Management Systems
in
Production Engineering

2025, Volume 33, Issue 4, pp. 517-524

Date of submission of the article to the Editor: 04/2025
Date of acceptance of the article by the Editor: 10/2025

DOI 10.2478/mspe-2025-0052

INTEGRATED MULTIPHYSICS SIMULATION AND BUILT IN CONTROL DESIGN OF A THREE LINK ROBOTIC MANIPULATOR

Ľubica Miková, Erik Prada, Michal Kelemen
Technical University of Kosice

Abstract:
The aim of this paper is to create a complex model of a robotic manipulator in a multiphysics simulation environment and design its control exclusively using the built-in resources of this environment, without the application of external analytical or numerical methods for calculating controller parameters. The introductory chapter sets the goal and emphasizes the importance of control systems in technical practice. At the same time, the basic parameters of the modeled mechanism and the characteristics of the operation that the simulated system is to perform are specified. In the following chapters, a simulation model of the mechanism is derived. The kinematic structure of the robotic arm, the number of degrees of freedom, as well as key dynamic characteristics that significantly affect the behavior of the system during movement are described. This model forms the basis for subsequent virtual implementation in the simulation environment. The next part is devoted to the implementation of the simulation model, where the mechanism structure consists of a stationary base, three rotating links and an end effector. The chapter describes in detail the specified motion profiles and boundary conditions of the simulation, which reflect realistic loading and operating modes of the system.

Key words: robotic manipulator, simulation model, controller, MSC Adams

INTRODUCTION

The control of three-arm (3-DOF) robotic manipulators is an important area in robotics, with applications in industrial automation, medicine or autonomous systems. Classical control methods, such as PID control and sliding mode, are often used to achieve accurate trajectory tracking and robustness to disturbances [1]. Adaptive and intelligent control techniques offer flexibility in controlling systems with unknown dynamics and provide better performance under complex conditions [2]. Dynamic modelling of manipulators, for example using Denavit-Hartenberg convention and Newton-Euler formulations, is essential for accurate design of control algorithms [3]. In recent years, simulation tools have been increasingly used to design and verify the control of 3-DOF robotic arms [4]. Practical applications, such as three-arm handling systems for drones or planar robotic arms, place high demands on design, manufacturing, and control [5, 6].

In the field of automated control, controllers play an important role in ensuring the correct operation of systems. Their task is to compare the desired value with the actual output value and, based on the error that has arisen, to create a control intervention that minimizes this deviation. Among the most widespread and most widely used are PID controllers, which consist of three basic components – proportional (P), integrator (I), and derivative (D). The proportional part reacts to the current error, the integrator fixes any ongoing mistakes over time, and the derivative part anticipates how the error will change, helping to reduce sudden shifts and make the system more stable. [1] Correct adjustment of individual controller parameters (P, I, D) is essential for achieving the desired system behavior, stability, and response speed. Various methods are used to design these parameters, such as the Ziegler-Nichols method, the pole-zero method, the Skogestad method (SIMC), or the dynamic inversion method. In practice, simplified types of controllers are also often used. The P-controller (proportional) responds only to the magnitude of the error and is suitable for systems where a certain permanent deviation can be accepted. The PI controller combines the proportional and integral components and ensures stable control with zero steady-state error, and is often used, for example, in temperature or flow control. The PD-controller is suitable in cases where a fast response without significant overshoot is important, but complete elimination of the error is not a priority. In dynamically changing conditions or in systems whose model is not exactly known, adaptive controllers are used. They can adapt their parameters in real time based on the behavior of the system. They are used, for example, in robotics, aviation or VC process control with variable loads. For systems with nonlinear behavior or with a difficult-to-define model, it is appropriate to use fuzzy controllers. These are based on rules derived from system observation and operate with linguistic descriptive rules such as "if the error is large, increase control intervention." These controllers are suitable for smart home systems, air conditioning units, or automated appliances. [7, 8]. In the paper “PD/PID-fuzzy logic controller-based tracking control of 2-link robot manipulator” [9] presents a PD-Fuzzy logic control approach for trajectory tracking of a 2-link robotic manipulator subject to uncertainties and disturbances. Using the error and its derivative as inputs, the fuzzy logic controller compensates disturbances, providing accurate tracking in real-time. The system was modeled mathematically via Euler-Lagrange equations, and simulations conducted in MATLAB demonstrated minimal tracking errors. Results show rapid error convergence to zero after 0.05 s and 0.07 s for joints 1 and 2, respectively, confirming the effectiveness of the proposed method. Future extensions could involve applying this control strategy to multi-link manipulators and adaptive fuzzy sliding mode control systems. If the control objective is to optimize a certain performance indicator of the system (e.g., minimizing energy, time, or error), it is appropriate to use optimal controllers, such as the LQR (Linear Quadratic Regulator). This type of control requires an accurate mathematical model of the system but allows for high control quality, especially in more demanding applications [8]. The selection of a suitable controller depends on the specific system, the desired control properties, the availability of the model, and the computational capabilities. In most industrial applications, a properly designed PI or PID controller is sufficient, while for more complex or adaptive tasks, more advanced control methods are appropriate. In Paper „Tracking Control of Robotic Manipulators With Uncertain Kinematics and Dynamics“ [10] the authors propose an observer-based control strategy utilizing two terminal sliding-mode observers to estimate uncertain kinematics and unknown torques, including gravitational and frictional effects. Based on these estimates, a nonlinear controller compensates uncertainties, ensuring zero tracking error in finite time. The method was validated numerically and experimentally using a 7-DOF industrial robotic manipulator. Results demonstrated the controller's high accuracy, convergence, and robustness, effectively guaranteeing precise trajectory tracking despite system uncertainties. Next paper [11] addresses control challenges of flexible link systems, aiming to balance transient position performance with minimal tip vibration using a PID controller tuned via Multiobjective Differential Evolution (MODE). MATLAB-based optimization provides Pareto-optimal solutions for both objectives. The proposed MODE-based PID controller is benchmarked against an LQR controller supplied by QUANSER. Results from simulations and experiments indicate that while the LQR controller offers superior transient position responses, the MODE-PID controller significantly reduces tip vibration. Combining these advantages with the PID controller’s simplicity and ease of implementation makes it preferable. Next article [6] presents a differential evolution (DE)-based method for dual-arm cooperative manipulation tasks in robotic systems. The approach utilizes Denavit-Hartenberg kinematic representation, eliminating the need for Jacobian inversion, thus effectively avoiding singularities and handling joint limit constraints through penalty functions. Simulation and real-world experiments using a dual-arm KUKA Youbot validated the approach. The proposed method significantly outperformed conventional methods in handling singularities and diverse robotic configurations, proving its suitability for redundant/non-redundant, revolute/prismatic systems. Future research includes extending the methodology to systems with multiple manipulators. In the following article [12], SS-ASMC (strong and stable adaptive sliding-mode control) with two coupled adaptive laws provides fast stable switching gains, resulting in high tracking accuracy and increased robustness. Using time-delayed estimation and pole placement, it eliminates the need for a detailed model, while Lyapunov analysis confirms BIBO stability and simulations on robotic manipulators show superiority over existing approaches. In the following article [13], an adaptive admission law was proposed that dynamically adjusts the robot stiffness in critical areas of the workspace (singularities, joint constraints, and workspace), ensuring safe and smooth manual interaction. Experiments with constraint activation test and 3D tracing confirm better accuracy, speed, and user satisfaction compared to traditional admission parameter tuning strategies. In the following article [14], dual-loop hierarchical control combines an external optimal kinematic loop controller with an internal adaptive NN-AC dynamic controller, which provides both globally optimal trajectories and cost-effective action inputs. The kinematic control algorithm uses closed-loop optimal control via analytical solution of the HJB equation while maintaining feedback tracking, while the dynamic module minimizes long-term costs through online learning. Lyapunov analysis and experiments on a 7-DOF Barrett WAM in a warehouse environment confirmed stability, robustness, and smoother, more economical trajectories compared to standard methods. In the following paper [15], they compare the basic ABC and the improved MGABC algorithm for optimizing PID gains for a 3-DOF rigid manipulator using a new Lyapunov-based objective function. MGABC achieves faster convergence and better escape from local optima than the basic ABC. The optimized PID with LBF provides significant improvements in path tracking, robustness against disturbances and load variations even without vibration at flexible joints.

MATHEMATICAL MODEL OF THE MANIPULATOR

A robotic arm is a more complex electromechanical system that exhibits complex coupling and nonlinear properties. In order to evaluate the overall performance of the robotic arm, its mechanical properties and the control system scheme must be taken into account. For the traditional design approach, it is necessary to use repeated iterations. This method is not only challenging to improve the mechanical properties of the robotic arm, but can also consume a large amount of financial and human resources, which is particularly problematic if the design of the mechanical system is performed independently of the control system, without finding relationships between them [16, 17].

However, if we build a virtual prototype before building a real model, we can perform kinematic and dynamic simulation of the robotic arm, which will allow us to improve the mechanical system of the machine by simulating the control system. This approach will greatly shorten the development cycle and promote the solution of the dynamic problems of the robotic arm and improve the efficiency of research and development.

A robot kinematic representation is a mathematical description of the kinematic structure of the robot. Probably the most common mathematical representation of robot kinematics is the Denavit_Hartenberg (DH) conventionThe robotic arm with three degrees of freedom has three rotary joints: the base joint, the big-arm joint and the elbow joint. The kinematic representation of this type of RR is represented using the modified D-H method. Fig. 1 shows us the D-H configuration of the RR with the parameters and configurations that will be used for dynamic calculations and control design [17, 18, 19].

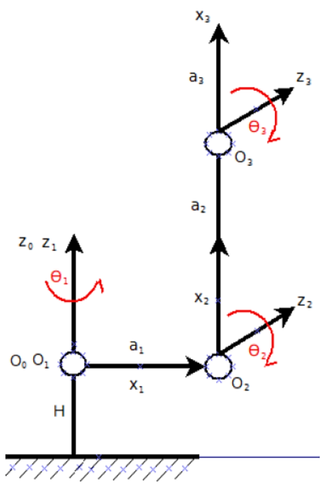


Fig. 1 D-H manipulator configuration

Denavit-Hartenberg method is used to analytically describe the kinematics of robotic arms and manipulators using homogeneous transformation matrices. This formal approach systematically defines the interrelationships between the links of serial kinematic chains, thus facilitating computations for forward and inverse kinematics [20, 21].

Defining the general form of the homogeneous transformation matrix (${}^{i-1}_{i}\boldsymbol{T}$):

$$ {}^{i-1}_{i}\boldsymbol{T} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i\cos\alpha_{i-1} & \cos\theta_i\cos\alpha_{i-1} & -\sin\alpha_{i-1} & -\sin\alpha_{i-1}\, d_i \\ \sin\theta_i\sin\alpha_{i-1} & \cos\theta_i\sin\alpha_{i-1} & \cos\alpha_{i-1} & \cos\alpha_{i-1}\, d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

where:

θ_i is angle about the z-axis $zi{-}1$,

d_i displacement along the axis $zi{-}1$, which defines the distance between the starting points of two consecutive coordinate systems,

a_i length of the articleor the distance between the axes $zi{-}1$ a zi along the axis xi a α_i rotation of the coordinate systems $zi{-}1$

zi along the axis xi.

The homogeneous matrix for the end-effector (${}^{3}_{0}\boldsymbol{T}$) has the form:

$$ {}^{3}_{0}\boldsymbol{T} = \begin{bmatrix} \sin(t_2+t_3)\cos(t_1) & \cos(t_2+t_3)\cos(t_1) & -\sin(t_1) & \cos(t_1)\,(a_1+a_2\sin(t_2)) \\ \sin(t_2+t_3)\sin(t_1) & \cos(t_2+t_3)\sin(t_1) & \cos(t_1) & \sin t_1\,(a_1+a_2\sin(t_2)) \\ \cos(t_2+t_3) & -\sin(t_2+t_3) & 0 & a_2\cos(t_2) \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

CREATE A MODEL OF THE MANIPULATOR IN THE MSC ADAMS/VIEW SIMULATION ENVIRONMENT

This chapter focuses on the detailed procedure for designing the robotic arm model in the MSC Adams/View simulation environment. The resulting model was then subjected to analysis, with all calculations and verifications performed through numerical simulation. A key aspect within the design was the correct choice of the geometric and mass parameters of the model, as well as the determination of an adequate simulation duration to capture all relevant phenomena.

Through the simulation analysis, it was possible to accurately identify the required output quantities, especially by graphically representing the motion trajectory of the robotic arm and its end effector. The obtained results serve as a basis for the next steps in the design of the control system. The main objective of this stage was to create a functional model of a robotic arm with three degrees of freedom that can perform manipulation tasks – namely, moving objects, in this case cubes, from one platform to another (Fig. 2).

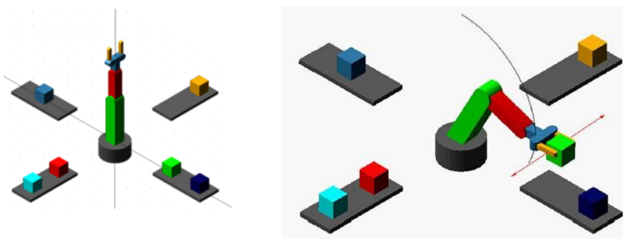


Fig. 2 Model of the workplace with the robotic arm and plotting the motion trajectory of the effector base

Figure 3 shows the moments required to ensure movement in each axis. These forces depend on the dynamics of the motion, the mass and the position of the individual links. They also point out the areas of greatest load, which is key in controller design to compensate for significant dynamic effects.

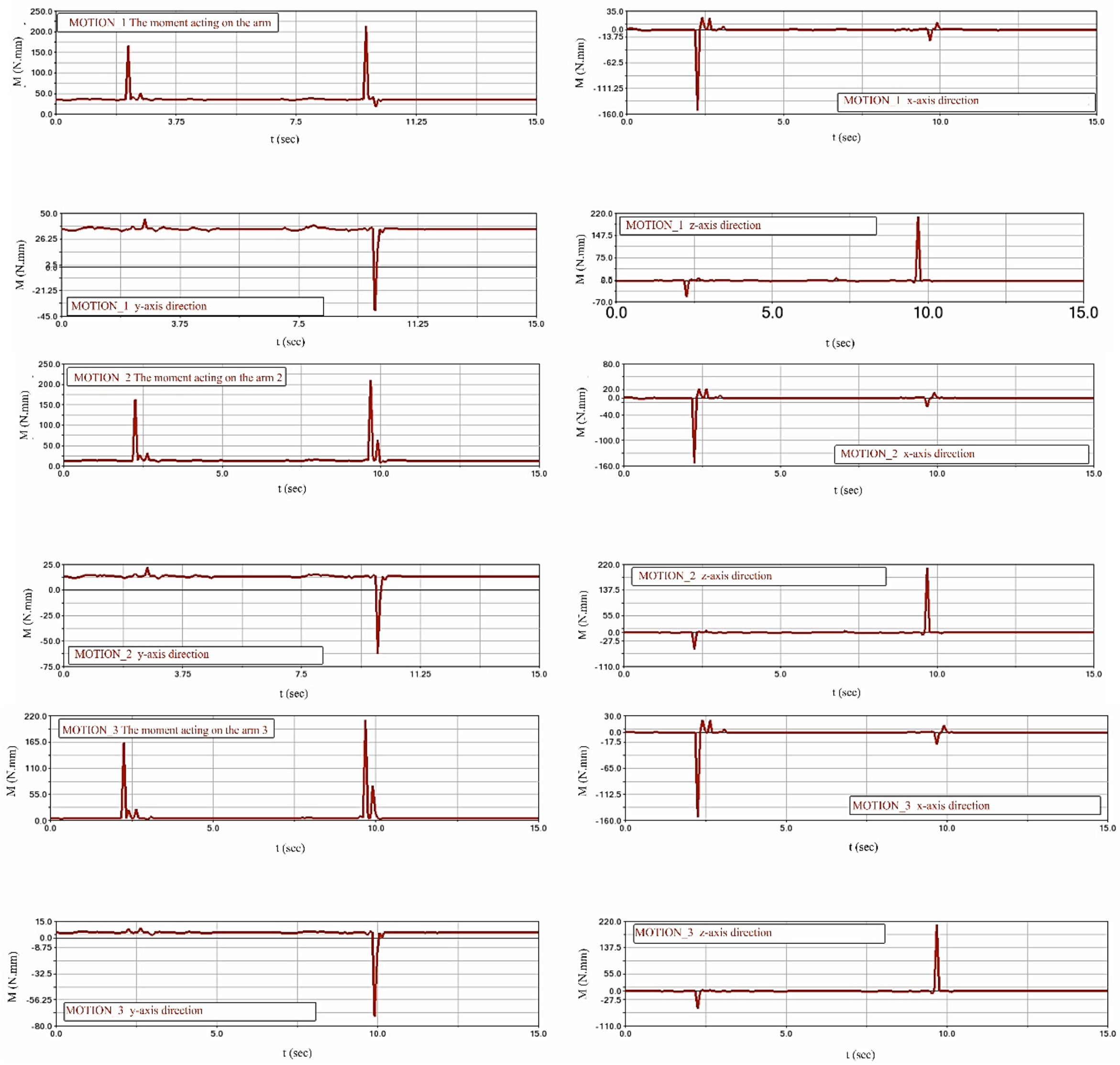
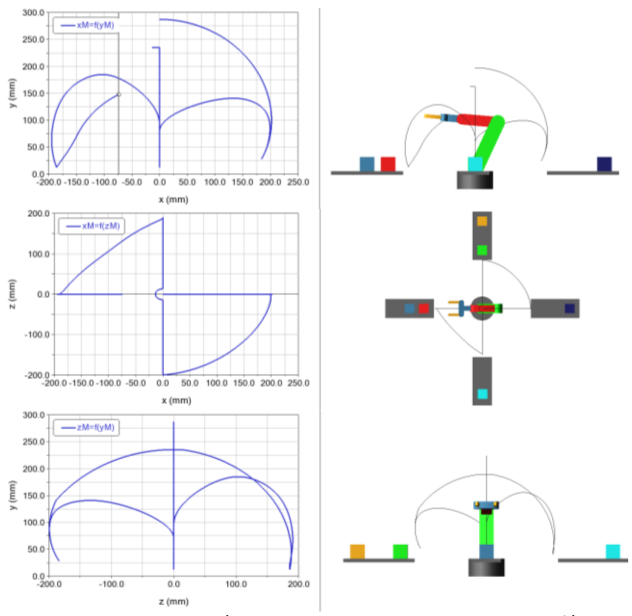


Fig. 3 Moments acting on the individual members

The trajectory of the effector motion from different viewing angles is shown in Figure 4a. This visual output provides a spatial view of the motion profile of the effector across the entire workspace of the mechanism. In parallel, the correctness of the trajectory representation was verified through a post-processing module in a software environment, where a comparison plot was generated based on numerical simulation. The outputs obtained from the postprocessor (Fig. 4b) showed complete agreement with the simulation data exported from the dynamic model, thus confirming the correct functionality of the visualization tools and also verifying the accuracy of the proposed model. This correlation between visual and numerical output is an important step in model validation and ensures the reliability of further analyses performed in the MSC Adams/View environment.



a) b)

Fig. 4 The trajectory of the effector from the front, top and the right b) and its comparison in Postprocessor a)

CONTROLLER DESIGN

Conventional approaches and methods applied in the production environment are gradually becoming non-sufficient due to the increasing competitive pressure [22]. Proportional control (P-control) is a simple but often sufficient way of controlling mechanical systems where the goal is a fast and robust response to a desired input. In this paper, the control design focuses on the implementation and analysis of P-control of manipulator motion in the ADAMS View environment, which allows accurate simulation of multibody dynamics and detailed modelling of motion constraints [23, 24]. The simulation results show the practicality of using simple control structures in combination with the advanced simulation tool ADAMS View, which can be particularly beneficial in the early design phases of mechatronic systems [25, 26]. This chapter focuses on the design of the control mechanism for the created robotic arm model. At the beginning, a control structure will be presented that shows the relationships between the system elements and the control circuit. To achieve a stable torque that ensures the effector remains in the desired position even when the angle of the robot arm changes in the horizontal plane, it is necessary to implement a proportional closed-loop feedback controller according to the control scheme shown in Fig. 5.

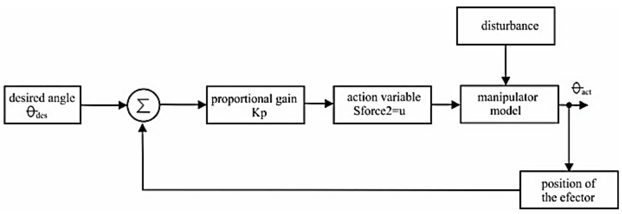


Fig. 5 The block structure of the control circuit

The control system is implemented in the MSC Adams/View environment via the Controls Toolkit module, which provides a library of standardized blocks for control circuit design. This toolkit allows the creation and configuration of feedback control loops directly in the simulation environment without the need for external interfacing. Within the proposed control structure, the following are defined as system input signals: the desired value of the arm rotation and the actual measured value, based on which a control intervention is generated in the system. An important part of this is to ensure that torque is applied between the effector and arm bodies at the marker location (Fig. 6). The torque value is calculated by the control system to keep the effector in a horizontal position. In modelling, it is important to maintain the correct definition of the action and reaction terms to keep the effector in the desired horizontal position.

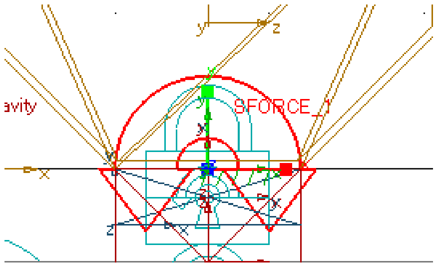


Fig. 6 Ensuring the application of torque between the effector bodies and the arms at the marker location

In the following simulation results, the dynamic behaviour of the designed model will be demonstrated in two different configurations – without control and with the implemented control algorithm. The aim of this analysis is to compare the system response in both cases and to evaluate the effectiveness of the applied control algorithm in terms of accuracy and dynamic performance. The simulations were carried out in the MSC Adams/View environment, and the post-processing module of this software was used to process and evaluate the results. Thanks to this methodology, it was possible to visualize and quantify the key motion parameters and interactions of the individual components of the mechanical system, thus obtaining a comprehensive view of the dynamic characteristics of the model under different control conditions.

The angular velocities ω that were generated in actuators 1 to 3 (denoted as MOTION_1 to MOTION_3) of the robotic arm model prior to the application of control action via the P-controller are shown in Figs. 7-9. These plots provide a detailed insight into the dynamic response of the system in the state without control action, i.e. in the so-called open loop. The recorded data reveals how the individual actuators behave under the influence of the control signals without feedback, allowing a better understanding of the inherent dynamics of the mechanism, including possible oscillations, time delays and stability characteristics before the actual introduction of control.

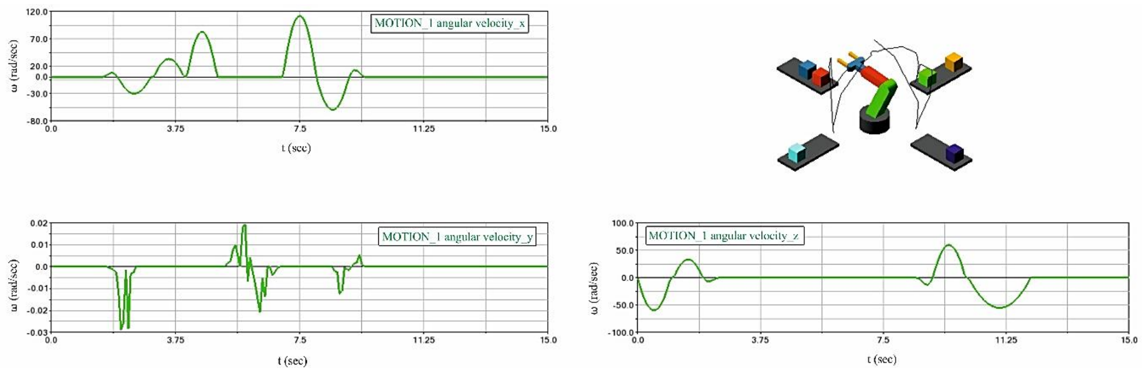


Fig. 7 Angular velocity MOTION_1

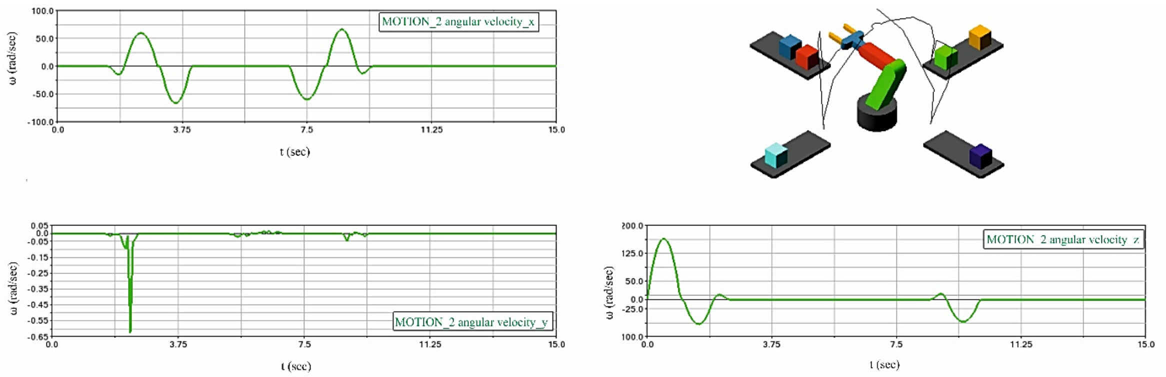


Fig. 8 Angular velocity MOTION_2

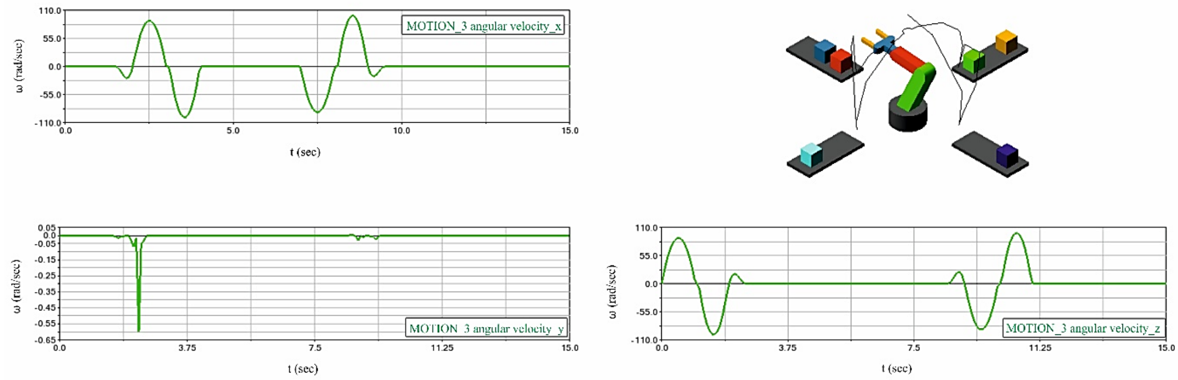


Fig. 9 Angular velocity MOTION_3

The analysis of this time course allows the identification of several characteristics of the dynamic behaviour of the system. These parameters provide valuable information about the response of the system in the uncontrolled state and serve as an initial frame of reference for later comparison with the behaviour of the system after the application of a control intervention. Based on the obtained data, it is possible to more accurately assess the effectiveness and benefit of the applied control actuator, especially in terms of oscillation suppression, rise time reduction and accuracy of the controlled system. A modified model of the robotic arm can be seen in Fig. 10, in which a control mechanism based on a proportional controller (P-controller) has been implemented. This modification was implemented in order to provide basic feedback and to allow more accurate real-time control of the effector motion. Along with the model, output plots showing the progress of two key variables are presented: theta_actual, which represents the actual joint position, and SFORCE_4, which shows the force applied at a specific location of the mechanism. These plots provide a detailed view of the dynamic behaviour of the system during the simulation with the control on.

During the simulation, small changes in the values of the observed variables can be observed (Fig. 11, 12), which arise as a result of the effector's interaction with the manipulated objects – specifically the cubes, which in this case serve as load and test elements. These fluctuations are an expected and natural phenomenon when simulating the mechanical contacts and dynamic forces that occur during the manipulation of physical objects.

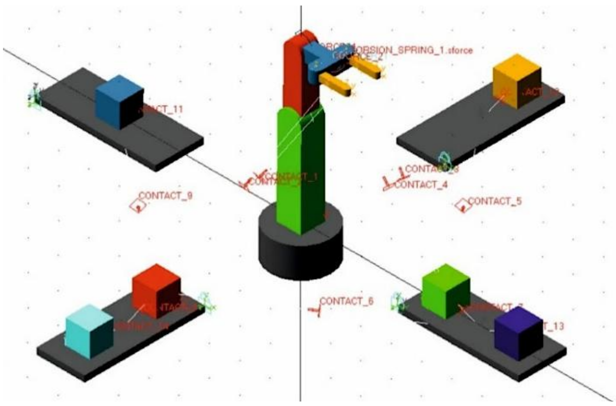


Fig. 10 Manipulator model with added control

Despite these effects, the implementation of a P-controller plays a key role in damping these variations, thus contributing to maintaining the desired trajectory and overall stability of the system throughout the simulation cycle. This demonstrates not only the ability of the system to adapt to external disturbances, but also the effectiveness of the control algorithm used in ensuring the accuracy and reliability of the effector motion.

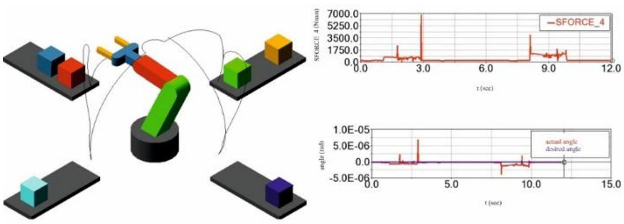


Fig. 11 The current and desired manipulator angle and SFORCE_4 torque at the end position

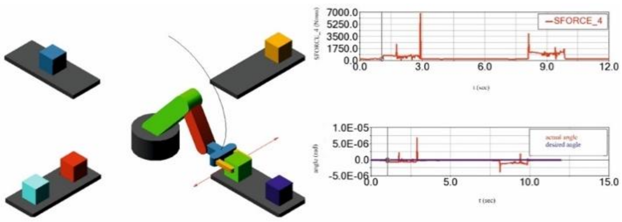


Fig. 12 The current and desired manipulator angle and SFORCE_4 torque when grasping the green cube

CONCLUSION

The aim of this paper was to design and implement a control system for a robotic arm model in a simulation environment, focusing on achieving stable and accurate motion control without using external computational methods such as the Ziegler-Nichols method or the poles and zeros method. The control interventions were implemented exclusively using tools and blocks available directly in the MSC Adams/View environment, demonstrating the ability of this software to handle the basic implementation of feedback control. In the initial phase, a three-dimensional model of the robotic arm with three degrees of freedom was built. The model consisted of a base, three rotary links and an end effector. Subsequently, the actuators were added and the simulation input conditions were defined, which allowed the geometry of the mechanism and its dynamic properties to be precisely adjusted. In the next phase, regulation based on proportional control was implemented. Through iterative testing of different values of the control parameters, the motion trajectory was optimized in order to minimize dynamic fluctuations, reduce settling time and increase the accuracy of reaching the desired positions. The obtained results confirm that effective control of a mechanical system can be realized even without the use of complex analytical methods, directly in the simulation environment. This approach is particularly suitable for the rapid development and testing of control algorithms in the design of various types of mechanisms.

ACKNOWLEDGEMENTS

The authors would like to thank to project KEGA 016TUKE-4/2025 "Development of progressive methods of design of mechatronic systems and their implementation in the study programme Industrial Mechatronics".

REFERENCES

[1] Zheng, Y., Liu, J. & Huang, Q., 2015. Modeling and control of a 3-link robotic manipulator using PID and sliding mode control. *IEEE Transactions on Industrial Electronics*, 62(9), pp. 5618-5627.

[2] Wang, L. & Li, X., 2018. Adaptive and intelligent control for robotic manipulators with unknown dynamics. *Journal of Intelligent & Robotic Systems*, 91(3-4), pp. 423-436.

[3] Popescu, D., Ionescu, C. & Georgescu, M., 2018. Dynamic modeling and control of a 3-DOF robotic manipulator. *Acta Technica Napocensis*, 61(2), pp. 175-182.

[4] Kumar, A. & Singh, P., 2020. Design, simulation, and control of a 3-DOF planar robotic manipulator using Matlab-Simulink. *International Journal of Advanced Robotic Systems*, 17(2), pp. 1-12.

[5] Johnson, M., Smith, R. & Brown, T., 2019. Development of a multi-purpose 3-DOF manipulator system for UAV applications. *Journal of Robotics and Automation*, 35(4), pp. 245-258.

[6] Lee, S., Park, J. & Kim, H., 2020. Implementation of a 3-DOF planar robotic arm for precise manipulation tasks. *Robotics and Autonomous Systems*, 128, 103523.

[7] Dogan, I. (2023) *PID-based Practical Digital Control with Raspberry Pi and Arduino Uno*. Limbricht: Elektor International Media.

[8] Landau, D.I., Lozano, R., Saad, M. and Karimi, A. (2011) *Adaptive Control: Algorithms, Analysis and Applications*. London: Springer.

[9] Kumar, A., Kasera, S. and Prasad, L.B. (2019) 'PD/PID-Fuzzy logic controller based tracking control of 2-link robot manipulator', *i-Manager's Journal on Instrumentation and Control Engineering*, 7(2), pp. 18-25.

[10] Xiao, B., Yin, S., Kaynak, O. (2016) 'Tracking control of robotic manipulators with uncertain kinematics and dynamics', *IEEE Transactions on Industrial Electronics*, 63(10), pp. 6439-6449. doi: 10.1109/TIE.2016.2569068.

[11] Tijani, I.B., Akmeliawati, R., Muthalif, A.G.A. and Legowo, A. (2011) 'Optimization of PID controller for flexible link system using a Pareto-based multi-objective differential (PMODE) evolution', in *Proceedings of the 4th International Conference on Mechatronics (ICOM)*, Kuala Lumpur, Malaysia, 17-19 May 2011. Piscataway, NJ: IEEE, pp. 1-6. doi: 10.1109/ICOM.2011.5937190.

[12] Baek, J., & Kwon, W. (2020). Practical Adaptive Sliding-Mode Control Approach for Precise Tracking of Robot Manipulators. *Applied Sciences*, *10*(8), 2909.

[13] Reyes-Uquillas, D., & Hsiao, T. (2021). Safe and intuitive manual guidance of a robot manipulator using adaptive admittance control towards robot agility. *Robotics and Computer-Integrated Manufacturing*, *70*, 102127.

[14] Prakash, R., Behera, L., Mohan, S., and Jagannathan, S. "Dual-Loop Optimal Control of a Robot Manipulator and Its Application in Warehouse Automation," in IEEE Transactions on Automation Science and Engineering, vol. 19, no. 1, pp. 262-279, Jan. 2022, doi: 10.1109/TASE.2020.3027394.

[15] Azeez, M.I., Abdelhaleem, A.M.M., Elnaggar, S., Moustafa, K.A., & Atia, K.R. (2023). Optimization of PID trajectory tracking controller for a 3-DOF robotic manipulator using enhanced Artificial Bee Colony algorithm. Scientific reports, 13(1), 11164.

[16] Sagova, Z., et al. (2022). 'Study of anisotropic friction in gears of mechatronic systems', *Applied Sciences, 2022*, Vol. 12 (21), 11021. EISSN 2076-3417.

[17] Denavit, J., Hartenberg, R.S. (1955) 'A kinematic notation for lowerpair mechanisms based on matrices', *J. Appl. Mech.*, vol. 22, no. 2, pp. 215-221, Jun. 1955.

[18] Uicker, J.J., Denavit, J., Hartenberg, R.S. (1964) 'An iterative method for the displacement analysis of spatial mechanisms', *J. Appl. Mech.*, vol. 31, no. 2, pp. 309-314, Jun. 1964.

[19] Huczala, D., Kot, T., Pfurner, M., Krys, V., Bobovský, Z. (2022) 'Multirepresentations and multiconstraints approach to the numerical synthesis of serial kinematic structures of manipulators', *IEEE Access. 2022*, vol. 10, pp. 68937-68951.

[20] Grepl, R. (2007) 'Kinematika a dynamika mechatronických systémů', Vysoké učení technické v Brně, 2007.

[21] Song, H., Rong, X., Li, Y., (2017). A co-simulation approach based on ADAMS-SIMULINK for development of manipulator position controller. *ResearchGate*.

[22] Zajačko, I., Gál, T., Ságová, Y., Mateichyk, V., Wiecek, D. (2018) 'Application of artificial intelligence principles in mechanical engineering', *MATEC Web of Conferences 2018.*

[23] Kuether, R.J., Brake, M.R.W., Lamb, D.R. (2020) 'Integrating control systems with multibody dynamics using ADAMS and MATLAB/Simulink', *Mechanism and Machine Theory*, 144, 103121.

[24] Li, Y., Wang, Y. (2022) 'Co-simulation of PID control for robotic arm using ADAMS and Simulink', *International Journal of Advanced Robotic Systems*, 19(3), pp. 1-12.

[25] Åström, K.J. and Hägglund, T. (2006) *Advanced PID Control*. Research Triangle Park: ISA.

[26] Zhang, B., Li, Y., Chen, H. (2021) 'Real-time co-simulation platform for vehicle dynamics and control using ADAMS and Simulink', *Simulation Modelling Practice and Theory*, 108, 102258.

Ľubica Miková (corresponding author)
Department of Industrial Automation and Mechatronics
Faculty of Mechanical Engineering
Technical University of Kosice
Košice, Slovakia
e-mail: lubica.mikova@tuke.sk

Erik Prada
Department of Industrial Automation and Mechatronics
Faculty of Mechanical Engineering
Technical University of Kosice
Košice, Slovakia
e-mail: erik.prada@tuke.sk

Michal Kelemen
Department of Industrial Automation and Mechatronics
Faculty of Mechanical Engineering
Technical University of Kosice
Košice, Slovakia
e-mail: michal.kelemen@tuke.sk