PD.TVZ.HR

POLYTECHNIC & DESIGN Vol. 10, No. 3, 2022.

Mrežna inačica: ISSN 2459-6302
Tiskana inačica: ISSN 1849-1995

DOI: 10.19279/TVZ.PD.2022-10-3-09
DOI: 10.19279/TVZ.PD.2022-10-3-19

IMPLEMENTACIJA MATEMATIČKOG MODELA I USPOREDBA UPRAVLJAČKIH ALGORITAMA RUKE INDUSTRIJSKOG ROBOTA

IMPLEMENTATION OF A MATHEMATICAL MODEL AND COMPARISON OF CONTROL ALGORITHMS FOR AN INDUSTRIAL ROBOT ARM

Ivan Šulekić[1], Krešimir Osman[1], Zvonimir Lavrić[2]

[1]University of Applied Sciences – Department of Electrical Engineering, Zagreb, Croatia
[2]University of Applied Sciences – Department of Mechanical Engineering, Zagreb, Croatia

ABSTRACT

The paper presents research on the development of a mathematical model and control algorithms for an industrial robot arm. Starting from the simplified architecture of the robot in plain view, a mathematical model of the direct kinematics with the representation of the position of the robot arm tool is derived. Based on this, the equations of inverse kinematics are derived, which provide the equations for the variables of the angles in the spaces of the ankle. In addition, a dynamic model of the actuator moments in the joints was created. Three algorithms for torque control with PD and PI controllers were proposed for the rotation of the robot joints in which the servo motors are located, as well as for the control with the contact force of the robot tool. The mathematical models and control algorithms were implemented in the computer program MATLAB Simulink, and a simulation of the response variables was performed for each of these algorithms. The simulation results are presented and a comparison of the given algorithms is given.

Keywords: *industrial robot arm, mathematical model, control algorithm, PD regulator, PI regulator, touch force control*

1. UVOD

1. INTRODUCTION

During the introduction of the Bologna Process into the higher education system in the Republic of Croatia, there was a tendency to use various computer programs in the teaching process [1]. By this we mean the use of various computer programs specialized in solving certain types of applications (e.g. MATLAB, AutoCAD, NI Mutisim, LabView, etc.) in technical applications. With their introduction, this type of teaching, especially the conduct of laboratory exercises, has been modernized. What is always set as an objective in such cases when conducting teaching is that [2] such an approach to update education by affirming the use of various information and communication technologies, which serves as an additional incentive and indispensable tool. All this is done with the aim of creating an open learning context that is close to today's generations of students. The use of computers in today's teaching is valued mainly for their potential to create an open, creative learning environment in which students can think and create knowledge at a higher level [3]. Thus, teaching and learning [4] becomes much more interesting, attractive, faster and efficient for newer generations of students through the use of various computer programs. In this way, the approach to learning traditional subject matter is improved [5], regardless of the amount of subject matter taught to students in technical faculties. We can say that this encourages and further encourages students to master the material in certain courses.

The control algorithms for an industrial robot arm presented in this article are used for educational purposes. The main purpose of the robot arm is to test and illustrate the effects of various adjustable control parameters on several control algorithms. Students will have the opportunity to simulate and experimentally perform these different control algorithms. They will also see the effects of the different parameters on the mathematical model of the robotic arm. The goal of the proposed lab exercises is to learn how to optimise algorithms for controlling torque, force, and position. The mathematical models and control algorithms were implemented in the computer programme MATLAB Simulink. For this purpose, the most commonly used control algorithms in industrial robotics were selected. The following control algorithms are presented and compared in this paper: position control with PD, cascade control with PI controllers and impedance and touch force control with PI controller. The most important thing we wanted to achieve is that students recognise the main advantages and disadvantages of each algorithm and learn to distinguish between them.

2. PREGLED SLIČNIH ISTRAŽIVANJA U LITERATURI
2. LITERATURE REVIEW

The work of Ghaleb and Aly [6] presents the modelling, simulation, and control of a robotic arm with two degrees of freedom of movement. An inverse and direct kinematic model and a dynamic model of the observed robot were created. The PID control algorithm was used and the model was created in MATLAB Simulink. In the work of Hueseyinoğlu et al. [7] they also show the modelling of a robot arm with two degrees of freedom of motion. Here, the Lagrange-Euler approach is used to model the dynamic model of the robot. The control algorithm code compares the two algorithms sliding-mode control (SMC) and proportional-integral-derivative (PID) control. A similar approach is also presented in the work of Okubanjo et al. [8], where a dynamic robot model is derived based on second order nonlinear differential equations and using the Euler-Lagrange approach. A PID controller was also used for the control algorithm. The work of the author Titov et al [9] shows the approach to control with the algorithm for force and torque control. All this was also applied to the example of a robot arm with two degrees of freedom of movement. Here, two types of control strategies are applied. One for the slow part of the dynamics of the system based on impedance control and force/torque control with gravity compensation, and trajectory tracking based on impedance control. In the work of authors Yoshida and Hayoshi [10], the application of control algorithms based on force and motion control was studied. A three-legged robot with an ultimate spring end-effector was chosen as an example. A mechanical damper was used to simulate the spring end-effector. Also, a comparison of simulation and experimental data is presented.

3. MOTIVACIJA
3. MOTIVATION

With this paper, the authors aimed to present the implementation and application of several control algorithms in teaching laboratory exercises of a course in robotics using the example of a robotic arm selected for this purpose. The computer programme MATLAB Simulink was chosen in which the control algorithms presented in the paper were implemented. The results presented come from the internal project „KO006-2020/1 - Establishment and equipping of laboratory for the course „Systems and Control Algorithms in Robotics" and „Mobile Robotics" at the Graduate Study of Electrical Engineering at Zagreb University of Applied Sciences, Zagreb, Croatia", which was started in the academic year 2021/22. The idea was to introduce the concept of laboratory exercises on course "Control Systems and Algorithms in Robotics". The mentioned project is currently in the phase of equipping the laboratory with all the equipment and carrying out the accompanying installations and other works necessary for its functioning. The algorithms presented, which the students will perform using the MATLAB Simulink programme, are based on previous laboratory exercises in which the kinematics and dynamics of the robot, as well as the trajectory planning.

4. KINEMATIČKI I DINAMIČKI MATEMATIČKI MODEL ROBOTSKE RUKE

4. KINEMATIC AND DYNAMICAL MATHEMATICAL MODEL OF ROBOT HAND

The mathematical model is presented for a robot arm with two degrees of freedom of movement according to the simplified kinematic scheme in Figure 1. For this purpose, the following parameters are used for the length of each article, their masses and dynamic moments of inertia:

$l_1 = l_2 = 1{,}5$ m – the lengths of individual segments (articles) of the robot arm

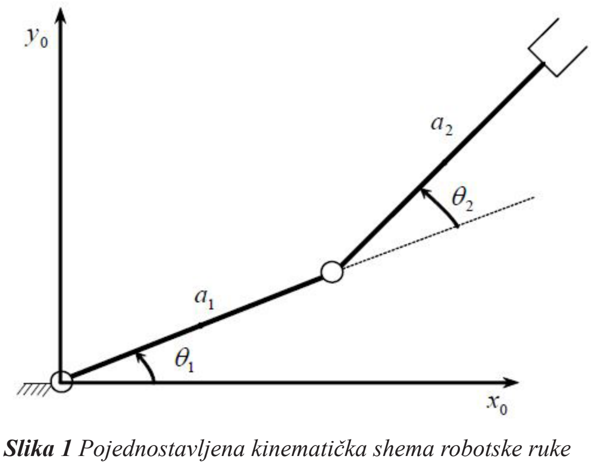
$m_1 = m_2 = 50$ kg – masses of individual segments (articles) of the robot arm

$I_{l1} = I_{l2} = 10$ kgm^2 – dynamic moments of inertia of individual segments (articles) of the robot arm

$m_{m1} = m_{m2} = 5$ kg – masses of DC motors in the joints of the robot arm

$I_{m1} = I_{m2} = 0.01$ kgm^2 – dynamic moments of inertia of individual DC motors in the joints of the robot arm

$k_{r1} = k_{r2} = 100$ – gear reduction ratios in robot arm joints



Slika 1 *Pojednostavljena kinematička shema robotske ruke sa dva stupnja slobode gibanja*

Figure 1 *Simplified kinematic scheme for robot arm with two degrees of freedom*

It is important to note that both segments (articles) of the robot arm have exactly the same structure. Both joints of the robot arm are driven by DC motors.

According to Figure 1, first a direct and then an inverse kinematic model of a robot arm is created and performed. Since similar problems with derivatives of equations have already been presented and derived in the literature [6 - 8], here the authors give only final formulas for the problem, which are important for setting up a mathematical model of a robot arm and for later simulation of individual control algorithms.

Tablica 1 *Denavit-Hartenberg-ovi parametri za robotsku ruku sa dva stupnja slobode gibanja*

Table 1 *Denavit-Hartenberg's parameters for robotic arm with two degrees of freedom*

Link	a_i	d_i	ϑ_i
1	L_1	0	ϑ_1
2	L_2	0	ϑ_2

In deriving the individual equations of motion, we use the Denavit-Hartenberg's (D.–H.) algorithm. For this purpose, it is necessary to define the parameters shown in the following Table 1, according to Figure 1.

Based on the presented parameters and using homogeneous transformation matrices, the equations of the positions (direct kinematics) in the configuration space of the tool p_x and p_y for robot arm with end effector (tool) are determined:

$$p_x = L_1 cos\vartheta_1 + L_2 \cos(\vartheta_1 + \vartheta_2) \quad (1)$$

$$p_y = L_1 sin\vartheta_1 + L_2 \sin(\vartheta_1 + \vartheta_2) \quad (2)$$

The next step in the kinematic analysis is to determine the angles in the observed robot arm joints in the joint space. For this purpose, the equations of direct kinematics (1-2) are used and the following equations of angles (inverse kinematics) are obtained:

$$\vartheta_2 = \pm arctg \frac{sin\vartheta_2}{cos\vartheta_2} \quad (3)$$

where the following is:

$$sin\vartheta_2 = \pm\sqrt{1 - sin\vartheta_2^2} \quad (4)$$

$$cos\vartheta_2 = \frac{1}{2L_1L_2}(p_x^2 + p_y^2 - L_1^2 - L_1^2) \quad (5)$$

The angle ϑ_1 in the first joint is then calculated using the following equation:

$$\vartheta_1 = arctg\frac{(L_1+L_2cos\vartheta_2)p_y \pm L_1 sin\vartheta_2 p_x}{(L_1+L_2cos\vartheta_2)p_x \pm L_2 sin\vartheta_2 p_y} \quad (6)$$

where the following is:

L_1, L_2 [m] – lengths of links 1 and 2,

ϑ_1 [°] – angle closed by the robot body with link 1,

ϑ_2 [°] – angle closed by link 1 with link 2.

Equations (1) - (6) completely define the kinematic model of a robot arm with 2-DOF.

For the dynamic model of the robot arm, the Lagrange-Euler approach is used, which is important for us in the force control algorithm and is based on the Lagrange equation of a different kind.

The equation representing the dynamic model of the robot arm [8] is written in the form:

$$\boldsymbol{D}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{c}(\boldsymbol{q}, \ddot{\boldsymbol{q}}) + \boldsymbol{h}(\boldsymbol{q}) + \boldsymbol{b}(\dot{\boldsymbol{q}}) = \boldsymbol{M_t} \quad (7)$$

where following is:

D(q) – manipulator inertia tensor - symmetric matrix of dimensions *n* x *n*,

c(q,q̈) – connection vector of the ith joint product of joint velocities and connection matrix of velocities C – matrix of dimensions *n* x *n* – represents centrifugal and Coriolis forces,

h(q) – vector of gravitational action - vector of dimensions *n* x 1 – describes the influence of gravity on the manipulator,

b(q̇) – represents the friction that opposes the movement of the robotic arm,

q – vector of joint variables.

As for the kinematic model, there are already derived formulas for the dynamic model [6 - 8].

After conversion and simplification, the following equations of the forces (8) and (9) in the individual joints of robot 1 and 2 are obtained:

The above equations (8) and (9) describe a dynamic model of a robot arm with 2-DOF. The following parameters for DC motors in the robot joints are also used in the further calculation and development of the simulation model:

$D_1 = D_2 = 0.01$Nms/rad – DC motor diameters,

$R_{a1} = R_{a2} = 10\Omega$ – DC motor armature resistances,

$K_{t1} = K_{t2} = 2$ Nm/A – DC motor torque constants,

$K_{v1} = K_{v2} = 2$ Vs/rad – DC motor voltage constants.

In addition, the following condition must be satisfied: $D_i << k_{vi} = k_{ti}/R_{ai}$.

5. PRIKAZ UPRAVLJAČKIH ALGORITAMA ROBOTSKE RUKE

5. PRESENTATION OF CONTROL ALGORITHMS FOR ROBOTIC ARM

5.1. UPRAVLJANJE POZICIJOM KORIŠTENJEM PD ALGORITMA

5.1. POSITION CONTROL WITH PD ALGORITHM

PD regulator with position and velocity control loop is presented in Figure 2 (velocity control loop is in inner loop). The equation that represents PD regulator [11, 13] of the robot arm is written in the form:

$$\boldsymbol{G}(\boldsymbol{s}) = \frac{\boldsymbol{K_p} + K_d s}{1+s\tau} \quad (10)$$

where are:

$\boldsymbol{K_p}, \boldsymbol{K_d}$ – proportional gain of regulator,

$\boldsymbol{\tau}$– time constant of regulator.

$$F_{\vartheta_1} = \left((m_1+m_2)L_1^2 + m_2L_2^2 + 2m_2L_1L_2cos\vartheta_2\right)\ddot{\vartheta}_1 + (m_2L_2^2 - m_2L_1L_2cos\vartheta_2)\ddot{\vartheta}_2 - m_2L_1L_2sin\vartheta_2\left(2\dot{\vartheta}_1\dot{\vartheta}_2 + \dot{\vartheta}_2^{\,2}\right) - (m_1+m_2)L_1gsin\vartheta_1 - m_2L_2gsin(\vartheta_1+\vartheta_2) \quad (8)$$

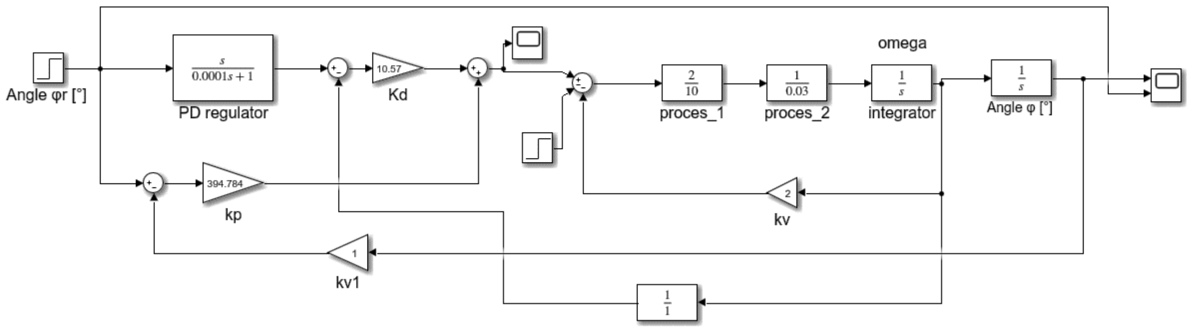
$$F_{\vartheta_2} = (m_2L_2^2 + m_2L_1L_2cos\vartheta_2)\ddot{\vartheta}_1 + m_2L_2^2\ddot{\vartheta}_2 - m_2L_1L_2sin\vartheta_2\dot{\vartheta}_1\dot{\vartheta}_2 - m_2L_2gsin(\vartheta_1+\vartheta_2) \quad (9)$$

Overshoots and oscillations of the response cannot be eliminated only with the PD controller. They depend on the value of ξ which depends on the moment of inertia. The simulation gives the best responses for $\xi = 1$ (Figure 3).

Load was changed from 0 to nominal and it didn't have any effect on changing parameters. Moment of inertia was changed from $0{,}3J_n$ to $3J_n$. The value of the parameter ξ decreases with decreasing moment of inertia and vice versa. The static error is always present when the load is present.

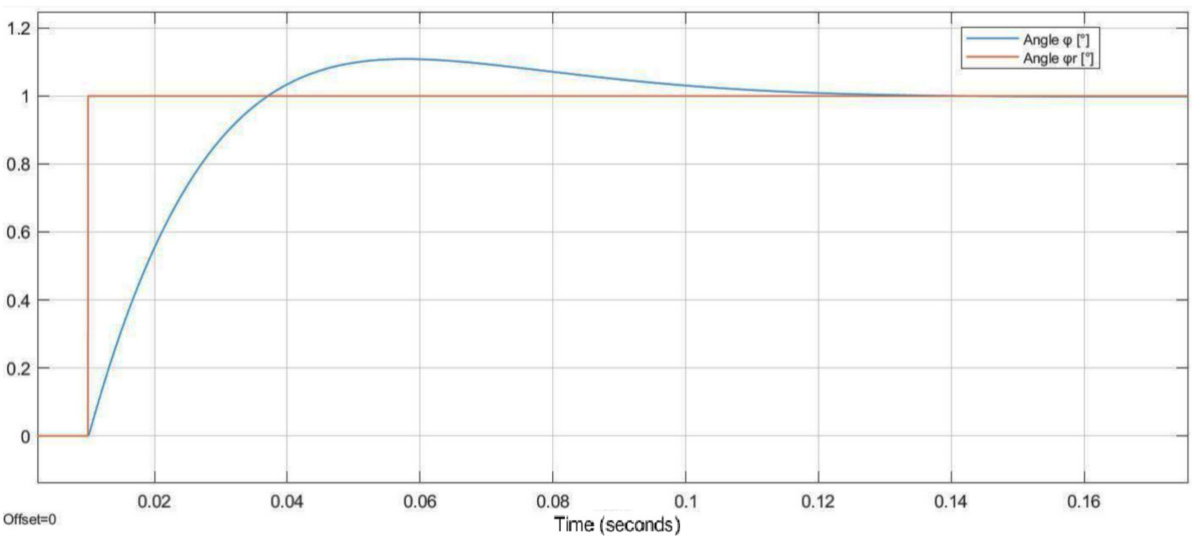
From Table 2 it is shown that with increase of parameter K_p, t_r (rising time) can be decreased but overshoot is rising. Parameter K_d can be used in wide range of values. With its increase overshoot and t_s (settling time) can be decreased. Also, with increase of τ (time constant of PD controller) overshoot and tr (rising time) can be decreased.

For optimal regulation of position control parameters in Table 3 was used.



Slika 2 *Shematski prikaz pomoću blok dijagrama upravljačkog algoritma pozicije sa PD regulatorom*

Figure 2 *Block diagram representation of position control with PD controller*



Slika 3 *Prikaz odziva za upravljanje pozicijom sa PD regulatorom sa parametrima Kp = 394,8 i Kd = 10,57*

Figure 3 *Step response for position control with PD controller with parameters Kp = 394,8 and Kd = 10,57*

Tablica 2 *Prikaz promjenjivih parametara za PD regulator u upravljanju pozicijom*

Table 2 *List of changing parameters for PD controller in position control*

K_p	σ [%]	t_r [ms]	K_d	σ [%]	t_s [ms]	τ [ms]	σ [%]	t_r [ms]
130,0	0,0	50,0	5,0	20,0	262,0	0,1	12,0	20,0
394.8	10,0	21,7	10,5	10,0	26,0	10	20,0	24,0
800,0	20,0	14,4	300,0	0,0	15,0	1000	3,0	56

Tablica 3 *Prikaz optimalnih parametara za PD regulator u upravljanju pozicijom*

Table 3 *List of optimal parameters for PD controller in position control*

K_p	K_d	τ [ms]
394.8	10,5	0,1

These parameters are optimal for wide range of loads.

5.2. UPRAVLJANJE POZICIJOM I BRZINOM SA PI REGULATOROM (KASKADNA REGULACIJA)

5.2. POSITION AND VELOCITY CONTROL WITH PI ALGORITHM (CASCADE CONTROL)

With this algorithm position of robotic hand can be easily controlled. In order to provide smooth motion, the velocity controller must have a filtered velocity signal obtained from the resolver. This filtering is necessary because the step-like nature of the stepper motor introduces high frequency noise into the position measurement. The velocity controller acts first, following a motion profile until the actuators position resides within a specified distance of the target position. Once within this target position, the position controller takes over and positions the actuator to within a smaller distance of the target position. Finally, when the actuator has been adequately positioned, both controllers quit operating on the actuator in order to prevent dithering and lock the actuator position in place.

The equation that represents PI algorithm [11, 13] for cascade control of the robot arm is written in the form:

$$G(s) = \frac{K_{p,v}(1+T_{p,v}s)}{s} \qquad (11)$$

where are:

$K_{p,v}$ – proportional gains of velocitiy and position regulator,

$T_{p,v}$ – time constants of velocitiy and position regulator

Based upon known values of ξ and ω it is possible to calculate K_P and K_V coefficients of speed and position regulator. Overshoot and rising time of the response depend on value of ξ. With the implementation of speed feedback loop overshoot of the response was eliminated. With the implementation of time constant of speed regulator equal to the electromechanical constant of the system, control system became independent of the systems moment of inertia. Static error was eliminated with $K_{FP} = 1$ in feedback loop. PI regulator ensures nonconciliatory response during hand movement.

Tablica 4 *Prikaz promjenjivih parametara za regulator pozicije u petlji upravljanja pozicijom i brzinom (upravljanje pozicijom)*

Table 4 *List of changing parameters of position controller for control algorithm with position and speed control loop (position control)*

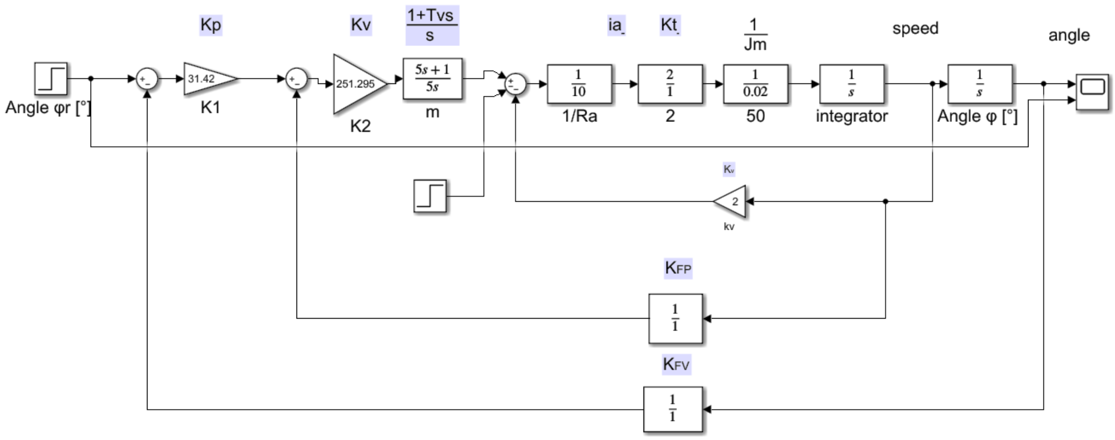
K_p	σ [%]	t_r [ms]
10,0	0,0	105,0
31,4	0,0	38,0
100,0	12,0	14,4

Tablica 5 *Prikaz promjenjivih parametara za regulator brzine u petlji upravljanja pozicijom i brzinom (upravljanje brzinom)*

Table 5 *List of changing parameters of speed controller for control algorithm with position and speed control loop (speed control)*

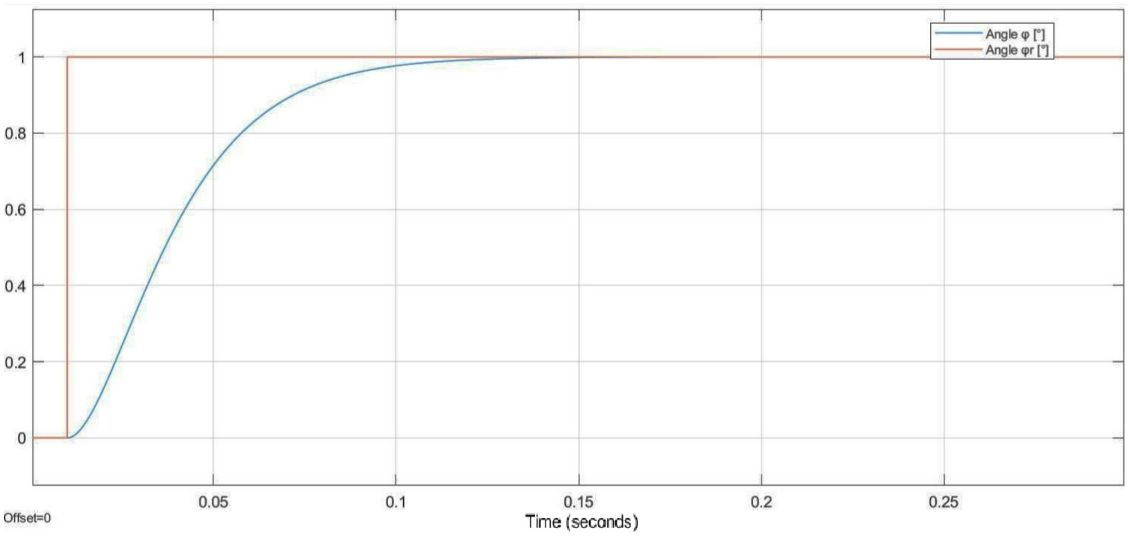
K_v	σ [%]	t_s [ms]	T_v	σ [%]	t_s [ms]
10	50,0	1,6	0,01	20	0,44
100	8,0	0,15	0,05	0	0,15
251,3	0,0	0,10	0,1	0	0,13

From Table 4 it is shown that with increase of parameter K_p, t_r (rising time) is smaller and beyond $K_p > 100$ overshoot becomes significant (beyond 20%). From Table 5 it is shown that with increase of parameter K_v, t_s (settling time) is reducing rapidly. Beyond $K_v > 100$ there is no overshoot. With all $T_v > 0,05$ there is no overshoot. From these parameters it shown that we should have $K_p < 100$, $K_v > 100$ and $T_v > 0,05$. Regulator gives the same results for a range of loads from value 0 to nominal. It is concluded that it's very robust.



Slika 4 *Shematski prikaz pomoću blok dijagrama upravljanja pozicijom i brzinom sa PI regulatorom (kaskadna regulacija)*

Figure 4 *Block diagram representation of position and velocity control with PI algorithm (cascade control)*



Slika 5 *Prikaz odziva za kaskadno upravljanje sa parametrima Kp=31,4, Kv=251,3, Tv=0,05*

Figure 5 *Step response of cascade control with parameters Kp=31,4, Kv=251,3, Tv=0,05*

5.3. UPRAVLJANJE IMPEDANCIJOM I SILOM DODIRA

5.3. IMPEDANCE AND TOUCH FORCE CONTROL

This control algorithm (Figure 6) can be used to control the touch force applied to the surface. Often this algorithm is combined with a control algorithm with position and velocity control loop. First, the control algorithm with position and velocity control loop is used to move the robot arm into position, and then the touch force control algorithm is used to manipulate objects. The equation that represents PI algorithm for impedance and touch force control [13] of the robot arm is written in the form:

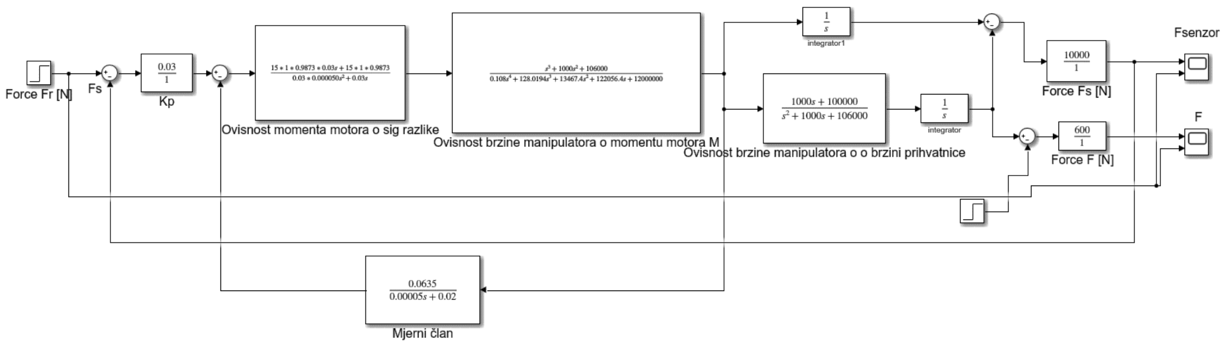
$$\boldsymbol{G(s)} = \frac{\boldsymbol{K_p}(1+T_p s)}{s} \qquad (12)$$

where are:

$\boldsymbol{K_p}$ – proportional gain of regulator,

$\boldsymbol{T_p}$ – time constant of regulator.

Figure 7 shows best response for regulator with parameter K_p = 0,03. From Table 6 it is shown that with K_p > 0,03 overshoot and ripples will increase and with K_p < 0,03, resulted signal would be aperiodic with bigger rise time. If value of K_p > 14,4 then signal will become unstable. Changing time constant T_p regulator gives similar results.



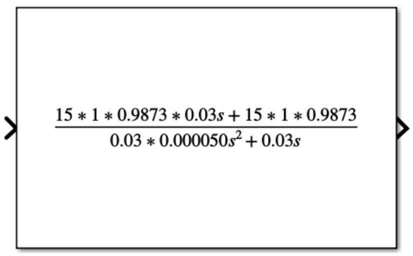
Slika 6 *Shematski prikaz pomoću blok dijagrama upravljačkog algoritma korištenjem impedancije i sile dodira*

Figure 6 *Block diagram representation of impedance and force touch control*

Tablica 6 *Prikaz promjenjivih parametara za regulator brzine u upravljanju impedancijom i silom dodira*

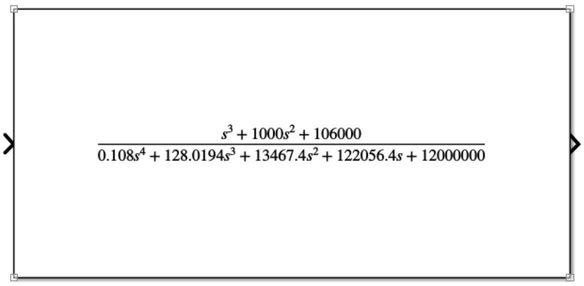
Table 6 *List of changing parameters of speed controller for impedance and touch force control*

K_p	T_p [ms]	σ [%]	t_s [ms]
0,003	0,005	0,0	4,2
0,030	0,030	11,0	0,4
1,000	0,100	64,0	0,22



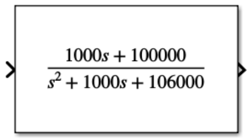
Slika 6.1 *Blok Ovisnost momenta motora o signalu razlike*

Figure 6.1 *Block Motor torque dependence on differential signal*



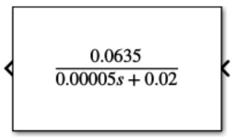
Slika 6.2 *Blok Ovisnost brzine manipulatora o momentu motora M*

Figure 6.2 *Block Manipulator speed dependence on motor moment*



Slika 6.3 *Blok Ovisnost brzine manipulatora o brzini prihvatnice*

Figure 6.3 *Block Gripper dependence on manipulator speed*



Slika 6.4 *Blok Mjerni član*

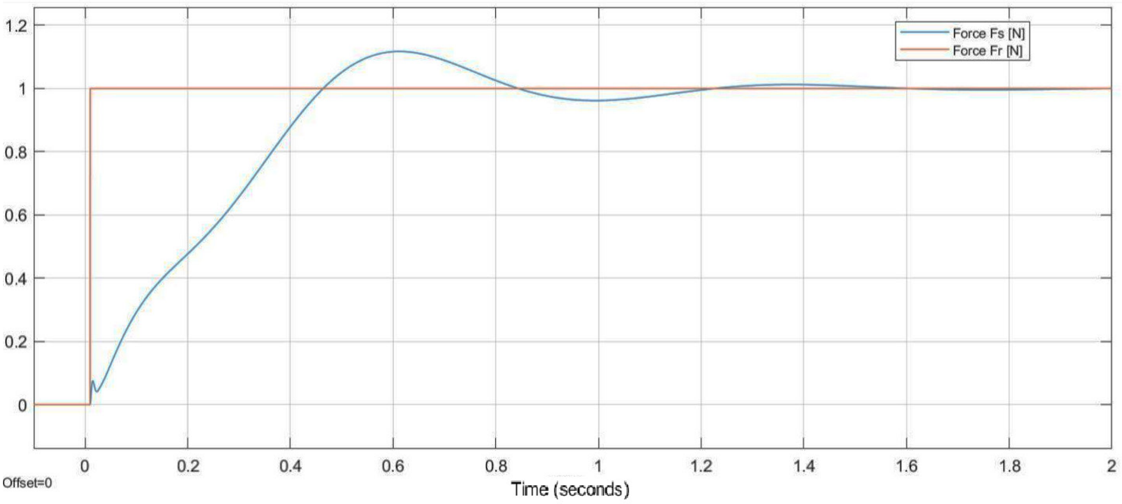
Figure 6.4 *Block measuring element*

5.4. USPOREDBA PRIKAZANIH UPRAVLJAČKIH ALGORITMA

5.4. COMPARISON OF PRESENTED CONTROL ALGORITHMS

By comparing Figure 3 and 5, it is shown that rise time of step response for cascade controller is larger than for PD controller, but there is no overshoot. The overshoot for PD controller is 10 %. The static error has value 0 for both controllers.

The impedance and touch force control (Figure 6) are used in combination with the cascade and PD control. The impedance and touch force control are used for fine positioning along the surface of the object. From Figure 7 and Table 6, it can be seen that with the increase of parameter K_p, t_r (rise time) becomes smaller, but overshoot increases. Simulations were performed for loads from value 0 to $3M_n$ and the results were similar.



***Slika** 7 Prikaz odziva za upravljanje impedancijom i silom dodira sa parametrom Kp = 0,03*

***Figure** 7 Step response of impedance and force touch control with parameter Kp = 0,03*

Aim of this comparison is to collect data that would help to choose right regulators parameters for different usage of a robotic arm.

6. ZAKLJUČAK I MOGUĆI SMJEROVI BUDUĆIH ISTRAŽIVANJA

6. CONCLUSION AND POSSIBLE WAYS OF FUTURE RESEARCH

In this paper, a simulation of the automatic control of a robot arm using different control algorithms is presented. A robotic arm with 2-DOF was chosen as an example. The following can be concluded:

- the torque control using the PD algorithm is based on the proportional relationship between motor torque and armature current in controlling the motor armature current. The main disadvantages of this algorithm are the precision of positioning and the need for the best possible dynamic characteristics of the servomotor torque.
- cascade torque and speed control are used when you want to isolate the primary loop from disturbances, nonlinearities and problems related to the executive element. It contains two control loops: an external position control loop and a slave speed control loop. The main difference is the different way of inputs. You can get a much more finely tuned system.
- the control with touch force of the robot arm also contains two control loops: the position control, and the force control, when the motion is limited by the environment. It is achieved by adjusting the impedance control parameters to achieve the desired flexibility with the given design of the robot body. This is achieved by a combination of simultaneous control of position and contact force. The advantage of this algorithm is that it can be used for fine positioning of the robot arm tip.

One of the next steps in further research would be to investigate some advanced control algorithms using the same example of a robotic arm. For this purpose, the following control algorithms would be interested: control with PD controller in the operating space with gravity compensation, Hsia's method of robust and adaptive control in joint space and hybrid control of manipulator position and touch force.

PRIZNANJE

ACKNOWLEDGMENT

The research presented in this paper is part of internal project „KO006-2020/1 - Establishment and equipping of laboratory for the course „Systems and Control Algorithms in Robotics" and „Mobile Robotics” at the Graduate Study of Electrical Engineering at Zagreb University of Applied Sciences, Zagreb, Croatia.

7. REFERENCE
7. REFERENCES:

[1.] Katić Žlepalo, M. and Kovačić, B.; O primjeni računalnih programa u nastavi matematičkih predmeta na Tehničkom veleučilištu u Zagrebu, Poučak: časopis za metodiku i nastavu matematike, 58, 2014, pp. 69-79.

[2.] Vrkić Dimić, J.; Razvoj paradigmi i modela uporabe računala u nastavi: od pomoći u poučavanju prema kreativnom i otvorenom kontekstu učenja, Acta Iadertina, 7, 2010, pp. 113-124, 2010.

[3.] Overbaugh, R. and Lu, R.; The Impact of a NCLB-EETT Funded Professional Development Program on Teacher Self-Efficacy and Resultant Implementation, Journal of Research on Technology in Education, 41(1), 2008, pp 43-61.

[4.] Saadon, S.; Azmin Sham Rambely, A and Mohd Suradi, N.R.; The Role of Computer Labs in Teaching and Learning Process in The Field of Mathematical Sciences, Kongres Pengajaran dan Pembelajaran UKM, 2010, Procedia Social and Behavioral Sciences, 18, 2011, pp. 348–352.

[5.] Ravishankar Rao, A. and Riddhi, D.; Developing hands-on laboratory exercises for teaching STEM students the internet-of-things, cloud computing and blockchain applications, 2019 IEEE Integrated STEM Education Conference (ISEC), 2019, pp. 191-198.

[6.] Ghaleb N.M. and Aly, A.A.; Modelling and Control of 2-DOF Robot Arm, International Journal of Emerging Engineering Research and Technology, 6(11), 2018, pp. 24-31.

[7.] Hüseyinoğlu M. and Abut, T.; Dynamic Model and Control of 2-DOF Robotic Arm, European Journal of Technic (EJT), 8, 2018, pp. 141-150.

[8.] Okubanjo, A.A.; Oyetola, O.K.; Osifeko, M.O.; Olaluwoye, O.O. and Alao, P.O.; Modelling of 2-DOF Robot Arm and Control, Futo Journal Series (FUTOJNLS), 3(2), 2017, pp. 80 – 92.

[9.] Titov, V.; Shardyko, I. and Isaenko, S.; Force-Torque Control Implementation for 2 DOF Manipulator, 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, 2013, Procedia Engineering, 69, 2014, pp. 1232 – 1241.

[10.] Yoshida, Y. and Hayashi, D.; Force and Motion Control for Cylindrical Curved Surface Wall Using a Three-Link Arm, 2008 10th International Conference on Control, Automation, Robotics and Vision, Hanoi, Vietnam, 2008, pp. 2283-2288.

[11.] Vukić, Z. and Kuljača, Lj.; Automatsko upravljanje – analiza linearnih sustava, Kigen, Zagreb, Hrvatska, 2005.

[12.] Astrom, K.J.; Hagglund, T.; PID Controllers: Theory, Design, and Tuning, second edition, International Society for Measurement and Control, Research Triangle Park, NC, USA, 1995.

[13.] Kovačić, Z.; Bogdan, S. and Krajči, V.; Osnove robotike, Graphis, Zagreb, Croatia, 2002.

AUTORI · *AUTHORS*

● **Ivan Šulekić -** obtained his Diploma in Electrical Engineering during the year 2008 at University of Zagreb, Croatia. Currently he is employed as Lecturer, Zagreb University of Applied Sciences, Department of Electrical Engineering, Croatia. His research and professional interests include automatic control, process control, robotics and design and supervision of electrical installations in building and industry.

Korespondencija · *Correspondence*
ivan.sulekic@tvz.hr

● **Krešimir Osman -** Krešimir Osman obtained his M.Sc. in Electrical Engineering and Ph.D. degree in Mechanical Engineering during the year 2011 and 2014 at University of Zagreb, Croatia. Currently he is employed as Lecturer, Zagreb University of Applied Sciences, Department of Electrical Engineering, Croatia. His research and professional interests include system structuring, modelling and optimization, decision making in engineering design, robust design, mechatronic system design, robotics and simulation-based design.

Korespondencija · *Correspondence*
kresimir.osman@tvz.hr

● **Zvonimir Lavrić -** obtained his Diploma in Electrical Engineering during the year 2021 at Zagreb University of Applied Sciences, Croatia. Currently he is employed as Teaching Assistant, Zagreb University of Applied Sciences, Department of Mechanical Engineering, Croatia. His research and professional interests include machine learning, embedded computer systems, robotics and programming algorithms.

Korespondencija · *Correspondence*
zvonimir.lavric@tvz.hr