Machinery & Energetics
Vol. 14, No. 4. 2023

Journal homepage:
https://technicalscience.com.ua/en

UDC 621:004.8
DOI: 10.31548/machinery/4.2023.131

Volodymyr Khotsianivskyi*
Postgraduate Student
National Aviation University
03058, 1 Liubomyr Huzar Ave., Kyiv, Ukraine
https://orcid.org/0000-0003-0415-777X

Victor Sineglazov
Doctor of Technical Sciences, Professor
National Aviation University
03058, 1 Liubomyr Huzar Ave., Kyiv, Ukraine
https://orcid.org/0000-0002-3297-9060

Robotic manipulator motion planning method development using neural network-based intelligent system

Abstract. The research relevance is determined by the constant development of industry and the use of robotic manipulators in production processes. The study aims to develop an approach to planning the trajectory of a manipulator robot using an intelligent system based on neural networks. An analysis method, as well as special methods such as design, machine learning, integration strategies, and optimisation techniques, were used to achieve this goal. The main results of the study cover a wide range of achievements in the development of methods for planning the motion of robotic manipulators and their integration into real production conditions. The analysis of existing methods for planning the motion of robotic manipulators and a review of intelligent control systems provided a comprehensive picture of the current state of the art. The developed methods of robot manipulator trajectory identified effective control strategies that consider both dynamic and static scenarios. Training a neural network to plan the optimal path of movement made it possible to detect, track and avoid obstacles in real-time. Hierarchical path planning, adaptive neural network control, genetic algorithms for path optimisation, and dynamic prediction for obstacle avoidance were used to integrate the developed methods into a real production environment. The optimisation and improvement of the created approaches have shown positive results in improving the safety and performance of robotic manipulators, reducing the risk of collisions, and avoiding damage to robots. In addition, the implementation of hierarchical trajectory planning and adaptive neural network control contributed to a significant increase in the accuracy and stability of manipulator movements in various production process scenarios. The practical significance of the study is to develop an intelligent control system and methods for planning the movement of robotic manipulators, which contributes to the efficiency and safety of their operation in real production conditions

Keywords: motion trajectory; industrial automation; mechatronic structures; smart technologies; deep learning models

INTRODUCTION

With the growing use of robotic manipulators in manufacturing, a need for improved methods of planning their movement to ensure the efficiency and accuracy of tasks is growing. The relevance is determined by the existing approaches that often have limitations and are unable to provide flexibility in dealing with dynamic production

Article's History: Received: 13.07.2023; Revised: 05.10.2023; Accepted: 22.11.2023.

Suggested Citation:
Khotsianivskyi, V., & Sineglazov, V. (2023). Robotic manipulator motion planning method development using neural network-based intelligent system. *Machinery & Energetics,* 14(4), 131-145. doi: 10.31548/machinery/4.2023.131.

*Corresponding author

scenarios. The development of an intelligent system based on neural networks opens up prospects for adaptive and autonomous robot motion planning, which is key to increasing productivity and reducing costs in modern production environments.

The research problem is the need to improve planning methods for the robotic manipulator movement in production conditions. Existing approaches often do not provide optimality and adaptability to changing conditions, which can lead to conflicts and unforeseen situations when robots interact in common areas or with dynamic obstacles. The problem also lies in the limited ability of existing robot control systems to respond to changes in the production environment, such as physical deformations of robot components or dynamic changes in the working environment.

Other studies on this topic should be considered for a more thorough analysis. As such, O. Nalobina *et al.* (2022) emphasised that the low robot adoption rate is due to a lack of research on the feasibility and efficiency of their use compared to traditional technologies. The absence of theoretical design frameworks, efficiency assessment methods, and insufficient training emphasises the need for further research in this area, aimed at developing methods for determining the feasibility and training of personnel for the introduction of robotics in agriculture. The authors also analysed the turning movement of the robot and determined its centre of mass speed and the turning radius. V. Khotsyanivskyi and V. Sineglazov (2023) discussed the use of machine learning for the autocalibration of moving elements of robotic systems, in particular, for the control of stepper motors. The described auto-calibration algorithm uses machine learning to improve the accuracy of stepper motors. The authors emphasised that the use of machine learning methods in solving the tasks of calibrating the moving elements of robotic systems can improve their accuracy and efficiency, ensuring more efficient system operation.

M.V. Bogdanovsky (2020) examined the key technical parameters of industrial robots, such as movement speed, load capacity, degrees of mobility, positioning errors, and working area. The author analysed the classification distribution of robots by speed and accuracy to determine their characteristics. He also investigated the impact of the manipulation system's movement on the accuracy of an industrial robot and analysed the types of errors, considering their design features and factors that affect the robot's functioning. The analysis of the study conducted by O. Litvin and S. Pankov (2021) showed limitations of the study on special-purpose robotic manipulators and, as such, this study contributes to the improvement of their systems. Determination of special-purpose robotics and specification of the parameters of new robots are the main tasks for further research and development. As a result of the study, a concept for improving existing and developing new prototypes of robotic mechanisms was set.

P. Koruniak *et al.* (2023) focused on the improvement of equipment, technological processes, and mechanisation, which comprehensively affects product quality and reduces labour costs. The use of industrial robots, including those with remote and automatic control, in flexible automated systems and automation of assembly processes is a promising area. The development of industrial robots equipped with additional coordinate degrees of mobility expands their versatility and technological capabilities but raises new challenges in terms of simplifying structural elements and maintenance systems. In turn, V. Zalypka (2023) developed a new approach to improving the performance of multifunctional manipulators on the platform of multipurpose robotic platforms using Abenics technology. The proposed method allowed the manipulators to perform the functions of both predefined and walking or wheeled propulsion. The author also conducted a three-dimensional modelling and analysis of structural features.

The above studies did not address the issues related to the use of intelligent systems and neural networks for the movement of robotic manipulators. The absence of these aspects requires additional research to develop methods that will effectively use intelligent systems based on neural networks in motion planning for such robots. To solve the stated problem, this study developed methods for planning robot movement using neural networks.

MATERIALS AND METHODS

The study used analytical methods and special techniques: design, machine learning, integration strategies and optimisation methods. To conduct the analysis, a systematic review of scientific publications related to robot manipulator trajectory planning and intelligent control was conducted. The analysis was based on the identification of the advantages and disadvantages of existing approaches, which forms the basis for the development of new methods. In particular, the study included an analysis of the design of algorithms and system architecture to develop effective trajectory planning methods. These trajectory planning methods are aimed at improving the accuracy and speed of manipulators in production environments, providing optimal and efficient solutions to meet the challenges of the industry.

The design method was used to develop proprietary robot trajectory methods for the manipulator. It included careful design of algorithms, development of the optimal system architecture, and creation of highly efficient trajectory planning solutions. It is necessary to note that this method covers not only the technical aspects of the algorithms but also an integrated system that determines the optimal path for the robot to follow. This ensures a high level of productivity and the avoidance of conflicts in the production environment through intelligent motion planning and the selection of optimal paths.

To train the neural network, machine learning methods were used, where the optimal trajectories were used as input data. The main technical aspects of this method include a detailed study of various types of neural network training, such as supervised learning, unsupervised learning, and deep reinforcement learning (DRL) subtasks. In this context, a neural network can predict optimal solutions for manipulator movement, opening up prospects for improving machine learning-based trajectory planning methods and ensuring high efficiency in manipulator control.

Integration strategies were used to effectively combine the developed methods into a single system. The main technical component of this method involves the careful integration of trajectory planning algorithms and the control system. This process is aimed at ensuring their interaction, which leads to the stability and optimal functioning of robotic manipulators in a real production environment. This approach solves the problem of coordination and synchronisation between different stages of the system, increasing the overall productivity and reliability of automated processes.

Optimisation methods were used to systematically improve the efficiency and accuracy of the developed methods. System parameters and algorithms were optimised to achieve optimal results. These approaches have contributed to a significant improvement in the quality of robot trajectory planning and minimisation of the risk of collisions, which is particularly important in production processes. As a result of this optimisation approach, the functionality of the system has been improved, and its suitability for successful application in real-world production conditions has been increased.

The study included a table with a comparative analysis of existing approaches to planning the robotic manipulator motion, namely geometric trajectory planning, inverse kinematics, dynamic programming, random positions, and optimisation methods. Unity3D was chosen to model the movements of the robot manipulator, and MoveIt and UnityRoboticsHub were used for controls. The robot movements via Selective Compliance Articulated Robot Arm (SCARA) were considered. The Rapidly Exploring Random Trees (RRT) interpolation algorithm was used to compare the variables of the joint angle, velocity, and acceleration. Additionally, tables of obstacles in static and dynamic environments, relative errors of the trained model in predicting movement and the time of real movement were used. In addition, the study included the development of our ways to move robotic manipulators, analysis, and description of the architectures of various neural networks, schemes of neural networks for manipulators, recurrent neural networks and long-term memory layers, a schematic diagram of the network for determining controlled influences, and a formula for the space of locally feasible solutions.

The space of locally feasible solutions was calculated using the formula:

$$f_{local}\,(x_k, \phi_k | x_{goal} \,|\, \rightarrow u_k), \qquad (1)$$

where: f_{local} – the space of locally possible solutions; x_k – current robot state; ϕ_k – environment state; x_{goal} – target state; u_k – control command.

RESULTS

Analysis of existing and custom robotic manipulator motion planning methods and review of intelligent control systemscustom

The analysis of traditional methods of manipulator motion planning involves key techniques such as geometric trajectory planning, inverse kinematics, dynamic programming, as well as random positions and optimisation methods. Geometric path planning determines the robot's movement based on the geometric characteristics of the workspace. This method can be used to set the exact position and orientation of the manipulator but can be limited by the complexity of solving problems for complex configurations. In addition, geometric trajectory planning is one of the commonly used methods for industrial robot control. It is typically based on the assignment of high-level commands, such as moving from point to point. While this approach provides user-friendliness, it has its limitations, particularly in the area of adaptation to changing conditions.

The inverse kinematics method is used to determine the input angles or positions of a manipulator to reach a specific position or trajectory. This method is effective in solving problems for specific points in space but can lose accuracy in complex problems due to the large number of possible solutions. It is also used in most industrial robot control systems. The inverse kinematics method can be used to determine the kinematic parameters of a robot based on its position and orientation. Dynamic programming considers the movement of the manipulator as a sequence of actions to minimise criteria. This method is effective for optimisation problems and trajectory planning, particularly in cases where the dynamic constraints of the robot are important. However, it is possible to be computationally expensive for real-time applications in complex environments, especially when there are many dimensions of the solution space and complex problems. Regarding random positions and optimisation methods, these approaches often use random points to reduce the number of intermediate points in trajectories or optimisation techniques to achieve optimal solutions.

To perform a comparative analysis of the above approaches, a table showing their main characteristics and differences was compiled (Table 1).

Table 1. Comparative analysis of existing approaches

Appro ach	Peculiarities	Advantages	Limitations
Geometric trajectory planning	Determines robot movement by workspace parameters	Provides accurate position and orientation data	Difficult to use for complex configurations
		Easy user-manipulation	Limited adaptation to variable conditions

Continued Table 2.

Appro ach	Peculiarities	Advantages	Limitations
Inverse kinematics	Determines input angles or manipulator position to change to target state	Effective in determining specific points	Loss of accuracy in complex tasks
		Used in most movement control systems	High number of possible solutions in complex tasks
Dynamic programming	Implies manipulator movement as a sequence of actions with minimized criteria	Effective for trajectory optimization and planning	Computational-heavy for real-time processing
		Can be used to address robot dynamic restrictions	Hard to use in real-time operation
Random positions and optimization methods	Random-point	Reducing the number of intermediate points in trajectories	Dependency on initial random-point selection
	Optimization methods employment	Achievement of optimal solutions	Need for computing resources, especially for complex tasks

Source: compiled by the authors

Intelligent control systems play a key role in improving and optimising the motion of manipulators. One of the most advanced technologies is the use of neural networks to solve trajectory planning and motion control tasks. Neural networks prove to be effective in solving complex control problems, providing flexibility and adaptability to changing conditions. Deep learning models allow neural networks to automatically learn complex relationships and solve motion planning problems that may be difficult to solve using traditional methods. One of the main advantages of intelligent control systems is their ability to learn and adapt to new conditions. This makes it possible to optimise the operation of manipulators in real-time and increase their productivity. Other types of intelligent systems used to control the motion of manipulators include expert systems, genetic algorithms, and fuzzy logic control systems. Each of these approaches has its advantages and peculiarities when applied to robotics. The integration of intelligent control systems into the motion of manipulators opens up new opportunities for autonomous and adaptive robotic systems. Such systems can effectively adapt to changes in the working environment and perform a variety of tasks with high accuracy.

To achieve the goal of developing an intelligent system based on neural networks for planning the trajectory of a manipulator, it is worth proposing a new approach that considers the key aspects of motion optimisation and control. This approach can include such methods as a hierarchical trajectory planning method, a neural network for adaptive control, a genetic algorithm for trajectory optimisation, and dynamic prediction for obstacle avoidance. The hierarchical trajectory planning method operates on the principle of dividing the original task into successive stages, at each of which the corresponding trajectory is determined and planned. This allows us to divide complex tasks into smaller, more manageable subtasks, which contributes to the effective control of the manipulator's movement in real-time. A neural network for adaptive control is a method that uses a neural network to adaptively control the motion of a manipulator depending on changing environmental conditions. The neural network receives input about the state of the workspace and maintains the optimal path of movement, considering dynamic constraints.

The genetic algorithm for trajectory optimisation creates a population of potential trajectories that evolve to achieve optimal solutions. This allows for a variety of factors to be considered and improves the quality of traffic planning. Dynamic Obstacle Avoidance predicts future environmental conditions and uses this information to adjust the trajectory in real-time. Thus, the developed methods aim to improve the accuracy and speed of planning the trajectory of the manipulator, providing an optimal and adaptive approach to control.

Training a neural network to plan the optimal movement path

When considering the use of neural networks for robot manipulator motion planning systems, it is important to focus on analysing and describing architectures that optimally address the characteristics of these systems. For robot manipulator motion planning systems, the most suitable neural networks are convolutional neural networks, recurrent neural networks, deep neural networks, and autoencoders (Fig. 1).

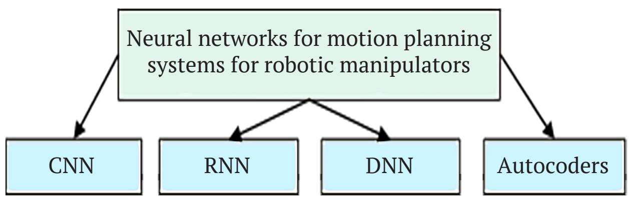

Figure 1. Neural networks for robotic manipulators
Note: CNN – Convolutional neural network; RNN – Recurrent neural network; DNN – Deep neural network
Source: compiled by the authors

CNN is suitable for object recognition. The use of CNNs allows for efficient real-time object recognition, which is key to planning the safe movement of the manipulator around objects in the workspace. The CNN architecture can be modified to include additional layers to accurately determine the position and shape of objects. RNNs are used to model motion dynamics. RNNs are suitable for considering time dependencies and modelling the dynamics of the manipulator's motion, which allows for predicting future states of the system. By applying RNNs to input data that includes previous and current states, the optimal control can be determined accurately. In turn, DNN is best used for trajectory optimisation. The use of DNNs allows optimising the manipulator's trajectories, considering geometric and dynamic constraints. Deep perceptrons can be used to solve complex path-planning problems involving a large number of parameters. For efficient feature extraction, autoencoders can be used. They can help in the efficient extraction of important features from the input data about the manipulator's working environment. The use of autoencoders helps to obtain a latent representation that is useful for analysing objects and obstacles in space.

The functionality of the neural network for determining the trajectory of the manipulator is focused on avoiding collisions and ensuring safe movement. Input data, such as the position and speed of each of the robot's axes and information about obstacles, are processed through the layers of the neural network to identify patterns. The network generates an optimal trajectory, avoiding collisions, by considering geometric constraints and safety limits. The output data, which includes new values for positions and velocities, is used to control the manipulator's motion. The neural network is trained on training data that includes various motion and obstacle scenarios to detect patterns and make the right decisions for a safe trajectory. In addition, it is important to emphasise the advantages of using a neural network to determine a collision avoidance path. These advantages include the ability to consider geometric constraints and the working area of the manipulator, automatic planning of collision avoidance paths, and the ability to adapt to changing environments and different motion scenarios. This ensures the safe and efficient operation of the manipulator in conditions where unpredictable dynamics of changes in its environment are observed.

In general, the analysis and description of these architectures provide an understanding of how neural networks can be adapted to solve specific challenges related to robot motion planning in manufacturing and industrial environments. Although all of these networks are suitable for robot manipulator motion planning, the best is the recurrent neural network. It is used to identify, process, and understand a large amount of visual information, such as analysing dynamic and static scenes. Moreover, RNNs can be used to solve the problem of determining the trajectory to avoid collisions in a manipulator (Fig. 2). The basic concept is that it will be used to model the dynamics of the manipulator's motion, capable of predicting future positions of the robot's axes based on the current state, which is achieved using long short-term memory (LSTM) layers.

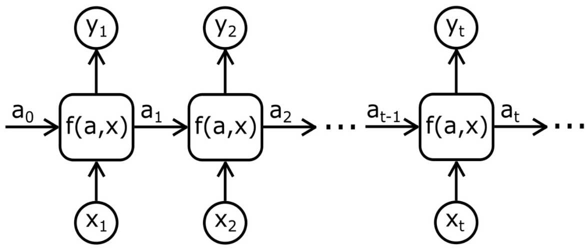

Figure 2. RNN architecture

Source: D. Lazar (2021)

The structure of an RNN consists of repeating blocks that allow information to be transmitted through successive time steps. Each such block contains components such as an input layer, a hidden layer, an output layer, and feedback. Input data, such as the current state of the robot, is entered through the input layer. In the case of trajectory determination, it can be a vector containing information about the positions and velocities of each axis of the manipulator. The hidden layer consists of neurons that perform calculations and transmit information in time steps. This layer can model the dynamics of the manipulator's movement and consider dependencies on previous states. The information from the hidden layer is transmitted to the output layer, which generates the predicted values of the positions of each axis of the manipulator at the next time step. The RNN also includes feedback loops that allow information to be passed back along the time axis. This allows the model to use the information from previous time steps to make more accurate predictions.

As for the LSTM, its structure of manipulator movement consists of an input layer, a repeating block, an output layer, feedback, and trajectory parameterisation (Fig. 3). The input layer accepts a vector containing the current state of the manipulator, including the positions and velocities of each axis. The repetitive block contains an LSTM layer that models the dynamics of the manipulator's movement and transmits information in time steps. The output layer generates the predicted values of the positions of each axis of the manipulator at the next time step. Feedback allows information to be passed from the output layer back to the repeating block to consider the context from previous time steps. Trajectory parameterisation means that the initial values of the positions of each axis can be expressed as trajectory parameters in the form of integer values.

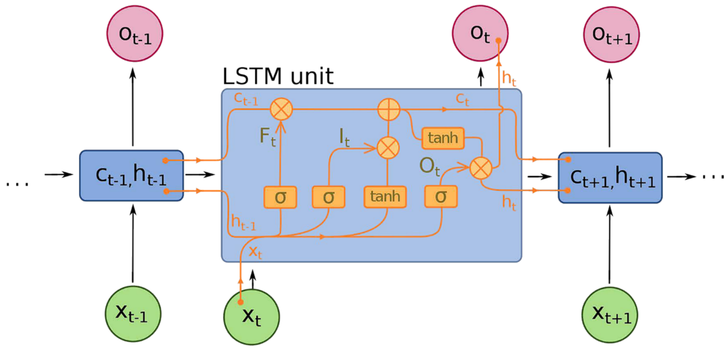

Figure 3. LSTM architecture

Source: M. Zulqarnain *et al*. (2021)

The movements planned via this approach correspond to standard motion commands. After generating the trajectory, it is important to determine the control influence and transform it into the commands required to execute the movement of the robot manipulator. Thus, the actuator control network plays a key role in the motion control system by converting high-level commands into specific control signals. The structure of the controlled action network includes an input layer that receives high-level commands to determine the desired movement; hidden layers that perform calculations and information processing; an output layer that generates controlled actions to implement the movement; feedbacks that adapt the network parameters based on the output signal and movement results; and action parameterisation, where output values can be adapted to meet the requirements of the actuators, such as force or controlled signals (Fig. 4).

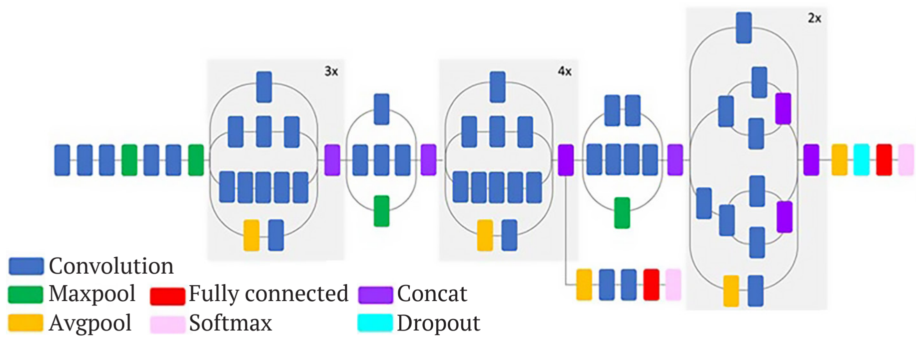

Figure 4. Schematic diagram of the managed impacts identification network

Source: compiled by the authors

The use of a neural network to determine the trajectory of the manipulator helps to increase the safety and efficiency of its operation, minimise the risk of collisions and avoid possible damage to both obstacles and the robot itself.

Ways to integrate the developed methods into a real production environment

The system described above requires a significant amount of data for training and realistic robot movements performed by the control system. However, simulating such movements in a real environment is resource and time-consuming. Thus, the Unity3D environment developed by Unity Software in the United States of America is used to achieve this goal (Fig. 5).

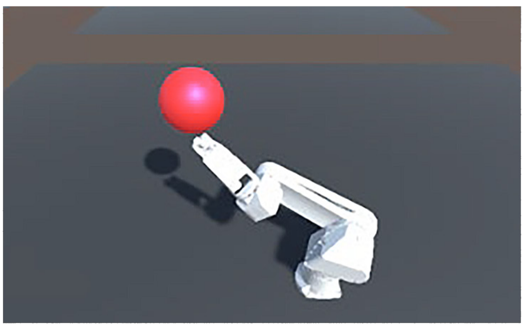

Figure 5. Using the Unity3D environment to model the movements of a robot arm

Source: compiled by the authors

MoveIt and UnityRoboticsHub motion planners were used to control the robot. Some robot movements were generated using high-level motion commands that were programmed manually. To evaluate the proposed approach, two robots will perform the task of picking up and placing objects in different environments that include static and dynamic obstacles (Table 2). It is important to note that, despite the presence of two SCARA robots, motion planning is performed for only one robot, while the other robot is treated as a static or dynamic obstacle.

Table 2. Static and dynamic interference in four environments

Environment type	Static interference	Dynamic interference
Simple static environments	Robot and a cube	None
Complex static environments	Robot and 3 cubes	
Simple dynamic environments	Cube	Movable robot
Complex dynamic environments	3 cubes	

Source: compiled by the authors

Table 2 shows the static and dynamic interference in the four different categories of environments. To evaluate the proposed approach, 100 environments were created that were not used during training. In each environment, 20 pairs of start and target coordinates were randomly generated. The performance of the proposed approach was evaluated in terms of validity, trajectory execution time, and computation time. In an application where the movements of SCARA robots are planned by different planners in an offline mode, the validation was performed only in visual aspects.

The robot can move along an incorrectly generated trajectory, or it can follow the planned trajectory exactly (Fig. 6).

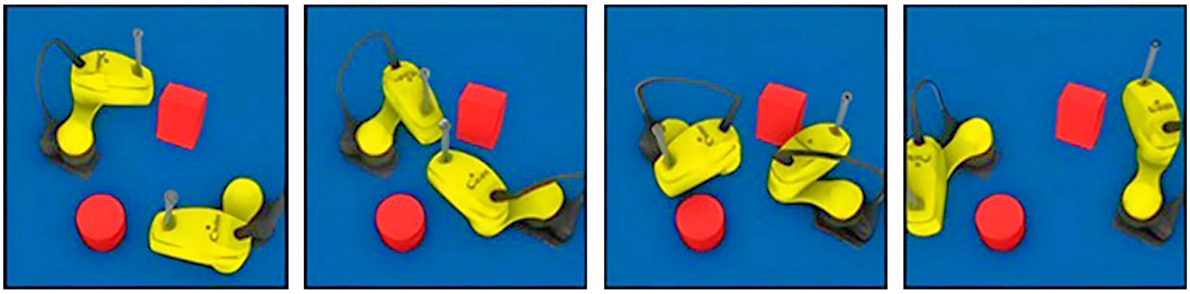

Figure 6. An example of the correct trajectory of the SCARA robot
Source: compiled by the authors

Figure 6 shows that when a robot accurately follows a given trajectory, the other robot moving from the right side crosses the common area earlier than the one moving from the left. Since exploring the full solution space is a complex and scalable task for other challenges, the proposed approach is to start with exploring the local solution space. The locally feasible solution space is the set of all possible control strategies that are constrained only by the state of the local system and guide the robot from the current state to the target area with the help of a specific control command at a certain stage of the search. The task of analysing the environment to determine its characteristics should also be considered, and it is advisable to use a convolutional neural network to solve this problem. In a dynamic scene, objects move or change their state over time. A neural network can be used to detect, track, and determine the speed and direction of movement of these objects. A static scene, on the other hand, describes the state of objects without their movement. The use of a convolutional neural network can be effective for object segmentation in an image, which means highlighting individual objects or regions in a scene.

The YOLOv7 neural network can be used to analyse dynamic and static scenes (Fig. 7). The YOLOv7 object detection algorithm is based on the use of neural networks to recognise objects in images and videos. This algorithm is widely used in the field of computer vision and image processing due to its speed and accuracy in detecting objects. It is safe to assume that YOLO is one of the most popular architectures for object detection and localisation, and YOLOv7 represents a modern modification of this architecture.

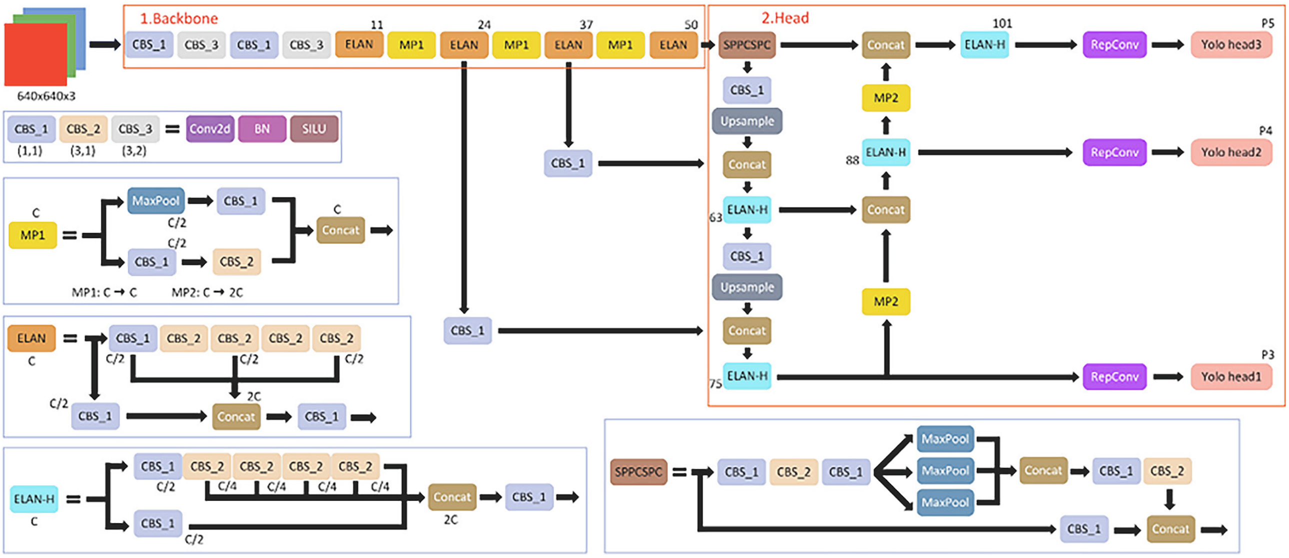

Figure 7. Structure of the YOLOv7 neural network

Source: W. Zhu *et al.* (2022)

YOLOv7 effectively detects objects in images, working with both dynamic and static scenes. Its main advantage is the high speed and accuracy of object detection, allowing it to be used in real-time. To analyse dynamic scenes using YOLOv7, the neural network can be trained on a wide range of videos with moving objects. Once trained, the model can automatically detect and track moving objects in the video, which is useful in video surveillance systems, transport systems, or autonomous vehicles where it is important to analyse moving objects in real-time. As for static scene analysis, YOLOv7 is used to segment and classify objects in an image. Thus, its architecture enables the network to efficiently detect and localise various objects, regardless of their degree of blur or shape, such as cars, people, animals, furniture.

In general, the integration of the developed methods into a real production environment requires a careful approach to ensure the efficiency and reliability of the manipulator. The integrated use of the obtained methods can ensure optimal and reliable planning of the manipulator's trajectory in real production conditions.

Comparison and optimisation of created approaches

After testing in 100 created environments, only 5 trajectories generated by the proposed approach contain errors (Table 3).

Table 3. Relative errors of the trained model when predicting the movement and the execution time of the actual movement

The values of the minimum and maximum possible robot speed, %	Average time error when performing a movement		Average trajectory prediction error	
	Point-to-point movement, %	Linear movement, %	Point-to-point movement, %	Linear movement, %
0-25	2.71	4.4	0.232	0.574
25-50	2.89	5.78	0.473	0.789
50-75	4.38	6.65	0.481	0.862
75-100	6.11	7.17	0.653	0.912

Source: compiled by the authors

Thus, in all scenarios of the experiment, the average discrepancy between the actual and predicted execution time of high-level movement commands is approximately 5%. It is also worth comparing the interpolation algorithm used to convert the planned robot movement into high-level commands, including RRT, with the trajectory generated by the developed system (Fig. 8). It is important to note that the runtime of the robot's trajectory varies significantly depending on the distance between the start and end points of the movement. Thus, for the test cases, it is necessary to classify distances into three categories: small distance (less than 30% of the robot's manipulator range); medium distance (more than 30% but less than 60%); and large distance (more than 60%).

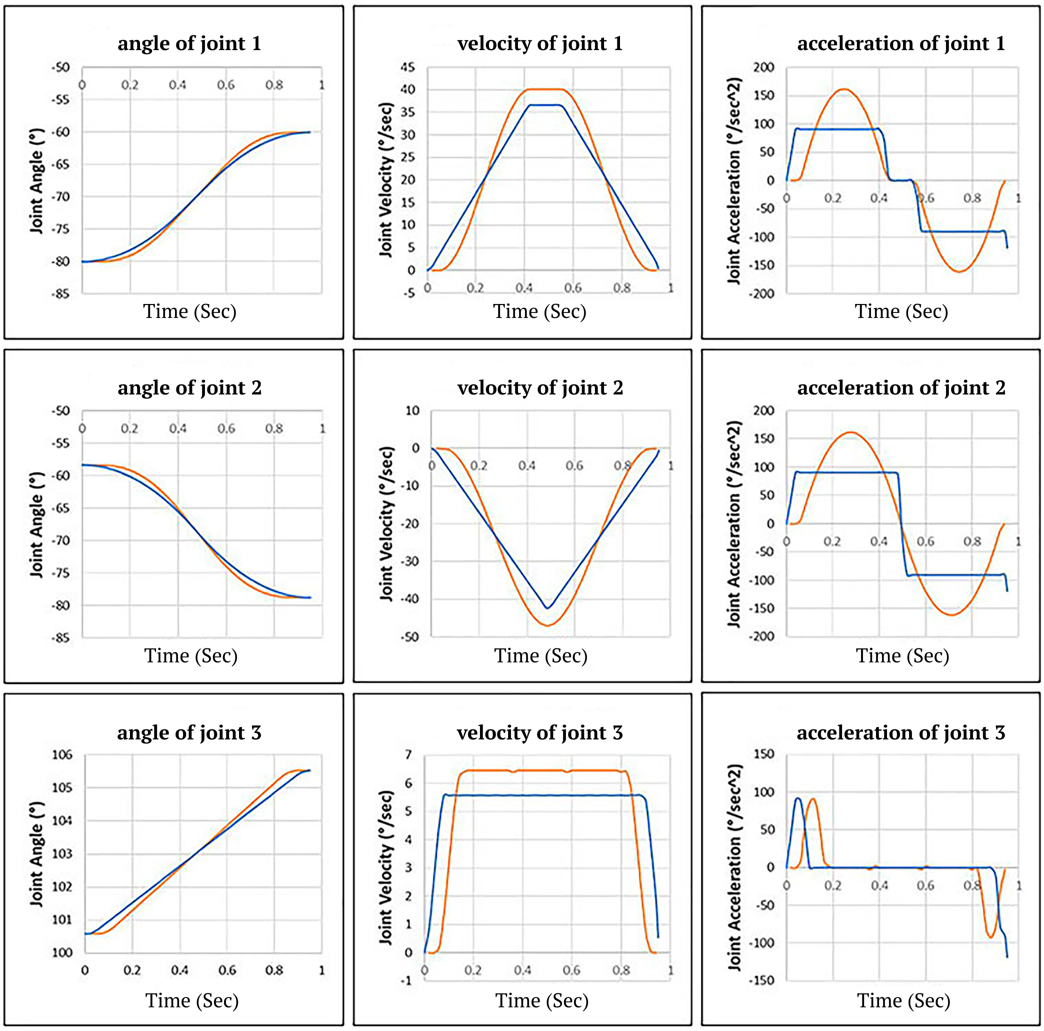

Figure 8. Joint movement at short, medium, and long distances

Note: blue line – joint movement planned via the existing approach; orange line – joint movement performed by the proposed approach

Source: compiled by the authors

Figure 8: "angle of joint 1, 2, 3" – change the angle of rotation of the joint; "velocity of joint 1, 2, 3" – change the velocity of the joint; "acceleration of joint 1, 2, 3" – change the acceleration of the joint. Thus, it is possible to calculate the execution time of the trajectories generated by the approach when using YOLOv7 and RRT (Table 4).

Table 4. Comparative table of the execution time of trajectories generated by the approach using YOLOv7 and RRT

Environment	Distance between start-point and end-point	Average runtime, ms		
		Proposed approach	RRT	Improved RRT
Simple static environments	Low	221	212	213
	Average	422	543	515
	High	659	836	694
Complex static environments	Low	291	372	304
	Average	603	797	663
	High	732	904	756
Simple dynamic environments	Low	244	272	277
	Average	496	581	558
	High	734	958	826
Complex dynamic environments	Low	419	462	465
	Average	765	975	829
	High	1071	1294	1113

Source: compiled by the authors

Therefore, the robot motion planned by the existing system is significantly different from the motion proposed by the trajectory planning system. This is because the RRT control algorithm used in the planning phase is different from the control algorithm described in this paper. In the planning phase, RRT assumes that the joints can reach their maximum acceleration, while the actual robot control system uses only 60% and 45% of the maximum acceleration for the respective robot axes. In general, the developed methods for planning the motion of manipulators already demonstrate significant advantages in solving the problems of obstacle avoidance and trajectory optimisation. However, to achieve an even higher level of efficiency and flexibility, it is worth suggesting several optimisation directions. For example, tuning the hyperparameters of neural networks. Further research and tuning of model parameters will help to achieve an optimal balance between speed and accuracy. Optimising the weights and architecture of the networks can help improve training results and prediction accuracy.

It is necessary to use deep reinforcement learning or DRL. The use of DRL will allow the robot manipulator to learn in real-time, adapting its strategies to new conditions. This can improve the robot's ability to quickly adapt to unpredictable scenarios in a production environment. There is also the integration of additional sensors and real-world data. Incorporating additional sources of information, such as additional cameras or sensors, can improve the robot's perception of its surroundings and provide a more accurate model of the work environment. It is worth paying attention to the improvement of genetic algorithms. Exploring different variants of genetic algorithms and adapting them to the specific requirements of the production process will allow for a better balance between speed and trajectory optimisation capability. Additionally, defining and using clear metrics to evaluate system performance will help to accurately measure the improvements achieved. Metrics can include robot speed, prediction accuracy, and reaction time to changes in the environment. Identifying opportunities includes the use of quantum computing to optimise the large amounts of computation involved in training deep networks and optimising trajectories. These areas of improvement are aimed at expanding the capabilities and improving the motion planning system of manipulators, ensuring that they can effectively adapt to a variety of production process

Based on the study results, several recommendations for practical application in the field of motion planning for manipulative robots emerge. First of all, it is recommended to implement the developed approach in modern production processes that require autonomy and adaptability in the operation of manipulators. It is important to focus on staff training in the use of this technology to optimise workflows. Additionally, it is recommended to conduct other experiments and research aimed at improving the efficiency and speed of manipulative robots in real production conditions. This will expand the scope of the technology and increase its competitiveness. For practical implementation, it is recommended to cooperate with manufacturers of robotic equipment to integrate the proposed approach into new and existing manipulators. This will increase the availability and speed of implementation of this technology on production lines.

DISCUSSION

The study results demonstrate the success of the developed approach to motion planning for manipulative robots. This approach uses an intelligent system and neural networks and is highly accurate and flexible in performing various tasks. For a deeper understanding of this topic, it is worth analysing similar studies conducted by other researchers in this field.

For instance, W. Si *et al.* (2021) investigated composite dynamic movement primitives (DMPs) based on neural networks with radial basis functions for teaching robot skills through human demonstrations. These DMPs can simultaneously encode skills for manipulating position and orientation for transfer from human to robot. Since a robotic manipulator needs to perform tasks under uncertainty, Adaptability to new situations is a requirement. DMPs have been successfully used for various tasks, such as trajectory planning and obstacle avoidance, due to their adaptability to uncertainties and the ability to scale in space and time. In this study, the authors presented a composite DMP framework for simultaneously modelling pose and orientation for robot skill training using neural network techniques. The proposed approach was successfully validated through simulation and experiments. Thus, the common aspects between the present study and the above-mentioned one include the use of neural networks to teach robots movements and manipulation skills. In both cases, the authors aim to achieve robot adaptability in uncertain environments and to train them to perform a variety of tasks. On the other hand, the differences lie in the focus of this study on the use of neural networks for real-time robot manipulator trajectory planning, while the other study focuses more on modelling pose and orientation for teaching specific skills.

In the work of T. Rakhimov *et al.* (2023) emphasised that robots and robotic systems with innovative intelligent control are widely used in industrial production, but their diversity and complexity of control negatively affect working time and productivity. The authors considered a method of position-speed control of a manipulator based on an intelligent mechatronic module. They have developed a block diagram of the control system that calculates the relative speeds of the manipulator's links. They proposed requirements for the control system and methods for eliminating transients during control. They also showed the advantages of time saved and the method of position-speed control of the manipulator by the operator. Common aspects of both studies include an emphasis on the use of innovative technologies in production to improve efficiency. They examine the use of robots and intelligent control systems in industrial production. However, this study focuses on the development of a motion planning method using an intelligent system based on neural networks. Secondly, it focuses on improving the methods of controlling robotic systems in the context of a particular industry.

H. Kalani *et al.* (2017) developed the kinematics of a parallel robot and presented an improved hybrid method for more accurate and faster analysis of the forward kinematics problem (FKP) of parallel manipulators. Artificial neural networks and a 3rd order numerical algorithm are combined to obtain an approximate solution to the FKP problem, which is used as a starting point for the Newton-Raphson numerical technique. Applying this approach to a parallel Stewart-Gough manipulator shows a reduction in the number of iterations to achieve the desired accuracy and a corresponding reduction in FKP analysis time. The result is an innovative algorithm suitable for any rectangular kinematics of serial or parallel robots. Both studies describe innovative technologies to improve the performance of robotic manipulators. However, the research on rectangular kinematics of parallel robots is aimed at optimising motion analysis, in particular, rectangular kinematics analysis. This research focuses on developing ways to move robotic manipulators using an intelligent system based on neural networks.

W.F. Latorre *et al.* (2023) highlighted that a robot manipulator can learn from its environment using the method of reinforcement learning. Each movement of the robot is recorded and compared with past data to determine the best way to achieve the goal in the future. This makes robots more efficient, as they are constantly looking for the best path to follow. At the same time, a trajectory planning method is implemented to control the various movements of the three-stage manipulator in the workspace. The resulting trajectories are stored and used to train the movements, which allows the robot to analyse and compare them to determine the optimal performance of programmed tasks. Both studies aim to improve the movements of robotic manipulators but use different methods to achieve this goal.

Researchers Z. Zhang *et al.* (2023) proposed a new fuzzy recurrent neural network (FRNN) to overcome the internal spurious noise of robots that arises when performing effector tasks, which was developed and implemented on the programmable field of the studied arrays. The pipeline implementation guarantees the sequence of operations, and data processing based on the interaction between clock domains contributes to the acceleration of computing blocks. Compared to traditional gradient neural networks and null neural networks, the proposed FRNN is characterised by a faster convergence rate and more correct operation. In other words, both studies point to the use of innovative approaches in the field of robotics and neural networks to solve important problems. The common aspects include the use of neural networks in robotics to optimise and improve performance. The differences lie in the specifics of the tasks being solved: the above study focuses on improving the robot's internal stability by applying FRNNs to optimise control and overcome spurious noise, while this study aims to develop a method for planning the movement of robotic manipulators using an intelligent system and neural networks.

J. Mężyk (2016) emphasised that industrial manipulators have been replacing humans in production for many years, increasing productivity and product quality. However, the introduction of robotics is often complicated by high costs. The concept of an intelligent tool exchange system offers a solution to this problem. The system enables rapid tool changes by using a decision-making module to self-configure the manipulator, which assesses the situation, identifies the product, and makes decisions about operations and tools. This solution is suitable for variable production and short runs, particularly for small businesses that cannot afford multiple robots. Hence, both studies relate to industrial manipulators and focus on improving production efficiency. However, this study is concerned with the development of a motion planning approach using an intelligent system based on neural networks, while the other focuses on the development of the intelligent system itself for tool exchange and self-configuration of manipulators.

H.S. Kumhar and V. Kukshal (2022) recorded that industrial manipulators play an important role in intelligent manufacturing systems, ensuring high productivity and quality of manufactured products. However, the complex problems of inverse kinematics of industrial robotic manipulators with multiple degrees of freedom require new methods. In this study, a 6-degree-of-freedom manipulator is modelled, and its kinematic parameters are analysed. A CNN is used to solve the inverse kinematics problem, and the network is trained and tested, and performance and training graphs are examined. The results obtained indicate the high efficiency of the method in solving the inverse kinematics problem for a robot with 6 degrees of freedom. Thus, both papers discuss robotic manipulators and intelligent systems, as well as convolutional neural networks. However, this study uses different neural networks. In addition, unlike the other study, there are no specific kinematic parameters.

The archive management robot studied by J. Cui *et al.* (2021) can effectively increase productivity by providing autonomous access to different levels of archives without the need to change their structure or install additional devices. The robot uses a mobile navigation platform a motion algorithm based on laser sensing and a map to move independently. The bionic manipulator simulates the movement of manual access to archives and is attached to the robot's working arm to access different layers of archives. The robot can identify and access files at different levels using an industrial camera to read barcodes. Also, the developed robot was tested in an archival repository, demonstrating efficiency and cost-effectiveness compared to existing solutions for robotic access to archives. In other words, both studies are aimed at improving the movements of robotic manipulators. However, the aforementioned study focuses on the ability of the robot to perform tasks without modifying the structure of the archives, while this study focuses on creating ways to plan the movement of robotic manipulators.

Scientists D. Galvan-Perez *et al.* (2022) worked on the problem of controlling the trajectories of anthropomorphic manipulators in a disturbed environment using output feedback and artificial neural networks. The proposed adaptive robust control scheme uses three-layer artificial neural networks and time-series modelling to improve the efficiency of anthropomorphic manipulator trajectory tracking in high-precision applications such as laser cutting or welding. The system allows the robotic system to autonomously navigate, identify and access archives at different levels without the need to modify the storage or use additional devices. This approach uses neural networks to reduce the dependence on detailed modelling of robotic systems and does not require real-time estimation of uncertain dynamic interference. Both studies address the problems of robotic motion control and neural networks. However, this study focuses on robotic manipulators, while the other study focuses on anthropomorphic manipulators.

Furthermore, X. Wang *et al.* (2023) pointed out that the robotics industry and its applications are playing an important role in modern manufacturing. With the intelligent development of the manufacturing industry, the use of collaborative robots and human-robot cooperation technology is gaining popularity. In a collaborative scenario, in particular, interaction with a human upper limb, the technology of manipulator route planning requires high accuracy. The advanced velocity potential field algorithm proposed in this paper allows for effective collision avoidance and reduces the problem of local oscillations of manipulators near obstacles. The implementation of this algorithm can improve the safety and feasibility of the human-robot collaboration process, making collaborative robots safer and smoother in the work environment. A common aspect between this study and the present one is the active use of robots and robotic solutions to improve manufacturing processes and production efficiency. Both studies discuss technological developments in robotics, addressing the challenges of improving manipulator motion control and developing intelligent systems for robot motion planning. The differences lie in specific aspects of the research, such as the use of neural networks to analyse manipulator movements in this paper and the improvement of velocity potential field algorithms to avoid collisions in human interaction with the other.

J. Yu *et al.* (2023) applied a neural network-based adaptive control strategy for a flexible robot manipulator to track areas in the presence of constraints in the zone. The developed strategy can be used to globally stabilise the robot manipulator and manage model uncertainties and external unknown constraints. However, unlike previous approaches, the use of sliding mode technology and singular perturbation theory in this strategy does not require high-order derivatives of the chain state, which facilitates its implementation and provides greater stability in real-world conditions. Both studies are aimed at developing strategies for controlling robotic manipulators using intelligent technologies such as neural networks. However, while the aforementioned study focuses on adaptive control using neural networks to track areas with zone constraints, this study focuses on developing ways to move robotic manipulators based on an intelligent system based on neural networks.

W. Quan *et al.* (2023) analysed high voltage lines, which are a key element for electricity transmission. They emphasised that the use of electric robots for transmission line maintenance is essential to improve efficiency and safety. They also presented a configuration of a two-armed robot for optimal coordination of movement in various maintenance tasks. It employs polynomial interpolation and matrix laboratory (MATLAB) trajectory simulation techniques, improving the robot's efficiency and performance compared to the one-arm mode. Thus, the common aspects of the research include the use of robots and their movement. At the same time, the main focus of the aforementioned research is on the physical aspects of robot maintenance and coordination. In contrast, this study defines the planning of manipulator movement using intelligent systems, in particular, neural networks.

Scholars M. Suguitan *et al.* (2023) used a method to personalise human-robot interaction by using emotional facial expressions to create affective robot movements. Motion is a key means of expressing the affective states of robots, and using autoencoders to compress motion data and emotional images allows new movements to be generated according to the input. The use of reproduction loss and triplet loss to equalise emotional embeddings allows for affective robot movements. The evaluation was carried out using a user survey, confirming that the generated movements can match the original emotional face images. Both studies employ neural networks, but the difference lies in the use of neural networks for motion planning and interaction for this study and for generating affective expressions in the robot in the other study.

K. Takahashi (2021) considered high-dimensional neural networks based on hypercomplex numbers, such as quaternions, co-quaternions, and others that form a four-dimensional algebra over real numbers. Such networks are investigated for their applicability in the control system of a robotic manipulator. The control system uses the output of a high-dimensional neural network to control the movement of the robot's end-effector in three dimensions. Computational experiments on a long-lasting robotic arm show that the quaternion neural network performs better in learning and control than other networks. Both studies explore the use of neural networks in robotics. The aforementioned study focuses on high-dimensional neural networks based on hypercomplex numbers for controlling a robotic manipulator. This study focuses on an intelligent motion planning system for robotic manipulators using neural networks.

In general, innovative approaches such as the use of high-dimensional neural networks and intelligent motion planning systems play a key role in improving the efficiency and accuracy of robotic manipulator control. All of the above studies, including this one, contribute to the further development of robotic technologies and their implementation in various industries.

CONCLUSIONS

As a result of the study, an approach to planning the movement of manipulative robots using an intelligent system and neural networks was developed. To achieve this goal, analysis methods were used, as well as a special approach that included design methods, machine learning, integration strategies, and optimisation methods. Analysis methods were used to summarise and systematise existing data and approaches in the field of traffic planning. The design aims to create adaptive integrated systems that can effectively interact with the environment. Machine learning was used to train neural networks to ensure their ability to self-regulate and adapt to new conditions. Integration strategies and optimisation methods were applied to improve the efficiency and accuracy of robot motion planning in various scenarios. In other words, this integrated approach allowed us to achieve high autonomy and flexibility in manipulative robot control.

The evaluation of existing methods for planning the motion of manipulators and the analysis of intelligent control systems included the results of a comparative review of geometric path planning, dynamic programming, inverse kinematics, random positions, and optimisation. This review highlighted their characteristics, advantages, and limitations. The developed robot trajectory methods reflected a variety of approaches, such as hierarchical trajectory planning, the use of neural networks for adaptive control, dynamic prediction for obstacle avoidance, and genetic algorithms for trajectory optimisation. The process of training a neural network for optimal route planning has shown the advantages of using networks and architectures such as CNN, RNN, DNN, autoencoders and LSTM, as well as a network diagram to determine the controlled influences. As for the implementation of the developed methods in a real production environment, the use of the Unity3D tool was demonstrated, and static and dynamic interference was considered. An example of the correct trajectory of the SCARA robot is shown and the YOLOv7 structure is presented. The analysis and optimisation of the developed approaches determined the relative errors of the trained model, the average errors of the execution time of real movement, the extreme values of possible robot speeds, as well as the universal table of trajectory execution times, which was conducted using YOLOv7 and RRT.

In general, the developed approach to planning the motion of manipulative robots offers a promising way to achieve high efficiency and flexibility in their use in various production conditions. For further development in the field of motion planning for manipulative robots, it is recommended to study methods for optimising the system's operating parameters, improving machine learning algorithms, and expanding the application of the technology in the field of variable production and robotic systems for various tasks.

ACKNOWLEDGEMENTS

None.

CONFLICT OF INTEREST

None.

REFERENCES

[1] Bogdanovsky, M.V. (2020). Analysis of the main errors and assessment of the accuracy of industrial robots. *Technical Engineering*, 2(86), 67-72. doi: 10.26642/ten-2020-2(86)-67-72.

[2] Cui, J., Cui, L., & Jiang, H. (2021). Archive access robot for smart archive repositories. *Industrial Robot*, 49(4), 745-759. doi: 10.1108/IR-08-2021-0171.

[3] Galvan-Perez, D., Yañez-Badillo, H., Beltran-Carbajal, F., Rivas-Cambero, I., Favela-Contreras, A., & Tapia-Olvera, R. (2022). Neural adaptive robust motion-tracking control for robotic manipulator systems. *Actuators*, 11(9), article number 255. doi: 10.3390/act11090255.

[4] Kalani, H., Akbarzadeh, A., Moghimi, S., & Khoshraftar, N. (2017). Forward kinematics solution of stewart-gough using improved hybrid strategy (neural network and 3rd-order Newton-Raphson). *Journal of Computational Methods in Engineering*, 35(2), 113-129. doi: 10.18869/acadpub.jcme.35.2.113.

[5] Khotsyanivskyi, V., & Sineglazov, V. (2023). Machine learning in the task of auto-calibration of moving elements of robotic systems on the example of stepper motor control. In *International scientific and technical conference "AVIA"* (pp. 9.37-9.41). Kyiv: National Aviation University.

[6] Koruniak, P., Nishchenko, I., & Sheremeta, P. (2023). Vibration mobile devices in robotic systems. *Bulletin of Lviv National Environmental University. Series Agroengineering Research*, 26, 22-29. doi: 10.31734/agroengineering2022.26.022.

[7] Kumhar, H.S., & Kukshal, V. (2022). Inverse kinematic solution for 6-r industrial robot manipulator using convolution neural network. In *Select proceedings of IPDIMS 2021 "Recent trends in product design and intelligent manufacturing systems"* (pp. 923-930). Singapore: Springer. doi: 10.1007/978-981-19-4606-6_84.

[8] Latorre, W.F., Castro, F.C., Caviativa, Y.P., Amaya, J.C., & Sanz, F.A. (2023). Implementation of a reinforced learning algorithm in a simulation environment for path planning of a robot manipulator with 3 degrees of freedom. In *Proceedings of the 10th workshop on engineering applications "Applied computer sciences in engineering"* (pp. 151-162). Cham: Springer. doi: 10.1007/978-3-031-46739-4_14.

[9] Lazar, D. (2021). *Creating a simple RNN from scratch with TensorFlow*. Retrieved from https://medium.com/nabla-squared/creating-a-simple-rnn-from-scratch-with-tensorflow-8995a03c976d.

[10] Litvin, O., & Pankov, S. (2021). Robotic manipulators special purpose. *Technical Sciences and Technology*, 1(19), 81-88. doi: 10.25140/2411-5363-2020-1(19)-81-88.

[11] Mężyk, J. (2016). A concept for intelligent tool exchange system for industrial manipulators. *Solid State Phenomena*, 251, 158-163. doi: 10.4028/www.scientific.net/SSP.251.158.

[12] Nalobina, O., Holotiuk, M., Bundza, O., Shymko, A., & Mikhailov, A. (2022). The task of an agricultural robot during the passing of turns. *Advances in Mechanical Engineering and Transport*, 2(19), 141-147. doi: 10.36910/automash.v2i19.912.

[13] Quan, W., Zou, D., Xi, C., Qiao, M., Liu, L., & Jiang, W. (2023). Simulation research on trajectory planning of double-arm cooperative live working robot for transmission lines. *Journal of Physics Conference Series*, 2433, article number 012013. doi: 10.1088/1742-6596/2433/1/012013.

[14] Rakhimov, T., Erkinov, S., & Takhirova, G. (2023). Positional-velocity control of the manipulator built on the basis of an intelligent mechatron module. *E3S Web of Conferences*, 452, article number 03011. doi: 10.1051/e3sconf/202345203011.

[15] Si, W., Wang, N., & Yang, C. (2021). Composite dynamic movement primitives based on neural networks for human-robot skill transfer. *Neural Computing and Applications*, 35(32), 23283-23293. doi: 10.1007/s00521-021-05747-8.

[16] Suguitan, M., DePalma, N., Hoffman, G., & Hodgins, J. (2023). Face2Gesture: Translating facial expressions into robot movements through shared latent space neural networks. *ACM Transactions on Human-Robot Interaction*. doi: 10.1145/3623386.

[17] Takahashi, K. (2021). Comparison of high-dimensional neural networks using hypercomplex numbers in a robot manipulator control. *Artificial Life and Robotics*, 26(3), 367-377. doi: 10.1007/s10015-021-00687-x.

[18] Wang, X., Wu, Q., Wang, T., & Cui, Y. (2023). A path-planning method to significantly reduce local oscillation of manipulators based on velocity potential field. *Sensors (Basel, Switzerland)*, 23(23), article number 9617. doi: 10.3390/s23239617.

[19] Yu, J., Wu, M., Ji, J., & Yang, W. (2023). Neural network-based region tracking control for a flexible-joint robot manipulator. *Journal of Computational and Nonlinear Dynamics*, 19(2), article number 021003. doi: 10.1115/1.4064201.

[20] Zalypka, V. (2023). Simulation of the stress-deformed state of the elements of the multifunctional manipulator of multipurpose robotic platforms. *Advances in Mechanical Engineering and Transport*, 2(21), 101-109. doi: 10.36910/automash.v2i21.1214.

[21] Zhang, Z., He, H., & Deng, X. (2023). An FPGA-implemented antinoise fuzzy recurrent neural network for motion planning of redundant robot manipulators. In *IEEE transactions on neural networks and learning systems* (pp. 1-13). Piscataway: IEEE. doi: 10.1109/TNNLS.2023.3253801.

[22] Zhu, W., Wang, Q., Luo, L., Zhang, Y., Lu, Q., Yeh, W.-C., & Liang, J. (2022). CPAM: Cross patch attention module for complex texture tile block defect detection. *Applied Sciences*, 12(23), article number 11959. doi: 10.3390/app122311959.

[23] Zulqarnain, M., Khalaf Zager Alsaedi, A., Ghazali, R., Ghouse, M.G., Sharif, W., & Aida Husaini, N. (2021). A comparative analysis on question classification task based on deep learning approaches. *PeerJ Computer Science*, 7, article number e570. doi: 10.7717/peerj-cs.570.

Володимир Петрович Хоцянівський
Аспірант
Національний авіаційний університет
03058, просп. Любомира Гузара, 1, м. Київ, Україна
https://orcid.org/0000-0003-0415-777X
Віктор Михайлович Синєглазов
Доктор технічних наук, професор
Національний авіаційний університет
03058, просп. Любомира Гузара, 1, м. Київ, Україна
https://orcid.org/0000-0002-3297-9060

Розробка методу планування руху роботів маніпуляторів з використанням інтелектуальної системи заснованої на нейронних мережах

Анотація. Актуальність досліджуваної проблеми випливає з постійного розвитку промисловості та використання роботів маніпуляторів у виробничих процесах. Метою дослідження є розробка підходу до планування траєкторії робота маніпулятора за допомогою інтелектуальної системи на основі нейронних мереж. Для досягнення поставленої мети використано метод аналізу, а також спеціальні методи, такі як проектування, машинне навчання, інтеграційні стратегії та оптимізаційні методи. Основні результати дослідження охоплюють широкий спектр досягнень у розвитку методів планування руху роботів маніпуляторів та їхньої інтеграції в реальні виробничі умови. Аналіз існуючих методів планування руху роботів маніпуляторів та огляд інтелектуальних систем управління надали вичерпну картину сучасного стану цієї проблематики. Розроблені методи траєкторії робота маніпулятора визначили ефективні стратегії управління, які враховують як динамічні, так і статичні сценарії. Навчання нейронної мережі для планування оптимального шляху переміщення зробило можливим виявлення, відстеження та уникнення перешкод в реальному часі. Для способів інтеграції розроблених методів в реальне виробниче середовище використано ієрархічне планування траєкторії, адаптивне керування нейронною мережею, генетичні алгоритми для оптимізації траєкторії та динамічне прогнозування для уникнення перешкод. Оптимізація та вдосконалення створених підходів показали позитивні результати у покращенні безпеки та продуктивності роботів маніпуляторів, зниженні ризику зіткнень та уникненні пошкоджень роботів. Крім того, реалізація ієрархічного планування траєкторії та адаптивного керування нейронною мережею сприяла значному підвищенню точності та стабільності рухів маніпуляторів в різних сценаріях виробничого процесу. Практичне значення дослідження полягає у розробці інтелектуальної системи управління та методи планування руху роботів маніпуляторів, що сприяє підвищенню ефективності та безпеки їхньої роботи в реальних виробничих умовах

Ключові слова: траєкторія пересування; промислова автоматизація; мехатронічні структури; розумні технології; моделі глибокого навчання