input voltage calculated by controller [V],

i is electric current [A],

T is sampling time [s].

3. RESULTS

The simulation model was created and evaluated in Matlab & Simulink software. We created all the controllers with a sampling period of T=0,1 s to simulate the numerical controller's behaviour more realistically and the desired tangential speed of the robot v_R was for all simulations 0,5 $m \cdot s^{-1}$. For a distance sensor simulation, we used results from our previous work [15] where we compared 3 different distance sensors for use of measuring distance in vineyard row. A simplified simulation flowchart of the Simulink model is in Fig. 1 with highlighted quantities calculated in each block.

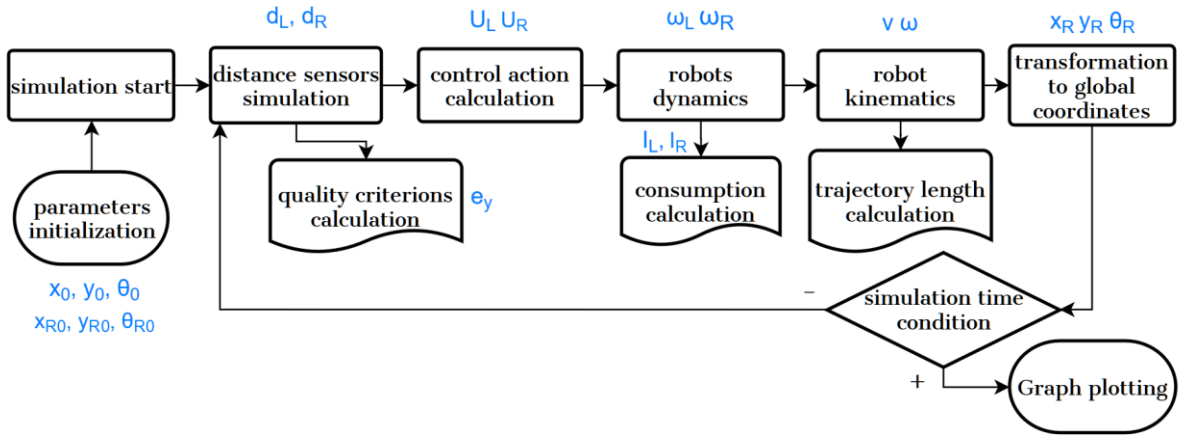


Fig. 1. Simulation program flowchart

The testing track design, distance sensor characteristics and their placement on the robot was done in the Matlab Driving Scenario Designer toolbox. All quantities required to be calculated by integrating were approximated by the trapezoidal numerical method at a given sample time to be more realistic. Since a simulated mobile robot is a MIMO (multiple-input and multiple-output) system with 2 inputs and 2 outputs, it was necessary to create 2 PSD controllers, each for one input quantity. To set the parameters of both controllers, we used the possibility of automatic tuning of the PID block in Simulink. We have tried to achieve the shortest possible time to achieve the desired values without overshooting the setpoint. The parameters obtained are in Table 3.

Constant	Velocity controller	Position controller
$\mathbf{K_P}$	2.9877	5.9199
$\mathbf{K_I}$	29.8773	0.6960
$\mathbf{K_D}$	0.0747	1.1792

Table 3. Tuned constants of PID (PSD) controller

We used the Matlab *Fuzzy Logic Designer* toolbox to design the fuzzy controller. Since the fuzzy controller allows the control of MIMO systems, it is enough to create a single controller for the overall control of the robot. From the linguistic variable in Table 2, we created triangular input membership functions shown in Fig. 2.

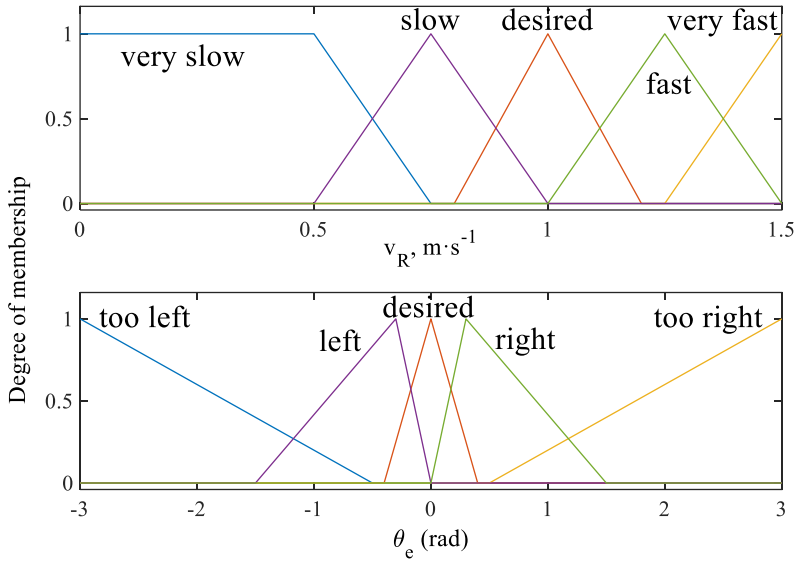


Fig. 2. Membership functions for robot velocity and robot orientation angle

The appropriate matrix of optimal gain K was determined by trial and error method by adjusting the parameters of the diagonal matrix Q. Matrix R we determined at the beginning. The controller has the best results for the parameters in (12). In our case, by increasing the value of the parameters, the weight of the associated state variables was reduced and vice versa.

$$Q = \begin{pmatrix} 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 80 & 0 \\ 0 & 0 & 0 & 0 & 0 & 20 \end{pmatrix} \quad R = \begin{pmatrix} 0.1 & 0 \\ 0 & 0.1 \end{pmatrix} \tag{12}$$

From Fig. 3 it can be observed that all the controllers were able to quickly suppress the control deviation and lead the robot relatively accurate through the centre of the track.

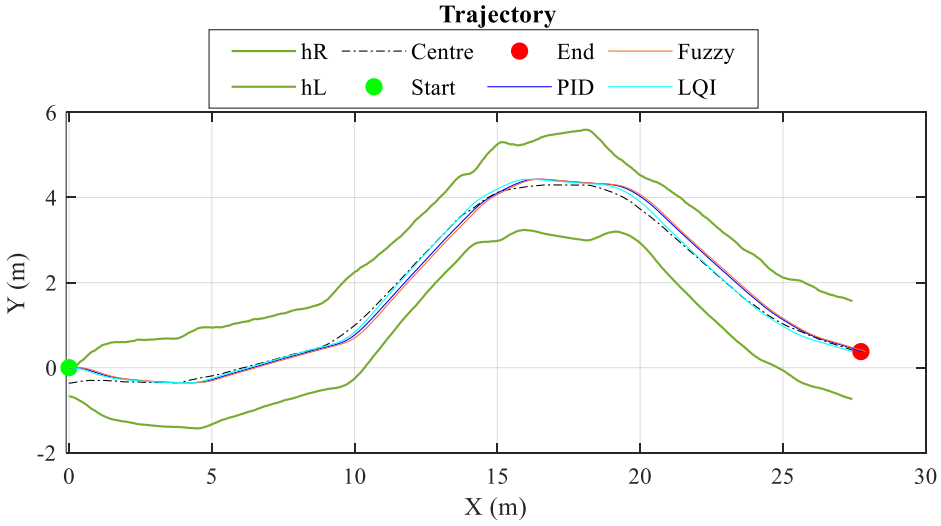


Fig. 3. Robot trajectories travelled using each control algorithm

Trajectory travelled with using LQI controller (cyan) is visually closest to the ideal line through the middle of the track what is also confirm our criterions. Noticeable difference among the controllers appeared from the courses of motor input voltages calculated by the controllers and output vector of the mobile robot's mathematical model as shown in Fig. 4. PID controller turned out to be very aggressive from the course of the input voltages and angular velocity values of the robot, even though its parameters in Table 3 were set to not cause an overshooting of setpoint. Fluctuations of input voltages also transmitted to the output vector of the robot, where especially the angular velocity ω_{PID} is not so smooth. This reflected in highest consumption among all the controllers in the results shown in Table 4. Fuzzy and LQI controllers were able to achieve much better control of the robot as is seen from input voltages and output vector. The most noticeable difference between these two controllers is in acceleration from standstill to desired velocity. Fuzzy controller accelerated faster and reach the velocity within a second, whereas LQI controller's acceleration was more gradual. Angular velocity response was in the same way because of faster acceleration of the robot with Fuzzy controller.

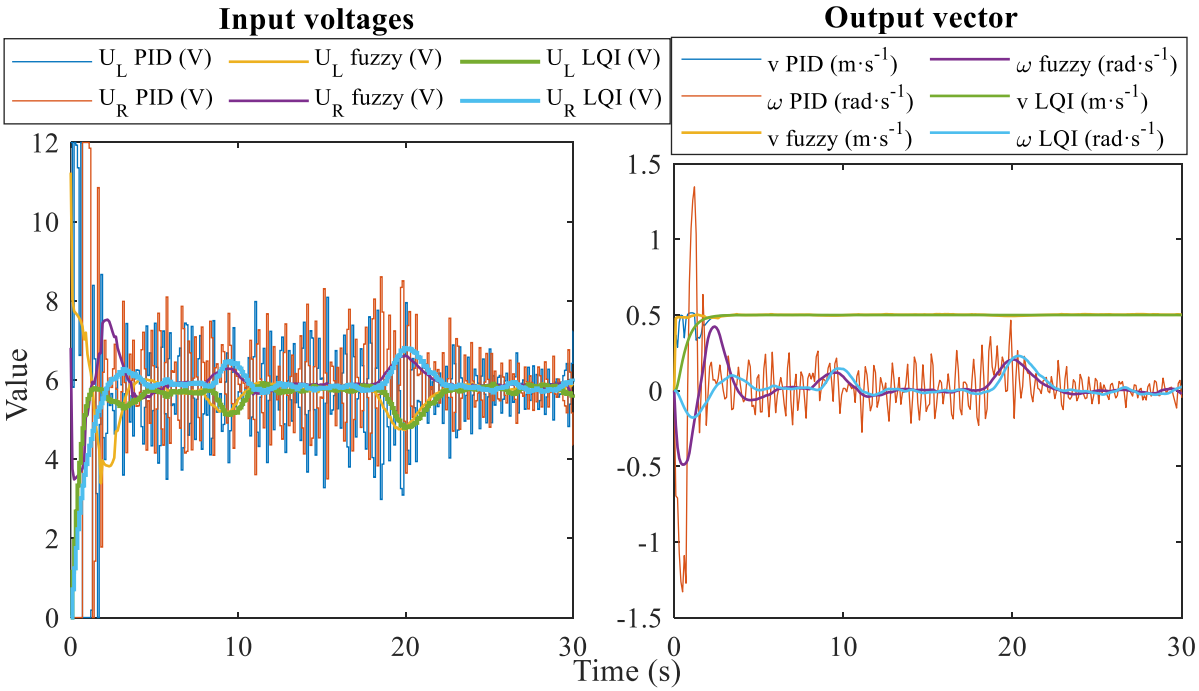


Fig. 4. Time courses of motors input voltages and output vector with each control algorithm

Controller	Travelled distance [m]	W [W·s]	Consumption difference [%]	I_{AE} [m]	Smoo [$m^{-1}s^{-3}$]	e_{avg} [-]
PID (PSD)	29.87	303.2	0	4.865	2.5215	0.163
Fuzzy with 25 rules	29.96	297.6	– 1.847	5.435	0.0637	0.181
LQI	29.67	291.8	– 3.376	4.339	0.0118	0.146

Table 4. The resulting comparison of motion control by individual controllers

4. DISCUSSION

From the results, the LQI controller can be determined as the best in all assessed parameters except for the distance travelled, which was 0,3 m lower than the average of the remaining. This is due to the smoother acceleration of the robot using this controller, which was reflected in the lowest consumption

of all. Based on a comparison of inputs and outputs of controllers in Figure 4 we conclude that the lowest power consumption of LQI controller W = 291,8 W·s was due to the smoothest changes of controller outputs with $Smoo$ = 0,0118 $m^{-1}s^{-3}$. Furthermore, the LQI controller also came out as the best in position control precision based on I_{AE} = 4,339 m and e_{avg} = 0,146, unlike results of the comparison carried out by [6] where compared controllers were either energy-efficient or precise, but not both.

5. CONCLUSIONS

The presented results of our work brought a comparison of the most popular feedback control algorithms from the not very observed criterion of energy consumption of actuators. In the following phase of the work, we will experimentally verify the functionality of the proposed controllers on the real robot and evaluate the differences between simulations and real experiments, and when observing reserves in the control, we will eventually try to further optimize the control algorithms for the lowest possible consumption. We expect that the greatest differences to simulations will be due to the character of the surface on which we will test the robot, which we have considered to be smooth and without elevations in the presented simulations. Overall, the work focuses on the application of mobile robotics in the field of agricultural production from the view of the impact of different control algorithms on the consumption of mobile robot devices. The results of the work can serve as a choice of control method when is demanded to achieve the lowest possible cost of operating a given mobile robot or autonomous vehicle.

ACKNOWLEDGMENTS

This paper was created within the project VEGA no. 1/0720/18 “Research of Alternative Navigation Algorithms for the Control of Autonomous Robots in Plant Production.”

REFERENCES

1. Mei, Y, Lu, Y, Hu, YC & Lee, CSG 2005, ‘A Case Study of Mobile Robot's Energy Consumption and Conservation Techniques’, *Proceedings of 12th International Conference on Advanced Robotics*, IEEE Operations Service, Seattle, WA, pp. 492-497.
2. Jaramillo, M & Gómez Mendoza, J 2018, “Mixed Energy Model for a Differential Guide Mobile Robot”, *Proceedings of 23rd International Conference on Methods & Models in Automation & Robotics*, Miedzyzdroje, Poland, pp. 114-119.
3. Jaramillo, M, Dogru, S, Gomez-Mendoza, & Marques, L 2020, “Energy estimation for differential drive mobile robots on straight and rotational trajectories”, *International Journal of Advanced Robotic Systems*, vol. 17, no. 2, viewed 6 March 2021 <https://doi.org/10.1177/1729881420909654>.
4. Wang, Y, Xiong, W, Yang, J, Jiang, Y & Shuoyu, W 2019, “A Robust Feedback Path Tracking Control Algorithm for an Indoor Carrier Robot Considering Energy Optimization”, *Energies*, vol. 12, no. 10, viewed 20 January 2021 <https://doi.org/10.3390/en12102010>.
5. Tokekar, P, Karnad, N & Isler, V 2011, “Energy-optimal velocity profiles for car-like robots”, *Proceedings of 2011 IEEE International Conference on Robotics and Automation*, IEEE, Shanghai, China, pp. 1457–1462.
6. Stefek, A, Pham, TV, Krivanek, V & Pham, KL 2020, “Energy Comparison of Controllers Used for a Differential Drive Wheeled Mobile Robot”, *IEEE Access*, vol. 8, pp. 170915–170927, viewed 7 April 2021 <https://ieeexplore.ieee.org/document/9193941>.
7. Beloev, I, Kinaneva, D, Georgiev, G, Hristov, G & Zahariev, P 2021, “Artificial intelligence-driven autonomous robot for precision agriculture”, *Acta Technologica Agriculturae*, vol. 24, no. 1, pp. 48 – 54, viewed 7 April 2021 <https://sciendo.com/article/10.2478/ata-2021-0008>.

8. Hrubý, D & Amrich, M 2008, *Applications of fuzzy control algorithms in agricultural practice*, Nitra, SPU, 123 pp. (In Slovak)
9. Cviklovič, V, Olejár, M, Hrubý, D, Palková, Z, Lukáč, O & Áč, PH 2016, "Navigation algorithm using fuzzy control method in mobile robotics", *Acta Technologica Agriculturae*, vol. 19, no. 1, pp. 19-23, viewed 11 April 2021 < https://sciendo.com/article/10.1515/ata-2016-0004>.
10. Tóth, L, Hrubý, D, Cviklovič, V & Olejár, M 2017, *Algorithms of autonomous mobile robots*, Nitra: Slovak University of Agriculture. 136 s.
11. Dušek, F, Honc, D & Rozsíval, P 2011, "Mathematical model of differentially steered mobile robot", *Proceedings of 18th International Conference on Process Control*, Tatranská Lomnica, Slovakia, pp. 221 – 229.
12. Khatoon, S, Naz, K & Nasiruddin, I 2014, "Eigen Value Analysis of Optimal Controller Design of Wheeled Autonomous Mobile Robot", *International Journal of Computer Applications*, Vol. 103, No. 17, pp. 13-18, viewed 25 November 2020 <https://www.sciencegate.app/document/10.5120/18293-9211>.
13. Munoz-Ceballos, N & Valencia-Velasquez, J 2014, "Benchmark Framework for Mobile Robots Navigation Algorithms", *Revista Facultad de Ingeniería*, vol. 23, no. 36, pp. 6-14, viewed 14 December 2020 <https://revistas.uptc.edu.co/index.php/ingenieria/article/view/2712/>.
14. Dirk-Jan, K 2011, *2D Line Curvature and Normals*, MATLAB Central File Exchange, Octorber 2011, viewed 14 December 2020 <https://www.mathworks.com/matlabcentral/fileexchange/32696-2d-line-curvature-and-normals.
15. Marko, D & Hrubý, D 2020, "Distance measuring in vineyard row using ultrasonic and optical sensors", *Proceeding of 22nd International conference of young scientists*, Praha, Česká zemědělská univerzita, pp. 194 – 204.