



ARTICLE

Obstacle Avoidance Path Planning for Delta Robots Based on Digital Twin and Deep Reinforcement Learning

Hongxiao Wang¹, Hongshen Liu¹, Dingsen Zhang^{1,*}, Ziyi Zhang¹, Yonghui Yue¹ and Jie Chen²

¹College of Information Science and Engineering, Northeastern University, Shenyang, 110819, China

²Xufeng Electronics Co., Ltd., Shenyang, 110819, China

*Corresponding Author: Dingsen Zhang. Email: zhangdingsen@ise.neu.edu.cn

Received: 30 October 2024; Accepted: 05 February 2025; Published: 16 April 2025

ABSTRACT: Despite its immense potential, the application of digital twin technology in real industrial scenarios still faces numerous challenges. This study focuses on industrial assembly lines in sectors such as microelectronics, pharmaceuticals, and food packaging, where precision and speed are paramount, applying digital twin technology to the robotic assembly process. The innovation of this research lies in the development of a digital twin architecture and system for Delta robots that is suitable for real industrial environments. Based on this system, a deep reinforcement learning algorithm for obstacle avoidance path planning in Delta robots has been developed, significantly enhancing learning efficiency through an improved intermediate reward mechanism. Experiments on communication and interaction between the digital twin system and the physical robot validate the effectiveness of this method. The system not only enhances the integration of digital twin technology, deep reinforcement learning and robotics, offering an efficient solution for path planning and target grasping in Delta robots, but also underscores the transformative potential of digital twin technology in intelligent manufacturing, with extensive applicability across diverse industrial domains.

KEYWORDS: Digital twin; deep reinforcement learning; delta robot; obstacle path planning

1 Introduction

Industrial robotics is crucial for improving productivity and product quality in industries that require high precision and speed, such as microelectronics, medical and pharmaceutical, and food packaging. Delta robots are especially suited for high-speed pick-and-place tasks, due to their high-speed motion and high-precision gripping capabilities [1]. Therefore, efficient and reliable path planning is critical. However, traditional path planning methods often prove inefficient in complex and dynamic environments. They struggle to effectively respond to unexpected obstacles and are costly to test and debug in real industrial settings due to safety risks.

Deep reinforcement learning, as a powerful machine learning method, has made significant progress in the field of robot control and path planning in recent years. It is capable of learning the optimal policy by interacting with the environment, effectively solving path planning problems in complex scenarios. Yang et al. designed a hierarchical controller in a V-rep simulation environment and utilize the deep reinforcement learning method to complete the navigation and motion control tasks for obstacle avoidance, which achieved promising results in the tests [2].

In recent years, digital twin technology, as an emerging modeling and simulation technology, has provided new ideas to solve the above problems. Digital twin models are characterized by high accuracy



and dynamic consistency, which facilitates effective visualization and monitoring of the operational state of physical entities. They also facilitate the efficient collection and transmission of virtual and real data, while supporting real-time interaction and control [3–5]. In addition, digital twin technology integrates various models and data types to ensure that they operate under a consistent and efficient framework, which enables reliable simulation results to be applied to production in real time and helps operators and planners to make decisions [4]. Based on the above advantages, digital twin technology is widely used in the applications in smart grids [6–9], healthcare [10–12], quality control [13], agriculture [14], industry, food [15], and other domains. Digital twin technology provides an ideal virtual environment for robot path planning. Combining digital twin technology with deep reinforcement learning can effectively improve the efficiency and safety of robot path planning and reduce the risk in real-world environments.

However, applying digital twin technology to real industrial scenarios still faces many challenges. Qinglei Zhang et al. developed a digital virtual simulation platform for the welding process and welding environment, which realized the efficient collaboration between the robot and the positioning machine during the welding process and significantly improved the operational efficiency. However, the real-time data delay caused by network communication during the welding process still needs further research [16]. Yang et al. designed a digital twin system for intelligent management of cell-level wheeled mobile robots to achieve high-fidelity modeling and emphasized the importance of realizing real-time bidirectional closed-loop interactions between the physical and virtual spaces [17].

In this paper, we have constructed a digital twin system for the Delta robot and proposed a DQN-based deep reinforcement learning algorithm with an improved reward mechanism. This algorithm solves the path planning and object-grasping problems for the Delta robot. We also conducted communication and interaction experiments between the digital twin robot and the physical robot, which yielded promising results. These findings demonstrate a potential application of digital twin technology in industrial settings, emphasizing its significant promise and wide-ranging applicability.

The innovations of this paper are as follows:

- (1) A digital twin system for Delta robots is constructed, designed to be applicable in real-world scenarios in this work.
- (2) A deep reinforcement learning algorithm for path planning of Delta robots based on the digital twin system is designed, and the learning efficiency is enhanced by an improved intermediate reward mechanism.

Section 2 will provide a detailed explanation of the architecture and modeling of the Delta Robot digital twin system. Section 3 focuses on the key modeling processes of the system, including robot kinematics analysis, visual perception system development, reinforcement learning environment construction, and algorithm design. Section 4 will present the experimental results, including the convergence analysis of the reinforcement learning algorithm and the experimental results of the digital twin model working with the physical robot. These results validate the algorithm's effectiveness in obstacle avoidance path planning for the Delta robot and demonstrate the strong coordination between the digital twin model and the physical system. Section 5 summarizes the full paper and looks at future research directions.

2 Construction of the Digital Twin System

2.1 Establishing the Framework for the Delta Robot Digital Twin System

While digital twin technology has been applied in various fields such as shipbuilding and power plants successfully [4,18], its application in Delta robot systems remains largely unexplored. This paper proposes a digital twin model for Delta robots, as shown in Fig. 1. The system comprises a digital twin component and a physical robot component. The digital twin component includes a Delta robot model, a robot operation

model, and a path planning module. The physical robot component includes a physical robot prototype, a robot operation unit, and an obstacle avoidance system. Data interaction between the digital twin and the physical robot is achieved via the OPC UA communication protocol, specifically:

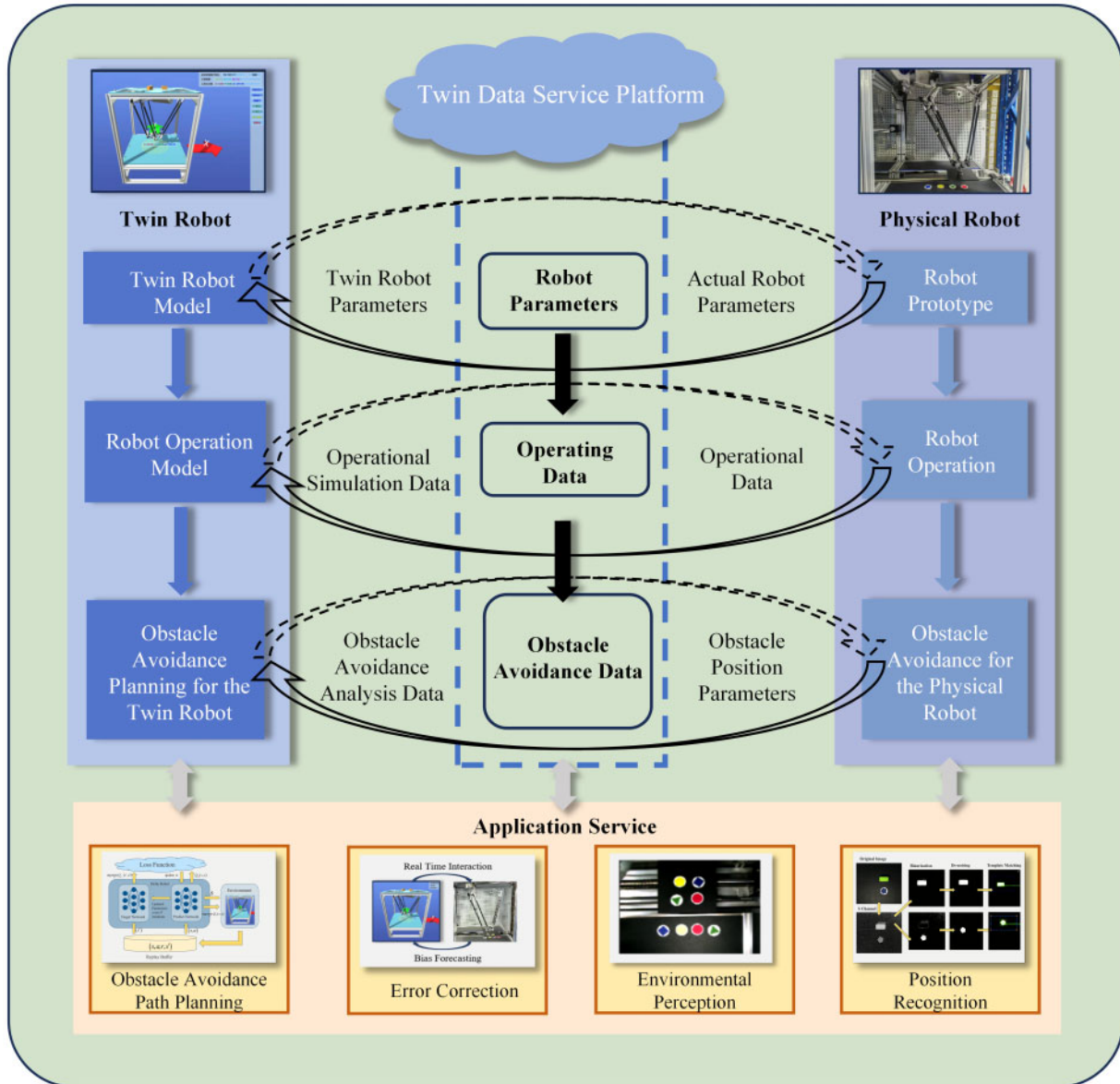


Figure 1: Digital twin Delta robot system model

(1) **Model Parameter Interaction:** The digital twin model and the physical robot prototype exchange robot parameters in real-time, dynamically adjusting the digital twin model.

(2) **Operational State Interaction:** The digital twin operation model and the physical robot operation unit exchange operational parameters in real-time, enabling real-time tracking of the physical robot's movement.

(3) **Path Planning and Obstacle Avoidance System Interaction:** The digital twin path planning module and the physical robot obstacle avoidance system exchange information in real-time. The physical robot

system provides the digital twin system with target position and obstacle position information, and the digital twin system utilizes a reinforcement learning algorithm to plan obstacle avoidance paths and transmit them to the physical robot.

The system offers multiple service functionalities, including obstacle avoidance path planning, error correction, environmental perception, and position recognition.

This Delta robot digital twin system is primarily composed of four layers, as shown in Fig. 2: data layer, model layer, control layer, and function layer [19].

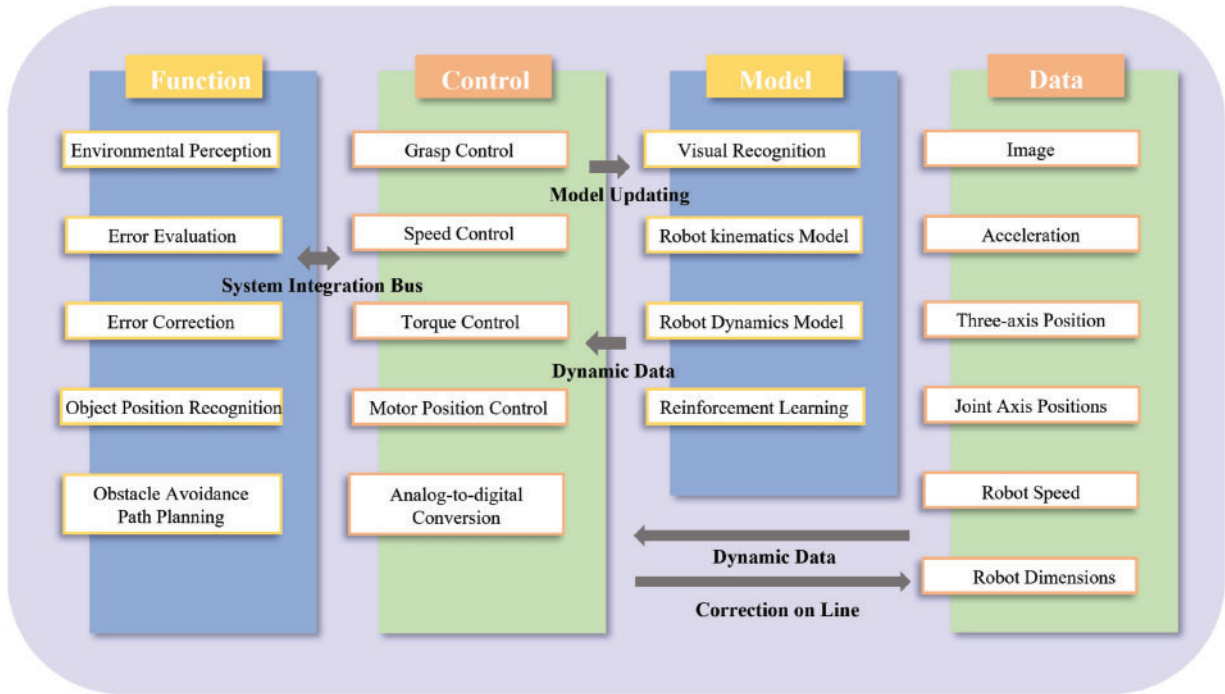


Figure 2: Digital twin delta robot framework

Data Layer: The data layer forms the foundation of the digital twin system's functionality. It transmits data to the model and control layers and receives instructions from the control layer. This layer collects real-time data and images from sensors and cameras on the physical robot, including image data, three-axis position, robot speed, acceleration, joint positions, and robot dimensions.

Model Layer: The model layer is the core component responsible for realizing the digital twin's functionality. It encompasses kinematic models, dynamic models, visual recognition models, and reinforcement learning models of the robot. The accurate construction of these models ensures the accuracy and realism of the digital twin, enabling its various functionalities.

Control Layer: The control layer manages various control tasks, including motor position control, speed control, torque control, gripping control, and analog-to-digital conversion. Utilizing data from the data and model layers, it constructs a closed-loop control system to achieve high-precision motion control of the Delta robot.

Function Layer: The function layer is the core of the Delta robot digital twin system, relying on the collaborative operation of the data, model, and control layers. Its main functions include environmental perception, object position recognition, obstacle avoidance path planning, error assessment and correction.

Environmental perception and object position recognition act as the robot's "eyes", identifying target workpieces, key positions, and obstacles. Error assessment and correction evaluate the accuracy of workpiece placement and make necessary adjustments. The core functionality of this system is obstacle avoidance path planning, ensuring the robot can effectively plan its path during material handling to avoid collisions.

2.2 Establishment of Digital Twin-Related Models

2.2.1 Kinematic Modeling of the Delta Robot Digital Twin

Based on the physical structure of the Delta robot shown in Fig. 3, a kinematic model is established. Forward kinematics modeling is the process of calculating the position of the Delta robot's end effector in space based on the rotation angles of the three motors (or joints). By analyzing the geometric relationships of various parameters such as the lengths of the Delta robot's links and the joint angles, a mathematical expression for the position of the end effector can be derived, as shown in Eqs. (1) and (2) [20], which will play an important role in subsequent path planning.

$$\begin{cases} x = x_1 + f_x z \\ y = f_2 + f_y z \end{cases} \quad (1)$$

$$z = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A} \quad (2)$$

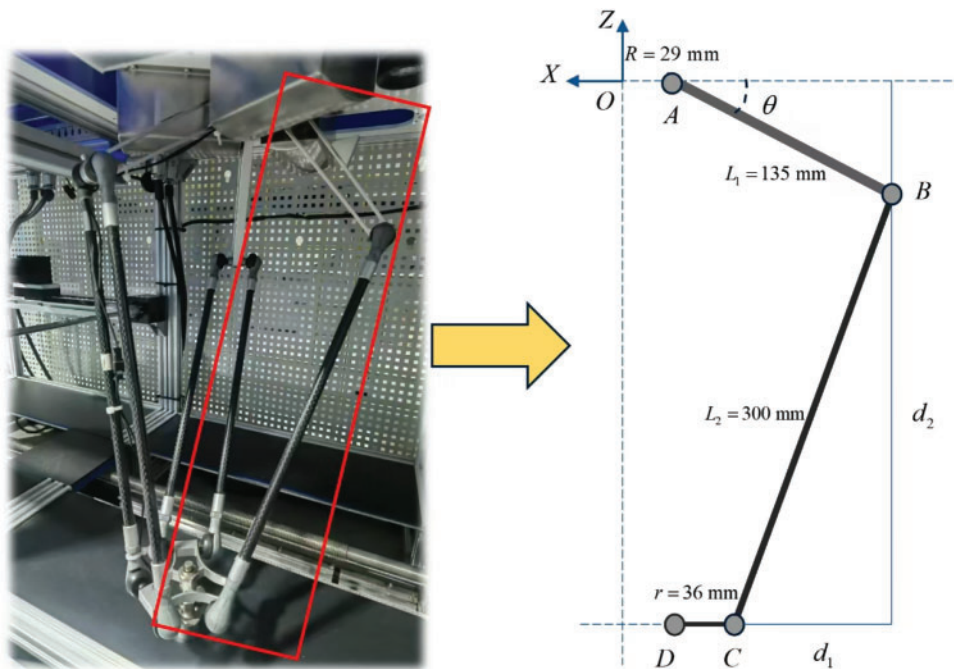


Figure 3: Establishment of the Delta robot model

On the other hand, inverse kinematics modeling calculates the angles needed to drive the three motors to achieve the desired spatial position of the Delta robot's end effector. In the digital twin system, inverse

kinematics can be combined with real-time sensor data, allowing the robot to perform effective path adjustments and optimizations in a dynamic environment, ensuring real-time responsiveness of its actions.

2.2.2 Visual Perception Model

The Delta robot digital twin system utilizes visual technology to identify the position of workpieces, target positions, and obstacles. Additionally, it evaluates the effectiveness of the workpiece placement.

A. Camera Calibration

In this paper, the nine-point calibration method is used to convert pixel coordinates into robot coordinates. The “eye-to-hand” configuration is adopted, where the camera is positioned externally to the robot’s hand and placed on the upper section of the workbench during the setup of the experimental apparatus. The main formulas for the nine-point calibration method are shown as Eq. (3) [21]:

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} a & b \\ a' & b' \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} + \begin{bmatrix} c \\ c' \end{bmatrix} \quad (3)$$

In this context, $\begin{bmatrix} X' \\ Y' \end{bmatrix}$ represents the robot coordinates, $\begin{bmatrix} a & b \\ a' & b' \end{bmatrix}$ denotes the rotation coefficient, $\begin{bmatrix} X \\ Y \end{bmatrix}$ refers to the pixel coordinates, and $\begin{bmatrix} c \\ c' \end{bmatrix}$ indicates the displacement coefficient. It can be observed that there are six unknown parameters in the entire transformation formula. Therefore, six coordinate points are sufficient to fully solve the transformation equation. However, to minimize errors, we opt to use nine points for camera calibration.

B. Image Processing

To ensure that the camera correctly identifies the coordinates of the object to be grasped and its corresponding target coordinate point, image transformation is necessary for the raw image captured by the camera [22]. First, the raw image undergoes an HSV transformation to extract the S-channel parameters. Next, a reasonable threshold is set based on the target color for binarization. Afterward, noise is removed from the image, followed by template matching. The corresponding pixel coordinates are then converted into robot coordinates using the transformation formula derived from the camera calibration process in the previous section. These coordinates are transmitted to the digital twin system and subsequently forwarded to the PLC system. The flowchart is shown in Fig. 4.

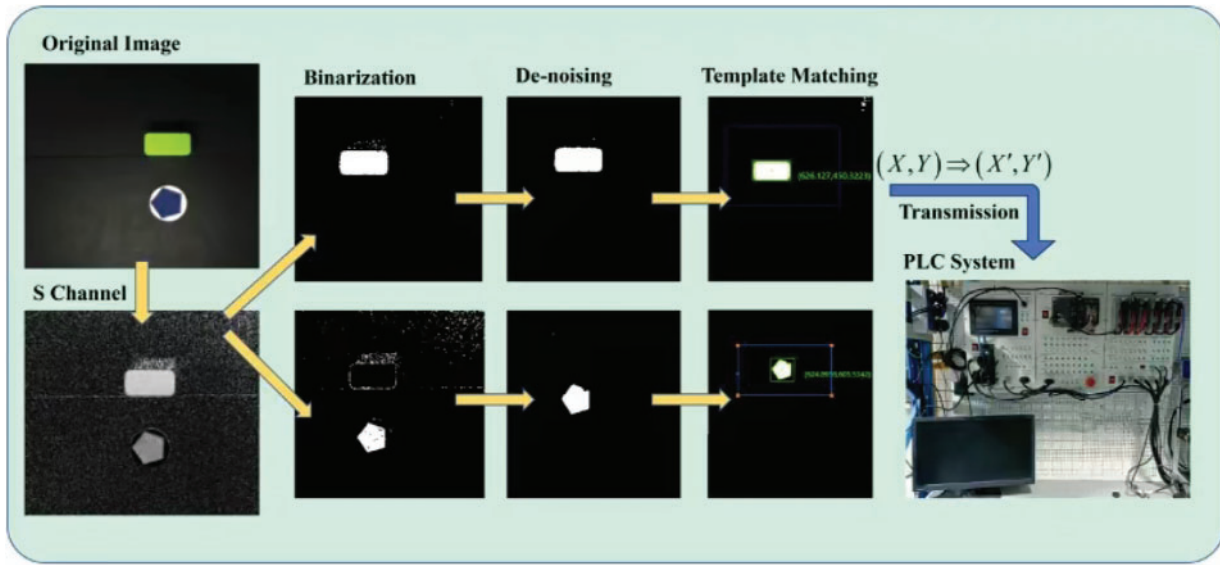


Figure 4: Image processing flowchart

3 Delta Robot Path Planning Based on Digital Twin

RL is a method in which an agent learns the optimal strategy through real-time interaction with its environment. It can be used in robotic systems to help implement path planning [23]. During the process of algorithm debugging for the Delta robot system, we found that the careful design of reinforcement learning elements and a reasonable reward mechanism can significantly improve learning efficiency. In this chapter, we focus on the design of reinforcement learning elements for the Delta robot digital twin system and introduce the intermediate reward mechanism we proposed, which is specifically tailored for the Delta robot system.

3.1 Design of Reinforcement Learning Elements

The research on obstacle avoidance planning for the Delta robot based on deep reinforcement learning in this paper is described as Markov Decision Process. The main elements of this process, including the action space, state space, and reward function.

(1) Action Space:

The Delta robot in this study is a three-axis parallel robot with three degrees of freedom. When defining the action space, each axis of the robot is set to rotate by 1 degree, 0 degrees, or -1 degree from its current position. For example, an action vector could be represented as $[0, 1, 1]$, indicating that the robot's 0-axis remains stationary, the 1-axis rotates by 1 degree, and the 2-axis also rotates by 1 degree. Clearly, with this rule, each axis has three possible states, resulting in a total of 27 possible actions in the action space.

(2) State Space:

The state space is defined by the current angles of the robot's three axes n_0, n_1, n_2 , as well as the three dimensional coordinates x, y, z of the moving platform's center point, which are obtained through the robot's forward kinematics solution. This can be expressed as in Eq. (4):

$$s = [n_0, n_1, n_2, x, y, z] \quad (4)$$

(3) Reward Function:

In the obstacle avoidance planning for the Delta robot, a sparse reward system is first established, categorized into three situations:

- The robot's end effector reaches the target area;
- The moving platform collides with an obstacle;
- The moving platform moves outside the predefined workspace.

These can be expressed as Eq. (5):

$$R_{sp} = \begin{cases} r_a & \text{reach the target} \\ r_b & \text{crash} \\ r_c & \text{move outside the work space} \end{cases} \quad (5)$$

However, it was found during the experiment that using only sparse rewards led to low training efficiency, which fell far short of our practical application requirements. Therefore, we introduced an intermediate reward during the robot's exploration process to guide the robot closer to the target area. The intermediate reward function is shown in Eqs. (6) and (7) [24]:

$$R_{sp} = \begin{cases} \omega \frac{\sqrt{(x_0 - x_d)^2 + (y_0 - y_d)^2 + (z_0 - z_d)^2}}{\sqrt{(x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2}} & \varphi > 1 \\ \omega \sqrt{(x_0 - x_d)^2 + (y_0 - y_d)^2 + (z_0 - z_d)^2} & \text{else} \end{cases} \quad (6)$$

$$\varphi = \sqrt{(x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2} \quad (7)$$

In this context, x, y, z represent the current three-dimensional coordinates of the moving platform's center position, x_0, y_0, z_0 represent the three-dimensional coordinates of the starting position of the moving platform's center, and x_d, y_d, z_d represent the three-dimensional coordinates of the target position. The parameter ω denotes the weight value. φ denotes the distance from the current position to the target position. This means that the reward is calculated based on the ratio between the Euclidean distance from the starting position to the target position and the distance from the current position to the target position. It is worth mentioning that when $\varphi < 1$, in order to avoid too small a value of φ , the reward value is too large, which causes the agent explore the place near the target position and thus reduces the efficiency of the algorithm. Therefore, it is stipulated that the reward value obtained by the intelligence when $\varphi < 1$, the reward value obtained by the agent is the Euclidean distance between the start position and the target position.

In addition, to encourage the robot to learn paths with fewer steps, a small penalty R_s (where $R_s < 0$) is deducted for each step taken.

Therefore, after each decision the robot makes, the reward it receives is given by Eq. (8):

$$R = R_{sp} + R_t + R_s \quad (8)$$

3.2 Delta Robot Path Planning Method Based on DQN

Q-learning is a value-based reinforcement learning algorithm that updates the Q-table using only the state s , action a , reward r , and next state s' , without requiring the current action. This makes it an offline learning method, which improves computational efficiency compared to online learning and is therefore widely used. The iterative update formula is shown as Eq. (9) [25]:

$$Q(s, a) \leftarrow Q(s, a) + \alpha[r + \gamma \max_{a'} Q(s', a') - Q(s, a)] \quad (9)$$

In practical Delta robot systems, experiments have revealed that due to the high dimensionality of the system state and action space, a phenomenon known as the “curse of dimensionality” occurs. Under such algorithms, the robot fails to converge to the maximum reward. Therefore, we employ the DQN algorithm, which introduces neural networks to approximate the Q-table. This approach effectively improves computational efficiency and addresses the issue of the “curse of dimensionality”.

The loss function in the DQN algorithm is the minimization of the squared expectation of the temporal difference, as shown in Eqs. (10) and (11):

$$L(\omega) = \frac{1}{N} \sum_i (y_i - Q_\omega(s_i, a_i))^2 \quad (10)$$

$$y_i = r_i + \gamma \max_a Q_{\omega^-}(s_{i+1}, a) \quad (11)$$

ω denotes the parameters of the training network, while ω^- represents the parameters of the target network.

Fig. 5 is the DQN algorithm framework diagram for the Delta robot. In subsequent experiments, we will compare the effects of varying the number of neurons in the network and the number of hidden layers.

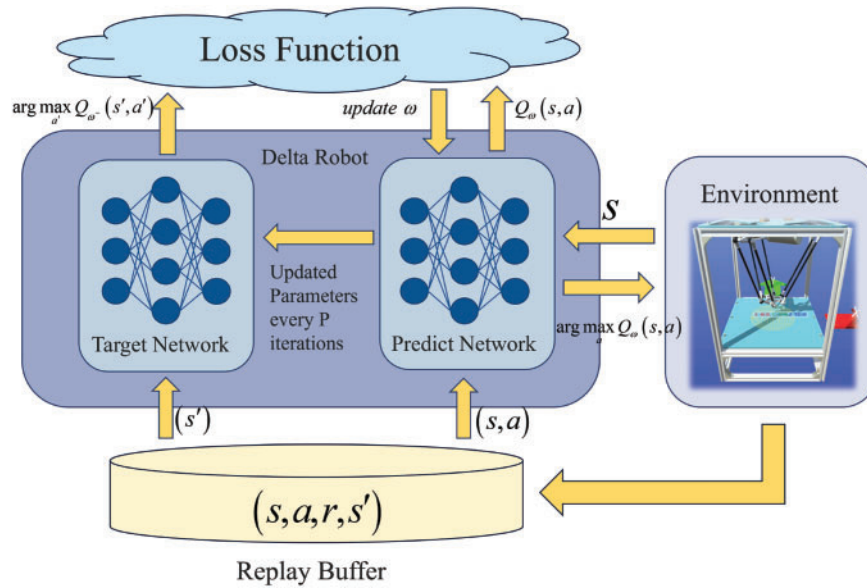


Figure 5: The DQN algorithm framework diagram for the delta robot

4 Experiments and Discussion

First, we set up the physical TM68 three-axis parallel robot. The experimental environment includes Python 10, Anaconda, and the Pytorch framework, as well as a computer with a 3.6 GHz 12th Intel Core i7 processor and GPU RTX 4060. Additionally, we used Unity3D to build the digital twin system.

4.1 Virtual Experiment

First, we trained the model using the DQN reinforcement learning algorithm. In the experiment, we found that a single-layer neural network structure achieved higher learning efficiency. Therefore, in

the following experiments, we present the use of a single-layer neuron architecture and test the optimal parameters by varying the number of neurons.

By visually detecting the length and width of the current obstacle, we can determine the spatial coordinates of the rectangular obstacle (with the height assumed to be a known value of 30 mm). Additionally, the center point coordinates of the target area are detected, where the target object is a circular iron plate with a diameter of 23 mm.

When training a reinforcement learning algorithm, selecting the right parameters can have a significant impact on the training results. In many explorations of the discount rate settings we found that the reinforcement learning algorithm has a high efficiency when γ is between 0.9 and 1, and there is a phenomenon of non-convergence when the gamma is greater than or equal to 1. Therefore, we selected γ to be 0.90, 0.92, 0.94, 0.95, 0.96, 0.97 and 0.98, respectively, and carried out the test. It is found that the algorithm converges the fastest when γ is equal to 0.98 as shown in the Fig. 6, so we choose γ as 0.98. All other key parameters in the reinforcement algorithm are derived from this scheme.

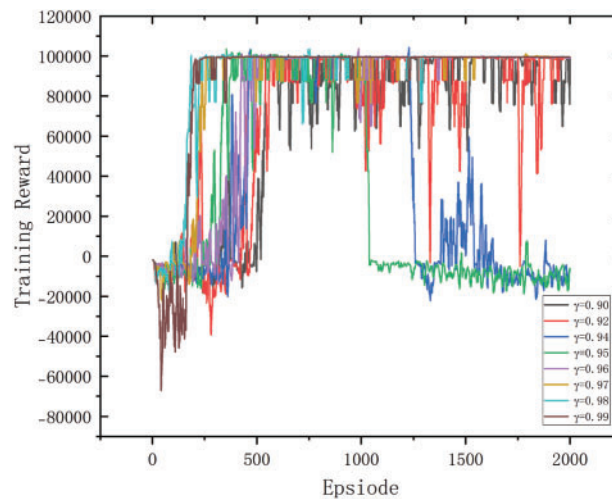


Figure 6: Chart of reinforcement learning training at different discount rates

After obtaining the above information, training was initiated. The results, shown in Fig. 7, reveal that with 1024 neurons, the convergence speed was relatively the fastest. However, using 64, 128, 256, and 512 neurons also yielded good convergence and training performance. With 2048 neurons, convergence to the maximum reward value becomes difficult even when the number of training iterations reaches 2000. Fig. 8 shows the number of trained action sequences, the size of the reward value of the optimal action, where the minimum number of trained action sequences is 48 for 1024 and 128 neurons. As shown in Fig. 9, a 3D schematic of the planned routes as well as the obstacle positions are also drawn, and it can be found that the optimal route trained under the no-obstacle condition is the line indicated in red, which has an obvious collision with the obstacle, while the other four lines avoid the obstacle and reach the goal point. In the subsequent physical experiments, the trained action sequences are transmitted to the PLC controller via communication to observe the experimental results.

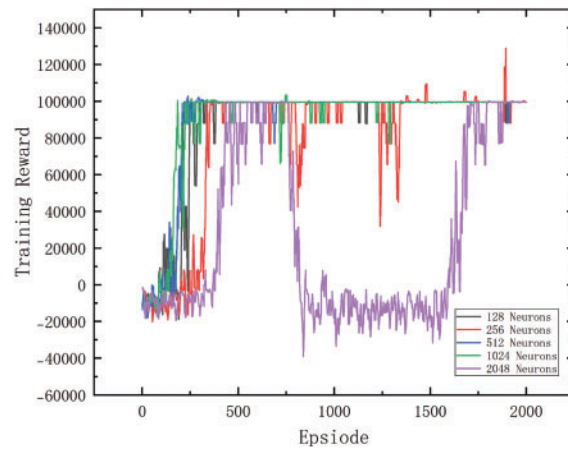


Figure 7: Reinforcement learning training results diagram

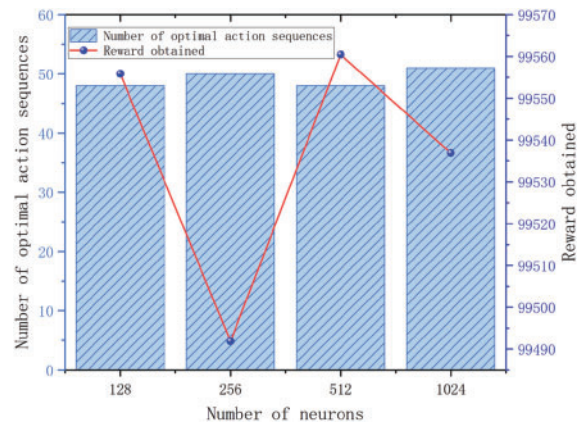


Figure 8: Comparison of different neuron counts under unobstructed and obstructed conditions for the physical robot. (a) Unobstructed condition; (b) Neuron count of 128; (c) Neuron count of 256; (d) Neuron count of 512; (e) Neuron count of 1024

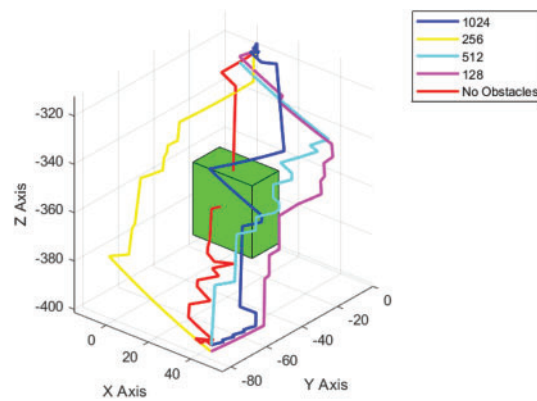


Figure 9: Chart of optimal action sequences and rewards for different numbers of neurons

4.2 Physical Experiment and Communication Experiment

Transmit the trained action sequences to the PLC controller via communication, observe the effects in the physical system, and complete the communication experiment between the digital twin system and the physical robot.

In addition to the experimental images shown in the paper, as shown in Fig. 10 we repeated the training with 5 more repetitions at each parameter, for a total of 20 experiments. The experimental results demonstrate that the planned paths successfully enable the robot to avoid obstacles and reach the target point. One of the routes trained with 1024 neurons was the fastest in the physical robot test in 5.41 s.



Figure 10: Plot of different numbers of neurons in the physical machine in the accessibility condition, vs. the accessibility condition (a) accessibility condition; accessibility condition (b) number of neurons is 128 (c) number of neurons is 256 (d) number of neurons is 512 (e) number of neurons is 1024

Simultaneously with the physical robot operations, shown in Fig. 11, we also conducted communication and interaction experiments with the digital twin. We find that physical and twin robots can stay synchronized.

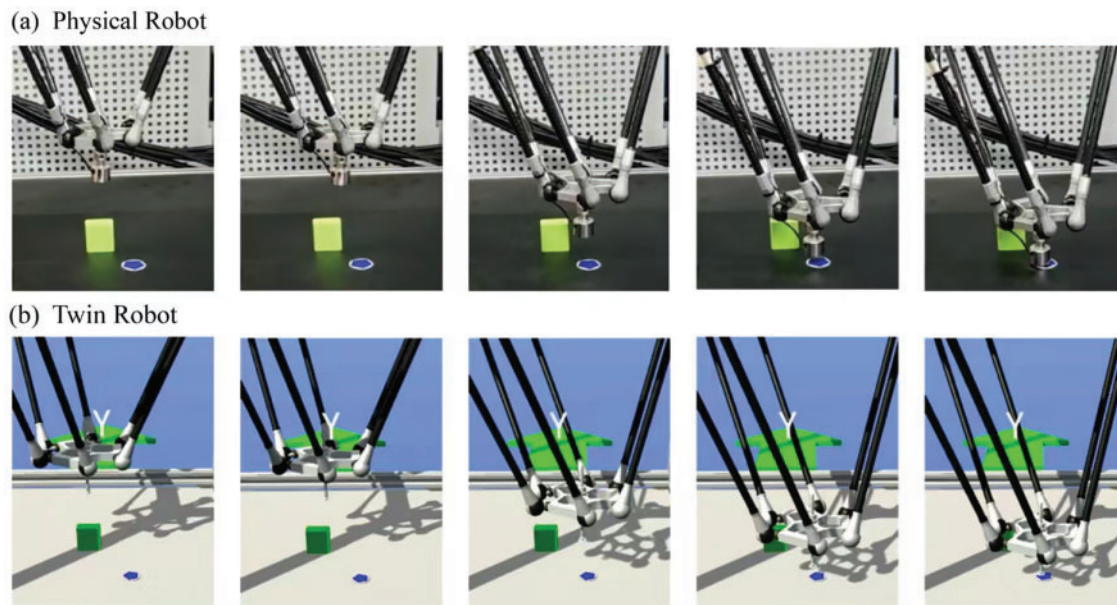


Figure 11: Correspondence between the motion of (a) the physical robot and (b) the twin robot

5 Conclusion

This work introduces the development of a digital twin system for Delta robots that seamlessly integrates virtual entities, physical components, service functions, and twin data connections. By incorporating deep reinforcement learning algorithms, it introduces a customized reward mechanism that effectively addresses path planning challenges, thereby enhancing the adaptability and safety of Delta robots in real-world industrial settings. Communication and interactive control between the physical robot and its digital twin are successfully established. The implementation of the digital twin enables more efficient and safer operational planning for Delta robots by mitigating safety risks associated with direct physical experimentation, reducing wear and tear, and significantly lowering experimental costs, thereby yielding considerable operational benefits. While this research introduces digital twin technology for individual Delta robots, future research should explore the cooperative control of multi-robot systems to expand its application across a broader range of industrial scenarios.

Acknowledgement: Not applicable.

Funding Statement: This work was supported in part by the National Natural Science Foundation of China under Grants 62303098 and 62173073, in part by China Postdoctoral Science Foundation under Grant 2022M720679, in part by the Central University Basic Research Fund of China under Grant N2304021, and in part by the Liaoning Provincial Science and Technology Plan Project-Technology Innovation Guidance of the Science and Technology Department under Grant 2023JH1/10400011.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Hongxiao Wang, Dingsen Zhang; data collection: Hongxiao Wang; analysis and interpretation of results: Hongshen Liu, Ziyi Zhang, Yonghui Yue, Jie Chen; draft manuscript preparation: Hongshen Liu. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data available on request from the Dingsen Zhang.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Alikoc B, Sustr V, Zitek F, Burget P, Lomakin A. Motion modeling of a 5-axis delta robot with telescopic shafts. *J Mech Robot Trans ASME*. 2024 Apr 1;16(4):041013. doi:10.1115/1.4062672.
2. Yang C, Gao Y, Tian C, Yao Q. Sim-to-real: designing locomotion controller for six-legged robot. In: 2019 9th IEEE Annual International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (IEEE-CYBER 2019). IEEE Annual International Conference on Cyber Technology in Automation Control and Intelligent Systems; 2019 Jul 29–Aug 02; Suzhou, China. p. 746–51.
3. Tan S, Guerrero JM, Xie P, Han R, Vasquez JC. Brief survey on attack detection methods for cyber-physical systems. *IEEE Syst J*. 2020 Dec;14(4):5329–39. doi:10.1109/JSYST.2020.2991258.
4. Wu J, Yang Y, Cheng X, Zuo H, Cheng Z. The development of digital twin technology review. In: 2020 Chinese Automation Congress (CAC); 2020; Shanghai, China: Chinese Automation Congress (CAC). p. 4901–6.
5. Liu X, Jiang D, Tao B, Xiang F, Jiang G, Sun Y, et al. A systematic review of digital twin about physical entities, virtual models, twin data, and applications. *Adv Eng Inform*. 2023;55:101876. doi:10.1016/j.aei.2023.101876.
6. Zhang Z, Liu M, Sun M, Deng R, Cheng P, Niyato D, et al. Vulnerability of machine learning approaches applied in IoT-based smart grid: a review. *IEEE Internet Things J*. 2024;11(11):18951–75. doi:10.1109/JIOT.2024.3349381.
7. Zhang Z, Yang Z, Yau DKY, Tian Y, Ma J. Data security of machine learning applied in low-carbon smart grid: a formal model for the physics-constrained robustness. *Appl Energy*. 2023 Oct 1;347:121405. doi:10.1016/j.apenergy.2023.121405.
8. Zhang Z, Deng R, Yau DKY. Vulnerability of the load frequency control against the network parameter attack. *IEEE Trans Smart Grid*. 2024 Jan;15(1):921–33. doi:10.1109/TSG.2023.3275988.
9. Zhang Z, Deng R, Tian Y, Cheng P, Ma JSPMA. Stealthy physics-manipulated attack and countermeasures in cyber-physical smart grid. *IEEE Trans Inf Forensics Secur*. 2023;18:581–96.
10. Zhang Z, Yang K, Tian Y, Ma J. An anti-disguise authentication system using the first impression of avatar in metaverse. *IEEE Trans Inf Forensics Secur*. 2024;19:6393–408.
11. Chakshu NK, Sazonov I, Nithiarasu P. Towards enabling a cardiovascular digital twin for human systemic circulation using inverse analysis. *Biomech Model Mechanobiol*. 2021;20(2):449–65. doi:10.1007/s10237-020-01393-6.
12. Zhou T, Chen M, Cao G, Hu J, Liu H. Inverse kinematics embedded network for robust patient anatomy avatar reconstruction from multimodal data. *IEEE Robot Autom Lett*. 2024;9(4):3395–402. doi:10.1109/LRA.2024.3366418.
13. Cheng DJ, Zhang J, Hu ZT, Xu SH, Fang XF. A digital twin-driven approach for on-line controlling quality of marine diesel engine critical parts. *Int J Precis Eng Manuf*. 2020 Oct;21(10):1821–41. doi:10.1007/s12541-020-00403-y.
14. Kampker A, Stich V, Jussen P, Moser B, Kuntz J. Business models for industrial smart services-the example of a digital twin for a product-service-system for potato harvesting. *Procedia CIRP*. 2019;83:534–40. doi:10.1016/j.procir.2019.04.114.
15. Defraeye T, Tagliavini G, Wu W, Prawiranto K, Schudel S, Kerisima MA, et al. Digital twins probe into food cooling and biochemical quality changes for reducing losses in refrigerated supply chains. *Resour Conserv Recycl*. 2019 Oct;149:778–94.
16. Zhang Q, Xiao R, Liu Z, Duan J, Qin J. Process simulation and optimization of arc welding robot workstation based on digital twin. *Machines*. 2023 Jan;11(1):53. doi:10.3390/machines11010053.
17. Yang H, Cheng F, Li H, Zuo Z. Design and control of digital twin systems based on a unit level wheeled mobile robot. *IEEE Trans Vehicular Technol*. 2024 Jan;73(1):323–32. doi:10.1109/TVT.2023.3309029.
18. Zhang D, Shang K, Zhang Y, Feng L. Soft sensor for blast furnace temperature field based on digital twin. *IEEE Trans Instrum Meas*. 2024;73:9516311. doi:10.1109/TIM.2024.3458070.

19. Zhang D, Gao X. A digital twin dosing system for iron reverse flotation. *J Manuf Syst.* 2022 Apr;63:238–49. doi:10.1016/j.jmsy.2022.03.006.
20. Liu C, Cao G, Qu Y. Safety analysis via forward kinematics of delta parallel robot using machine learning. *Saf Sci.* 2019 Aug;117:243–9. doi:10.1016/j.ssci.2019.04.013.
21. Zhang Z. A flexible new technique for camera calibration. *IEEE Trans Pattern Anal Mach Intell.* 2000 Nov;22(11):1330–4. doi:10.1109/34.888718.
22. Zhang D, Gao X, Wang H. Prediction model of iron reverse flotation tailings grade based on multi-feature fusion. *Measurement.* 2023 Jan;206:112062.
23. Wang B, Liu Z, Li Q, Prorok A. Mobile robot path planning in dynamic environments through globally guided reinforcement learning. *IEEE Robot Autom Lett.* 2020 Oct;5(4):6932–9. doi:10.1109/LRA.2020.3026638.
24. Cai M, Tang Y, Du Y, Liu T, Fan QY. A novel path planning approach based on exploration-efficient reinforcement learning. In: 2024 36th Chinese Control and Decision Conference (CCDC); 2024; Xi'an, China. p. 385–9.
25. Liu Y, Xu H, Liu D, Wang L. A digital twin-based sim-to-real transfer for deep reinforcement learning-enabled industrial robot grasping. *Robot Comput Integr Manuf.* 2022 Dec;78:102365.