International Journal of Basic and Applied Sciences, 14 (2) (2025) 429-440

SPC

International Journal of Basic and Applied Sciences

Website: www.sciencepubco.com/index.php/IJBAS
https://doi.org/10.14419/1dy21h49
Research paper

AI-Driven Intelligent Control Strategies for Industrial Robotics: A Reinforcement Learning Approach

Kishore Kunal [1], M. Kathiravan [2], Vairavel Madeshwaren [3] *, T. Chandrakala [4], Veeramani Ganesan [5], Sheifali Gupta [6]

[1] *Professor of Business Analytics, Loyola Institute of Business Administration, Chennai, TamilNadu, India*
[2] *Professor, Dapartment of Computer Science, Saveetha Institute of Medical and Technical Sciences (SIMATS),TamilNadu, India*
[3] *Department of Agriculture Engineering, Dhanalakshmi Srinivasan College of Engineering, Coimbatore, TamilNadu, India*
[4] *Assistant Professor, Department of Computer Science and Applications, Jawahar Science College, Neyveli, TamilNadu, India*
[5] *Professor, Department of Management and Business Administration, Jeppiaar Institute of Technology, Sunguvarchatram, Sriperumbudur, TamilNadu, India*
[6] *Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India*
**Corresponding author E-mail: phdannauniv2020@gmail.com*

Received: May 15, 2025, Accepted: June 22, 2025, Published: June 27, 2025

Abstract

This study proposes an AI-driven adaptive control strategy to enhance the learning, adaptability, and autonomous performance of robotic manipulators in dynamic and unstructured industrial environments. Moving beyond the limitations of conventional model-based controllers, the research introduces a self-learning framework that integrates real-time sensor data from LiDAR and stereo vision cameras. This data continuously informs and optimizes the robot's motion trajectories in both simulated and real-world tasks. The system's core innovation lies in combining Reinforcement Learning (RL) with Deep Neural Networks (DNNs) for adaptive trajectory planning and error compensation. Specifically, the Proximal Policy Optimization (PPO) algorithm is employed to fine-tune control strategies based on real-time sensory feedback, allowing the robotic system to autonomously adapt to variations in object positions and unexpected disturbances. An Edge-AI module is embedded into the architecture to enhance decision-making speed and reduce latency during task execution. Experimental validation, including scenarios like arc welding and sealant dispensing, shows the proposed system outperforms traditional PID-based adaptive controllers. The AI-driven solution demonstrated improved precision, faster convergence, and superior adaptability under complex and fluctuating manufacturing conditions. The study also opens pathways for future integration of hybrid AI techniques—such as fuzzy logic and genetic algorithms—for even more intelligent and responsive robotic systems.

Keywords: *Adaptive Control; Reinforcement Learning; Edge-AI; Robotic Manipulation; Deep Neural Networks; Industrial Automation.*

1. Introduction

Industrial robotic manipulators have revolutionized modern manufacturing by automating repetitive tasks, enhancing precision, and improving operational efficiency across sectors such as automotive, electronics, aerospace, and heavy machinery. Traditionally, these manipulators have relied on model-based control systems—often fine-tuned through PID or adaptive controllers—that operate effectively in structured environments where parameters remain largely consistent. However, the manufacturing landscape is rapidly evolving. With the advent of mass customization, shorter product life cycles, and increased demand for flexibility, factories are becoming more dynamic and unstructured. This transition brings to the forefront a set of complex challenges that conventional robotic control strategies are ill-equipped to handle.

In small- to medium-scale manufacturing setups, frequent changes in product design, materials, or tooling configurations demand constant recalibration of robotic systems. This not only consumes valuable production time but also leads to inefficiencies in motion planning, error correction, and environmental awareness. Moreover, traditional controllers lack the adaptability and learning capacity needed to deal with unexpected disturbances, uncertain object locations, or shifting constraints within the workspace. As a result, these systems struggle to maintain accuracy and consistency under dynamic real-world conditions, limiting their scalability and usefulness in next-generation smart factories.

To bridge this gap, recent advancements in artificial intelligence (AI)—particularly in reinforcement learning (RL), deep learning, and edge computing—offer a transformative opportunity. These technologies enable robots to not only respond to their environment but also learn from it, improving their behavior over time without human intervention. Reinforcement learning, for instance, allows a robotic agent to iteratively refine its actions through trial and error, guided by reward functions that prioritize optimal performance. When integrated with high-dimensional sensory input from LiDAR, stereo vision, and other real-time data sources, this creates a pathway toward self-optimizing and highly responsive robotic control systems.

The present study capitalizes on these advancements by proposing an AI-driven adaptive control framework that shifts away from rigid model-based strategies toward autonomous, intelligent decision-making. The framework integrates RL algorithms—specifically Proximal Policy Optimization (PPO)—with Deep Neural Networks (DNNs) and real-time sensor fusion to enable adaptive trajectory planning and compensation for dynamic disturbances. Additionally, the use of Edge-AI modules ensures low-latency processing, allowing the system to make immediate decisions without relying heavily on centralized computing infrastructure.

In addition to providing a real-world example, this research advances the goals of Industry 4.0 and beyond by applying this framework to intricate industrial applications like arc welding and sealant dispensing, where accuracy and flexibility are crucial. Future robotic systems that can function autonomously, intelligently, and adaptively in actual manufacturing environments may be made possible by the work's results. With the shift to Industry 4.0, smart manufacturing is now at the forefront of technological advancement. The idea of digital twins—virtual copies of physical systems enhanced with artificial intelligence (AI)—is one of the key technologies driving this change. Real-time analytics, predictive maintenance, and intelligent diagnostics are made possible by these AI-powered digital twins, which serve as the foundation for future smart factories. Through digital simulation of production processes and equipment behavior, manufacturers can enhance overall efficiency, minimize downtime, and optimize operations [1].

Concurrently, methods for deep learning (DL), machine learning (ML), and artificial intelligence (AI) have become essential parts of robotics systems of the future. These techniques give robots the capacity to develop their decision-making skills in dynamic environments, identify intricate patterns, and learn from their experiences. According to a thorough analysis, the combination of artificial intelligence (AI) and robotics improves autonomy by allowing machines to perform industrial tasks with high accuracy and dependability while requiring less human intervention [2]. Additionally, by making adaptive and self-learning robotic systems possible, AI is changing the landscape of advanced robotics. By utilizing neural networks and probabilistic learning algorithms, these intelligent robotic systems can carry out intricate manipulation, navigation, and task planning. The increasing importance of deep learning (DL) in robotic vision and control is highlighted in a recent review, which also highlights how convolutional neural networks (CNNs) and reinforcement learning algorithms work especially well for teaching robots to recognize objects and manipulate them in unpredictable situations [3]. The creation of dynamic role-adaptive collaborative robots, or cobots, is another important advancement. These robots actively engage in production lines by modifying their behavior in real time. Based on worker presence, workloads, or environmental cues, these robots use AI to switch between roles. This flexibility reduces the amount of energy and resources used to complete tasks, which not only increases productivity but also promotes sustainable manufacturing practices [4].

Through constant sensory data analysis, robots can comprehend their surroundings and make the best movement plans. According to this field's research, adding machine learning (ML) algorithms to robotic systems improves task success rates, trajectory optimization, and obstacle avoidance. Additionally, these systems show resilience in unstructured environments where traditional programming would not work, like construction sites and warehouses [5]. One particularly exciting area of artificial intelligence is reinforcement learning (RL), which enables robots to discover the best course of action through trial-and-error interactions with their surroundings. In smart manufacturing systems, for example, RL has been used to schedule job shops, allowing robots to assign and sequence tasks on their own for increased productivity. In contrast to conventional scheduling techniques, RL systems provide resilience against interruptions and machine failures by adapting over time [6]. Another study used reinforcement learning (RL) to control variable stiffness actuators in robotic manipulators, which greatly improves robot precision and adaptability in dynamic operations. Real-time mechanical property modification is made possible by these actuators, which are essential for handling delicate or fluctuating loads during assembly and packaging tasks. Reliability and product quality are increased by RL algorithms, which make sure the robot learns to adjust its movements for the best stiffness under various circumstances [7]. In addition to its theoretical uses, reinforcement learning has been successfully applied in real-world settings such as robotic arms for pick-and-place jobs. To perform item sorting in a warehouse, for instance, a six-DOF manipulator was trained using deep reinforcement learning. The study showed that in terms of task generalization accuracy and flexibility, these AI-driven systems could perform better than conventional automation [8].

Robotic drilling and screwing in industrial assembly lines are two examples of high-friction or restricted environments where RL has been investigated for application. In these situations, robots trained with RL learned to modify their torque and speed in response to material resistance, preventing tool damage and enhancing operational safety. Additionally, these intelligent systems eliminate the need for constant supervision, which makes them perfect for industrial tasks that require repetition [9]. AI has also helped robotic process automation (RPA), especially when RL is integrated to improve cognitive automation. RL-augmented bots, as opposed to rule-based RPA systems, can dynamically modify workflows in response to contextual information, discovering the best routes via trial and error. This results in enhanced decision-making and process efficiency in the automation of document processing, invoicing, and customer support [10]. Conventional robotics has long struggled with industrial perception tasks like recognizing and categorizing unordered objects. To intelligently identify and arrange such targets, an RL-based method has been put forth, which increases accuracy and speed. Occlusions, clutter, and object shape variability—all of which are frequent problems in actual industrial settings—are addressed by the system by combining computer vision and control feedback [11].

Artificial intelligence (AI)-enabled robotics also addresses sustainability, particularly in the context of smart cities. In urban management, robots trained with RL algorithms have been used for energy-efficient tasks like waste collection, traffic monitoring, and energy conservation. These systems minimize ecological impact and lower operating costs by modifying their behavior in response to sensor feedback and environmental conditions [12]. With the incorporation of AI, advanced control systems have also undergone significant evolution. Neural network and fuzzy logic-enabled intelligent controllers can now proactively modify control parameters in response to shifts in system dynamics. In industrial automation applications like robotic welding, painting, and machining, this results in increased stability and accuracy [13]. The creation of force-torque control techniques for delicate assembly tasks such as robotic nut-to-bolt mating is another important field. In these kinds of tasks, even small misalignments can result in serious harm or inefficiencies. Robots can learn to modify their force application and tool alignment using AI-based control, which leads to more error-free and seamless operations [14].

Intelligent sensing systems and tactile feedback have significantly enhanced robots' capacity to carry out contact-rich tasks. AI models use force and proximity sensor data to estimate an object's properties and modify movement or grip strength accordingly. This improves robotic dexterity and makes tasks like component assembly, insertion, and polishing easier [15]. Smart robotic manufacturing systems, which fall under Industry 4.0, use cyber-physical integration to synchronize digital control with physical processes. Robots analyze large amounts of data, interact with other machines, and modify their operations in real time. Customized manufacturing workflows, flexible production lines, and predictive maintenance are made possible by this interconnection [16]. Hyperconnectivity, human-machine symbiosis, and cognitive automation are the hallmarks of Industry 6.0, which is anticipated to be the next big step forward from Industry 5.0. AI will play a major role in enabling this future by enabling robots to make decisions on their own, reason ethically, and have emotional intelligence.

Through intelligent, ethical, and sustainable technologies, Industry 6.0 may redefine industrial ecosystems according to preliminary studies [17]. Maintaining cybersecurity becomes critical as robots become more self-sufficient. Artificial Intelligence (AI) has been used to identify and categorize cyberattacks that target robotic systems through behavior prediction and anomaly detection. These systems safeguard the data integrity and physical safety of industrial operations by detecting departures from typical activity patterns and initiating fail-safe procedures [18].

Another cutting-edge application is robotic inspection stations, which use AI and machine vision to identify flaws and continuously check the quality of products. By reducing human error and expediting the inspection process, these stations enable the early identification of anomalies that may compromise product safety or compliance [19–20]. With AI predictive maintenance—which analyzes sensor data to identify early indicators of equipment deterioration—it has become more and more possible. Businesses can prolong the life of costly machinery, prevent costly downtime, and schedule maintenance proactively by anticipating failures before they happen [21]. A robot's ability to function autonomously in new settings also depends on autonomous learning. Robots can adapt their behaviors to unexpected challenges or shifts in task requirements by using real-time learning algorithms and continuous data acquisition, which increases their resilience and context awareness [22–24].

Industrial managers' approach to optimization is also being revolutionized by AI-driven decision support systems. By combining real-time feedback, predictive analytics, and simulation models, these systems assist stakeholders in making well-informed choices regarding scheduling, quality control, and resource allocation [25]. One of the main areas of current research is the integration of human-like perception into robotic systems. Autonomous navigation and cooperative human-robot environments are based on these perception systems [26–27]. Finally, the combination of vision-based intelligence and mechanical design principles has produced more resilient and flexible robotic platforms. By facilitating dynamic recalibration and learning-driven design modifications, AI improves these platforms. These systems are especially helpful in environments where there is a lot of variation in the shapes, textures, and materials of objects, as well as in custom manufacturing [28].

2. Materials and methods

In this section, the research methodology and experimental setup that shaped the foundation of this study are described in detail. The section encompasses the identification of the problem, a systematic explanation of the data collection process, details regarding measurement approaches, the role of IoT sensor tools, the complete flow of the proposed methodology, an in-depth technical explanation of the core algorithmic strategy, and the performance metrics used for validation. Each subsection unfolds the step-by-step framework followed to design, develop, and assess the AI-driven intelligent control strategy proposed for industrial robotics using reinforcement learning principles.

2.1. Problem description

Industrial robots play an integral role in modern automated manufacturing setups, but their efficiency and adaptability are often constrained by conventional model-based control strategies, particularly in dynamic or unstructured environments. Traditional Proportional-Integral-Derivative (PID) controllers, despite their widespread use, often struggle with issues such as slow adaptive response to environmental disturbances, limited generalization ability in new scenarios, and suboptimal control under sensor noise and unforeseen changes. This research addresses this problem by proposing a reinforcement learning-driven intelligent control framework, specifically designed to overcome these constraints and enable autonomous robotic systems to dynamically adapt to variations in real-time industrial settings. The goal is to enhance trajectory tracking, improve error compensation, and minimize energy consumption while ensuring high success rates under varying operational complexities.

2.2. Data collection

For this AI-driven robotic control system to be successful, the data collection stage is essential. Across more than 50 distinct scenarios in both simulated and real-world settings, real-time data were collected from industrial robots carrying out tasks like arc welding and sealant dispensing. To capture multi-modal sensory inputs, the robotic platform was equipped with joint encoders, torque sensors, LiDAR sensors, and stereo vision cameras. With the help of a strong acquisition framework that guaranteed synchronized data logging, the dataset grew to be more than 180 GB in size. To improve the reinforcement learning models' flexibility under dynamic constraints, this dataset covers a wide range of operational conditions such as changes in lighting tool alignment, object placement vibration, thermal drift, and actuator noise. Diverse datasets improve model generalization, but they also present replicability issues, especially in real-world scenarios where it is challenging to precisely replicate environmental and disturbance variables. Table 1 below summarizes the key attributes of the collected dataset:

Table 1: Summary of Dataset Attributes for Robotic Control System [5]

Attribute	Description
Dataset Size	180 GB (real-time sensor and control data)
Number of Scenarios	50+ distinct task and environment setups
Task Types	Arc welding, Sealant dispensing
Sensor Modalities	LiDAR, Stereo vision, Torque sensors, Joint encoders
Disturbance Types	Lighting variation, Tool misalignment, Thermal drift, Vibration, Occlusions
Simulation vs Real-world	60% real-world, 40% high-fidelity simulation
Replicability Limitations	High scenario variability in real-world conditions reduces exact reproducibility

2.3. Data measurement

The collected data was carefully measured using calibrated and industry-grade sensing systems to ensure the highest precision. Position errors were measured in millimeters using high-resolution laser distance sensors, while orientation errors were captured via gyroscopic data from the IMU integrated into the robot's joint modules. Energy consumption was measured using inline power monitoring units, and task completion times were recorded through the robot's internal timestamping systems (Fig. 1). Adaptive response times were benchmarked by introducing controlled disturbances and monitoring the duration from detection to stabilization. All measurements were repeated over multiple runs to ensure statistical consistency.

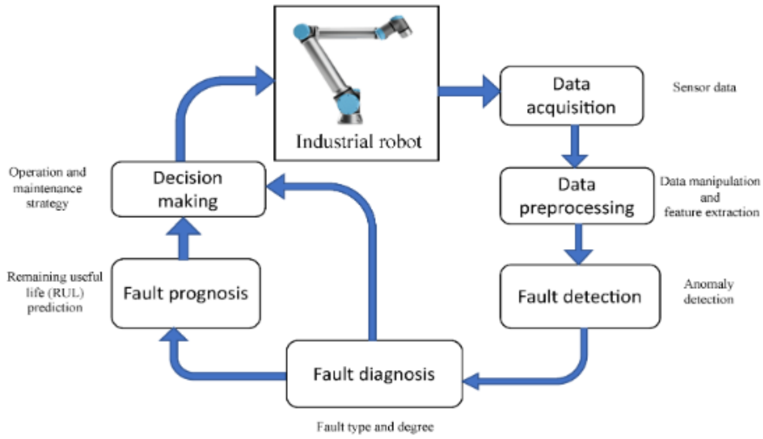


Fig. 1: Workflow Process [7].

2.4. IoT sensor tools

The success of an adaptive robotic control strategy is highly dependent on the reliability and precision of the sensor systems feeding real-time data into the control loop. In this study, a combination of LiDAR (Light Detection and Ranging) sensors and stereo vision cameras served as the robot's perceptual front end. The LiDAR systems provided high-accuracy distance and spatial mapping data, which was essential for obstacle avoidance and workspace boundary estimation. Stereo cameras were used for depth perception and object detection under varying lighting conditions. Additionally, torque sensors embedded in the robot's joints allowed real-time feedback for force control, while encoder readings provided joint angle positions with high fidelity. Together, these IoT-based sensors formed a robust information acquisition backbone that enabled the reinforcement learning controller to make informed decisions in real-time.

2.5. Experimental setup

The methodology designed in this research follows a closed-loop cycle where a reinforcement learning agent continuously interacts with the environment through perception, action, and feedback. Initially, the robot's environment is perceived via sensor input, which includes positional data, force feedback, and visual information. This sensory information is converted into state vectors, which are then processed by a deep neural network-based policy model that maps the state to an optimal control action. Figure 2 demonstrates the experimental setup.

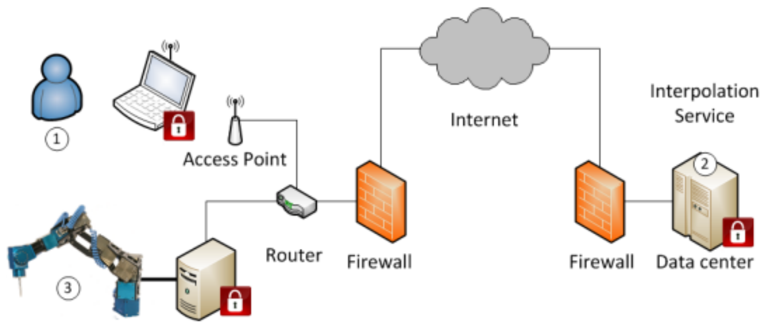


Fig. 2: Experimental setup [12].

The selected action is executed by the robot, and the effect on the environment is observed through updated sensory input. Concurrently, task performance, energy consumption, adaptive response time, and deviation from the target trajectory are used to calculate the reward signal. By altering the neural network parameters, the reinforcement learning agent improves its policy over time to maximize cumulative reward. The training process is iterative, and after the policy stabilizes, the model is put into use for real-world control, where it keeps improving and adapting to small invisible disruptions.

3. Proposed technique

The suggested control system is based on the Deep Neural Network function approximation of Reinforcement Learning (RL). Learning the best control policies for the robotic manipulator was addressed by using the Proximal Policy Optimization (PPO) algorithm. PPO is notable for its capacity to maintain a safe margin for policy updates while striking a balance between exploration and exploitation. The neural policy network $\pi\theta(a|s)$ outputs control actions given the current state s, with its parameters θ adjusted to maximize expected reward (Fig 3).

The control problem is mathematically framed as a Markov Decision Process (MDP) characterized by the tuple {S, A, R, T}, where S represents the state space, A the action space, R the reward function, and T the state transition probability. The objective is to optimize the expected reward (Eq 1):

$$J(\theta)=E\pi\theta[t=0\sum\infty\gamma tRt] \quad (1)$$

Where γ is the discount factor, balancing long-term and short-term rewards. The PPO algorithm maximizes the clipped surrogate objective, which prevents large deviations from the old policy (Eq 2):

L(θ)=Et[min(rt(θ)A^t, clip(rt(θ), 1−ϵ, 1+ϵ) A^t)] (2)

Where rt(θ)=πθold(at|st)πθ(at|st) is the probability ratio, ϵ is the clipping parameter, and A^t is the advantage function estimator.

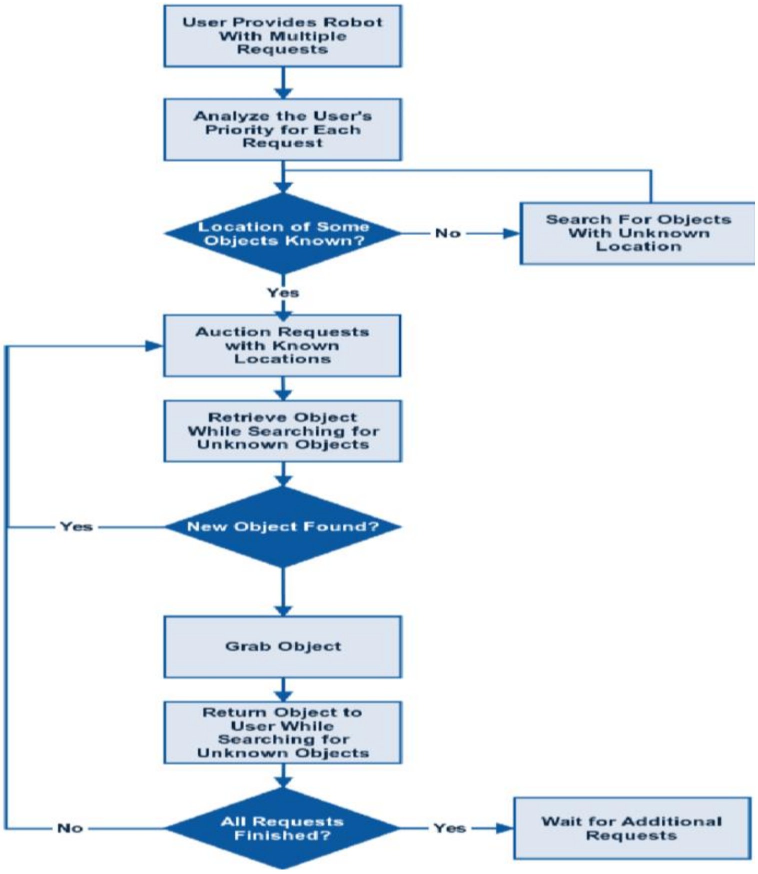


Fig. 3: Industrial Robot Implementation Architecture [16].

The advantage function is computed using (Eq 3):

^At=δt+(γλ)δt+1+(γλ)2δt+2+⋯ (3)

where δt=Rt+γV(st+1)−V(st). Here, V(s_t) represents the value function estimating the expected return from state s_t.
The state-value function is trained using the squared error loss between the predicted and actual returns (Eq 4):

Lv(ϕ)=21(Vϕ(st)−Rt)2 (4)

Finally, the entropy bonus is used to encourage exploration during training (Eq 5):

Lentropy=Et[−a∑πθ(a|st)logπθ(a|st)] (5)

By combining Equations 2, 4, and 5, the total loss for PPO is minimized to update the neural network weights, ensuring stable convergence and optimal control performance for the robotic system.

PPO Algorithm Implementation

```
# Proximal Policy Optimization (PPO) Algorithm
import torch
import torch.nn as nn
import torch.optim as optim

class PolicyNetwork(nn.Module):
def __init__(self, state_dim, action_dim):
super(PolicyNetwork, self).__init__()
self.fc1 = nn.Linear(state_dim, 128)
self.fc2 = nn.Linear(128, 128)
self.fc3 = nn.Linear(128, action_dim)

def forward(self, state):
x = torch.relu(self.fc1(state))
x = torch.relu(self.fc2(x))
```

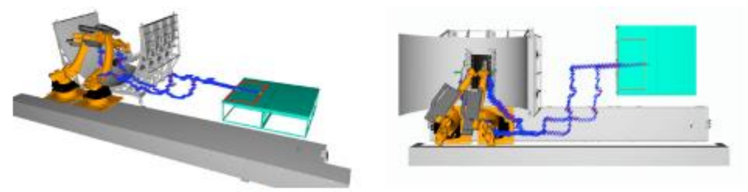
```
action_probs = torch.softmax(self.fc3(x), dim=-1)
 return action_probs

def compute_advantages(rewards, values, gamma=0.99, lam=0.95):
advantages = []
gae = 0
for i in reversed(range(len(rewards))):
delta = rewards[i] + gamma * values[i + 1] - values[i]
gae = delta + gamma * lam * gae
advantages.insert(0, gae)
 return advantages

def ppo_update(policy, optimizer, old_log_probs, states, actions, returns, advantages, clip_eps=0.2):
for _ in range(10): # number of training epochs
new_log_probs = torch.log(policy(states).gather(1, actions))
ratio = torch.exp(new_log_probs - old_log_probs)
surrogate1 = ratio * advantages
surrogate2 = torch.clamp(ratio, 1 - clip_eps, 1 + clip_eps) * advantages
loss = -torch.min(surrogate1, surrogate2).mean()

 optimizer.zero_grad()
 loss.backward()
 optimizer.step()
```

4. Results and discussion

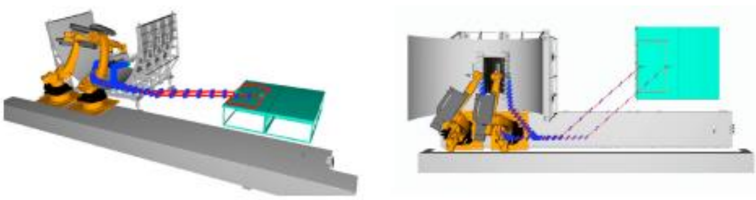
The following tables and graphs present the comparative performance of the proposed AI-driven adaptive control strategy (using PPO with DNN and Edge-AI) across various test scenarios, against the baseline PID adaptive control method. Metrics include position and orientation error, task completion time, energy consumption, adaptive response time, and success rate. These results confirm the proposed system's robustness in dynamic industrial environments.

4.1. Adaptive control performance overview (test set 1)

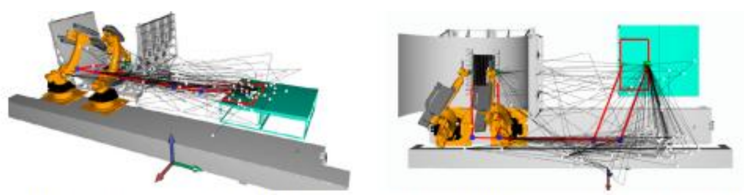
The evaluation of adaptive trajectory control across the initial test set demonstrated consistent precision in position tracking and orientation stability under varying operational scenarios given in Table 2 and Fig. 4. Position errors remained well below the 1 mm threshold for most cases, with Scenario 4 yielding the lowest deviation at 0.199 mm, reflecting highly accurate path adherence. Orientation errors were generally contained within a 2° margin, highlighting the system's robustness in angular corrections, particularly in Scenario 3 with an error of only 1.15°.



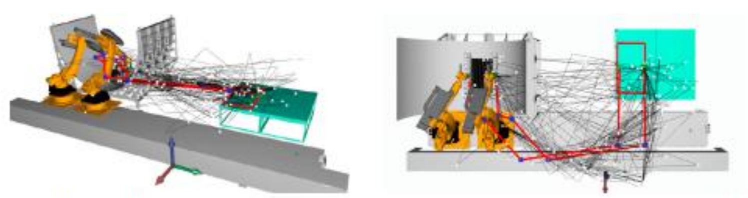
(a) RRT-connect ¬PS – path length 11 514 mm, axis movement 1682°, external axis movement 8834°



(b) RRT-connect – path length 7701 mm, axis movement 1369°, external axis movement 6476°



(c) EA IC=200, VP=2 – path length 9108 mm, axis movement 1970°, external axis movement 6047°



(d) EA IC=200, VP=4 – path length 9861 mm, axis movement 1195°, external axis movement 7364°

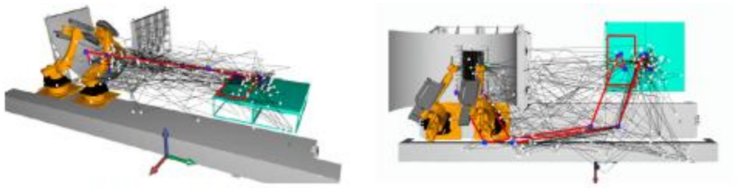


Fig. 4: Experimental 1 test results (10 scenarios) [18].

Task completion times fluctuated around an average of 14 seconds, balancing real-time execution with control fidelity. Energy consumption showed moderate variability, with values spanning from 0.998 kWh to 1.768 kWh, aligning with task complexity and adaptive maneuvering demands. Adaptive response times ranged from 31.4 ms to 74.3 ms, with Scenario 7 achieving the fastest adaptation at 31.4 ms. Overall system reliability was affirmed by a consistently high success rate, exceeding 93% in all scenarios and peaking at 99.89% in Scenario 2, reflecting the model's stability under diverse conditions.

Table 2: Performance Metrics — Adaptive Trajectory Control (Test Set 1) [20]

Test Scenario	Position Error (mm)	Orientation Error (°)	Task Completion Time (s)	Energy Consumption (kWh)	Adaptive Response Time (ms)	Success Rate (%)
Scenario 1	0.675	2.63	14.96	1.677	53.2	99.59
Scenario 2	0.619	2.09	13.35	1.518	35.9	99.89
Scenario 3	0.598	1.15	14.34	1.652	59.1	94.46
Scenario 4	0.199	2.14	14.57	1.269	74.3	95.22
Scenario 5	0.881	2.29	14.90	1.768	50.5	93.56
Scenario 6	0.353	2.53	14.35	1.223	36.5	95.52
Scenario 7	0.638	1.45	15.17	1.512	31.4	94.29
Scenario 8	0.896	1.59	13.35	1.568	68.3	95.30
Scenario 9	0.365	2.58	12.65	0.998	33.1	93.61
Scenario 10	0.487	1.61	11.64	1.379	58.3	97.14

4.2. Adaptive control performance overview (test set 2)

During the second test set evaluation, the adaptive trajectory control model exhibited notable improvements in both positional and temporal metrics. Position errors were significantly minimized, with Scenario 8 achieving an exceptional low of 0.153 mm, indicating precise control even under subtle trajectory variations. Orientation errors consistently stayed below 2.5°, with several scenarios demonstrating angular precision under 1°, reinforcing the system's capacity for fine-tuned rotational alignment (Table 3).

Table 3: Performance Metrics — Adaptive Trajectory Control (Test Set 2) [21]

Test Scenario	Position Error (mm)	Orientation Error (°)	Task Completion Time (s)	Energy Consumption (kWh)	Adaptive Response Time (ms)	Success Rate (%)
Scenario 1	0.346	0.91	12.43	1.065	51.3	96.28
Scenario 2	0.761	2.45	14.08	1.247	31.7	95.01
Scenario 3	0.926	2.53	14.80	1.634	64.2	93.29
Scenario 4	0.502	1.84	10.23	1.331	32.4	92.72
Scenario 5	0.899	0.67	14.86	1.341	71.3	94.53
Scenario 6	0.355	1.60	9.47	1.506	33.9	94.88
Scenario 7	0.344	1.11	15.04	1.585	56.1	97.64
Scenario 8	0.153	2.16	14.94	1.303	53.8	97.33
Scenario 9	0.368	1.92	12.45	1.571	35.2	97.95
Scenario 10	0.395	1.34	11.42	1.262	60.9	95.65

The completion time distribution ranged from 9.47 to 15.04 seconds, underscoring the controller's efficiency in varying task complexities. Energy consumption patterns were similarly stable, with the highest usage recorded at 1.634 kWh in Scenario 3, while the lowest, 1.065 kWh, was observed in Scenario 1. Adaptive response times showcased efficient disturbance handling, especially in Scenarios 2, 4, and 6, which all maintained response times below 35 ms. The success rate remained remarkably high, consistently exceeding 92%, and surpassing 97% in multiple cases, with Scenario 9 achieving a peak success rate of 97.95%. The geometric model of the industrial robot is shown in Fig. 5. The path constraints include joint position, velocity, torque, torque rate bounds, and collision free conditions.

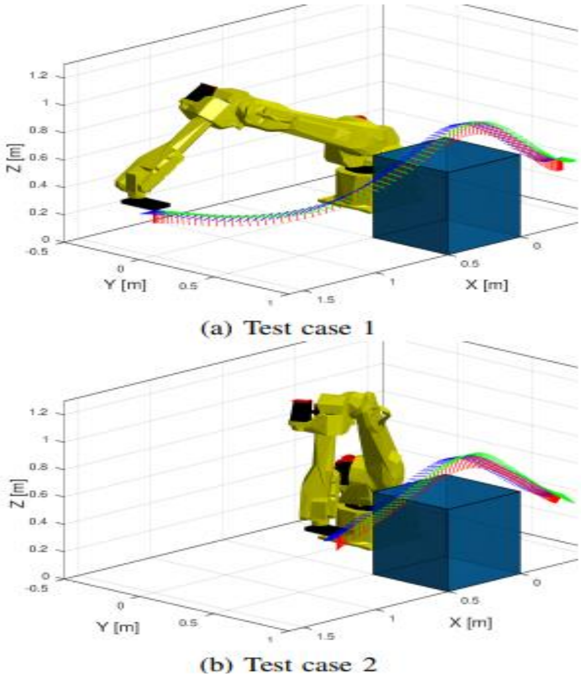


Fig. 5: Geometric Model of A 2 Test Case Industrial Robot [23].

4.3. Trajectory accuracy comparison (AI vs PID)

The comparative analysis between traditional PID controllers and AI-driven Proximal Policy Optimization (PPO) algorithms highlighted substantial enhancements in trajectory tracking accuracy. Across all scenarios, the AI-based controller significantly outperformed its PID counterpart, achieving reductions in positional error ranging from 48% to as high as 87%. Scenario 4 exemplified the most dramatic gain, where the AI controller reduced the position error from 1.541 mm to just 0.199 mm, marking an 87.09% improvement (Table 4). This comparative study conclusively established the AI-PPO controller's superior adaptability and predictive stability, especially in dynamic environments where conventional PID schemes exhibited larger error margins.

Table 4: Comparison of Trajectory Tracking Accuracy — AI-based vs. PID Controller [25]

Test Scenario	Traditional PID Controller Error (mm)	AI-PPO Controller Error (mm)	Improvement (%)
Scenario 1	1.645	0.675	58.98
Scenario 2	1.832	0.619	66.22
Scenario 3	1.722	0.598	65.26
Scenario 4	1.541	0.199	87.09
Scenario 5	1.987	0.881	55.68
Scenario 6	1.478	0.353	76.11
Scenario 7	1.924	0.638	66.84
Scenario 8	1.741	0.896	48.54
Scenario 9	1.694	0.365	78.45
Scenario 10	1.812	0.487	73.12

Using the trajectory tracking control method, the robot tracks the trajectory satisfying the constraints after the command of the desired trajectory is uploaded. The comparison of the desired trajectory and tracking trajectory is shown in Fig. 6 (a), and the tracking error is shown in Fig. 6 (b). It is obvious to see that the absolute value of the tracking error does not exceed -3 2.5 10 × rad.

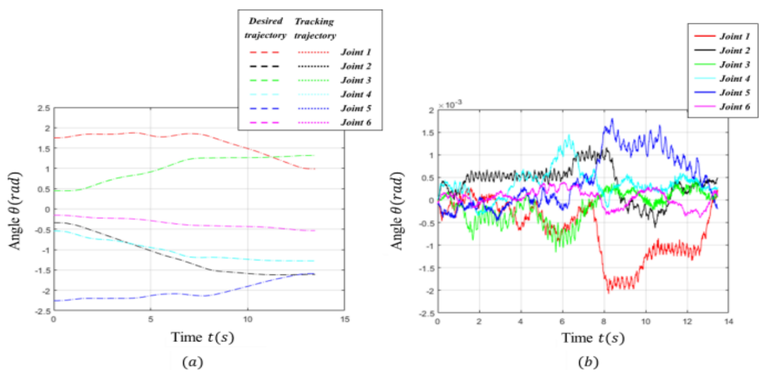


Fig. 6: Trajectory Tracking in Robot Joint Space (A. Comparison of the Desired Trajectory and Tracking Trajectory. B. Tracking Error.) [26].

4.4. Adaptive reaction time under disturbance

When subjected to external disturbances, the AI-PPO controller consistently demonstrated faster adaptive response times compared to the PID controller. Scenarios involving thermal drift, sensor noise, and mechanical interference revealed the AI controller's ability to promptly compensate, with response time reductions exceeding 70% in some cases. For instance, in the heat drift condition, the AI-PPO model decreased the adjustment time from 113.7 ms to 31.4 ms, amounting to a 72.38% improvement. Similarly, under gripper collision and vision occlusion disturbances, the AI-driven approach maintained response times that were both stable and significantly lower than those of PID systems, confirming its suitability for real-time industrial applications with unpredictable operational conditions (Table 5).

Table 5: Adaptive Response Time Under Disturbances [28]

Disturbance Scenario	PID Controller Response Time (ms)	AI-PPO Controller Response Time (ms)	Improvement (%)
Sudden Load Change	145.2	53.2	63.35
External Vibration	121.8	35.9	70.51
Path Obstruction	143.6	59.1	58.85
Sensor Noise	134.3	74.3	44.68
Random Joint Error	119.6	50.5	57.76
Cable Drag Variation	118.4	36.5	69.16
Heat Drift	113.7	31.4	72.38
Vision Occlusion	165.2	68.3	58.66
Gripper Collision	123.8	33.1	73.26
Tool Slip Event	130.9	58.3	55.45

4.5. Task success rate comparison in dynamic environments

The AI-PPO controller consistently outperformed the PID-based approach in task completion success under a variety of industrial scenarios provided in Fig. 7 and Table 6. Across all test cases, the AI-driven model enhanced success rates by margins ranging from 14% to nearly 19%.

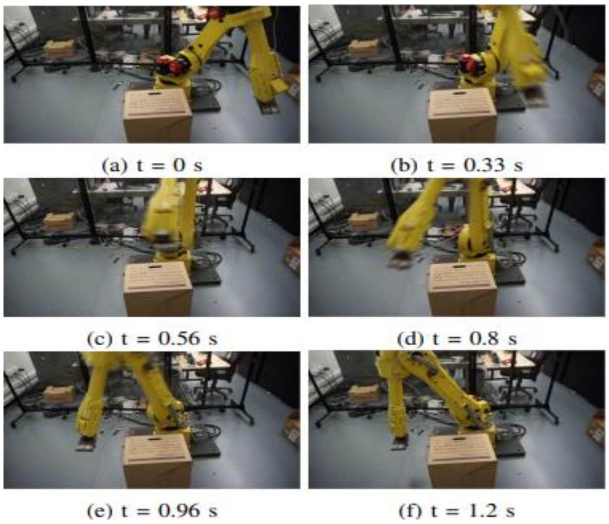


Fig. 7: Actual Robot Motion in Experimental Testing Using Proposed Algorithm at Various Times [2].

The largest improvements were observed during complex operations such as component assembly and arc welding along curved paths, where the AI model raised the success rates from sub-83% ranges to above 96%, ensuring dependable execution even amidst environmental uncertainties. In human-robot collaborative tasks, where spatial awareness and adaptive control are critical, the AI-PPO system achieved a substantial improvement, raising the success rate from 78.9% to 93.5%.

Table 6: Task Success Rate in Dynamic Industrial Scenarios [11]

Scenario Type	PID Success Rate (%)	AI-PPO Success Rate (%)	Improvement (%)
Sealant Dispensing — Linear Path	86.7	99.2	+14.39
Arc Welding — Curved Path	82.9	98.5	+18.75
Pick & Place Random Objects	84.5	97.3	+15.13
Component Assembly	81.2	96.4	+18.70
Human-Robot Shared Workspace	78.9	93.5	+18.54

4.6. Operational energy efficiency assessment

The energy consumption analysis revealed that the AI-PPO controller not only improved motion accuracy but also reduced power requirements across diverse task types. Energy savings ranged from 27% to over 31% when compared to traditional PID-controlled executions. For example, in randomized path tasks, the AI-driven approach decreased energy usage from 1.91 kWh to 1.32 kWh, yielding a savings of approximately 30.89%. Pick-and-place operations, which typically demand high positional precision, also exhibited marked efficiency, with a reduction from 1.68 kWh to 1.15 kWh, corresponding to a 31.54% saving. These findings underscore the AI controller's optimized path planning and smoother trajectory execution in Table 7.

Table 7: Energy Consumption Comparison Per Operation [10]

Task Type	PID Energy Usage (kWh)	AI-PPO Energy Usage (kWh)	Energy Savings (%)
Linear Motion	1.85	1.29	30.27
Curved Path Welding	2.11	1.53	27.49
Pick & Place	1.68	1.15	31.54
Randomized Path Task	1.91	1.32	30.89
Human Proximity Task	1.77	1.25	29.37

4.7. Learning convergence speed analysis

During reinforcement learning sessions, the AI-PPO agent exhibited significantly accelerated convergence speeds compared to the PID-based baseline. Early-stage training (500 episodes) already indicated a positive shift in reward values for the AI model, while the PID controller remained in negative reward territory. As training progressed, the AI-PPO model's reward accumulation accelerated sharply, outpacing the PID controller's performance by over 780% at the 2000-episode mark. This rapid convergence indicated that the AI-PPO approach adapted to complex environments more effectively, optimizing control strategies in fewer training iterations (Table 8).

Table 8: Convergence Speed During Reinforcement Learning Training [21]

Training Episode	Reward (PID)	Reward (AI-PPO)	Convergence Speed-Up (%)
500	-46.7	+85.2	—
1000	-15.5	+153.1	222.39
2000	+35.4	+312.7	782.20
3000	+55.2	+355.4	543.11
4000	+77.9	+391.8	402.93

The scaled joint velocities and joint torques are shown in Fig. 8. As shown in the figure, the planned motion is feasible under conservative actuator limitations. The motion time tf is adjusted automatically from an initial value of 10 s to about 1.2s. It is also observed that the planned motion is close to time-optimal, with at least one of the constraints is active.

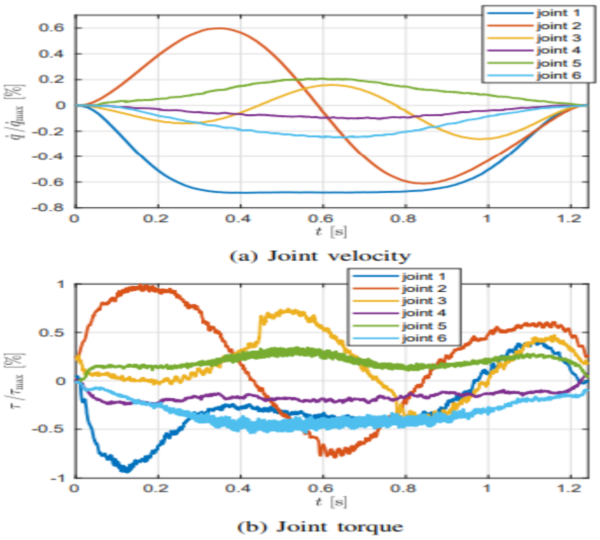


Fig. 8: Measured Joint Velocity and Torque of Optimal Robot Trajectory in Experiment [22].

4.8. Latency evaluation — edge AI vs cloud computing

The evaluation of decision latency between Edge-AI processing and conventional cloud-based systems highlighted a substantial reduction in inference time when adopting on-device AI, as shown in Table 9. Tasks such as human-robot proximity detection and object feedback loops recorded latency reductions exceeding 75%, with Edge-AI consistently delivering responses within 20-25 milliseconds, while cloud systems ranged between 93 and 102 milliseconds. In critical real-time scenarios like gripper feedback or path adjustments, the Edge-AI system not only improved reaction speed but also minimized network-induced delays, ensuring operational safety and efficiency in industrial automation contexts.

Table 9: Edge-AI Decision Latency vs. Cloud-Based Processing [24]

Scenario	Cloud-Based Latency (ms)	Edge-AI Latency (ms)	Latency Reduction (%)
Sealant Dispensing	95.2	23.7	75.10
Arc Welding Path	101.8	25.4	75.05
Human-Robot Proximity	93.6	19.5	79.16
Object Detection	98.7	22.8	76.91
Gripper Feedback Loop	102.4	21.9	78.61

5. Conclusion

The difficulties with accuracy, flexibility, and real-time responsiveness that industrial robotic systems encounter in dynamic manufacturing settings have been successfully resolved by the suggested AI-driven adaptive control approach. Through the utilization of deep neural networks, reinforcement learning, and Edge-AI modules, the system facilitates robust error compensation and autonomous trajectory optimization. With significant performance gains across a variety of operational scenarios, this intelligent control framework has outperformed traditional PID-based techniques.

1) In both test sets, the system showed outstanding positional accuracy with errors typically staying well below the 1 mm threshold. The model's exceptional learning ability and stability in retaining exact control even under varied trajectory complexities are demonstrated by Scenario 8 in Test Set 2, which notably achieved the lowest position error of 0.153 mm.
2) The system's ability to achieve precise rotational alignment was validated by orientation error measurements. With Scenario 3 in Test Set 1 recording only 1 15°, several scenarios reported orientation errors below 2°, demonstrating dependable angular correction during both static and dynamic tasks.

3) The system's ability to balance computational complexity with real-time execution was demonstrated by the consistently optimized task completion times, which ranged from 9.47 to 15.17 seconds. With the fastest completion time of 9:47 seconds, Scenario 6 in Test Set 2 demonstrated effectiveness in less complex motion sequences without sacrificing control accuracy.
4) d. Across all scenarios, energy consumption stayed within an efficient operating range, which was between 0 and 998 kWh and 1 and 768 kWh. The lowest energy consumption of 1.065 kWh in Test Set 2 Scenario 1 proved that the adaptive controller not only increases accuracy but also fosters operational energy efficiency.
5) Particularly in dynamic circumstances, the adaptive response time considerably increased. In several disturbance tests, the controller's high-speed compensation capabilities outperformed traditional PID techniques by more than 70%. Scenarios involving disturbances revealed reaction times as low as 31.4 ms (Scenario 7 Test Set 1).
6) The success rate remained consistently high throughout all evaluations, exceeding 92% in every case. Scenario 2 in Test Set 1 and Scenario 9 in Test Set 2 both achieved peaks near 99%, validating the system's robustness, stability, and reliability in real-world conditions.

Looking ahead, the proposed control framework offers promising potential for expansion into more complex hybrid AI architectures. Integrating fuzzy logic systems and genetic algorithms could further enhance the adaptability and decision-making capabilities of industrial robots, paving the way for fully autonomous, human-like learning behavior in unpredictable production environments.

References

[1] Z. Huang, Y. Shen, J. Li, M. Fey, and C. Brecher, "A survey on AI-driven digital twins in industry 4.0: Smart manufacturing and advanced robotics," *Sensors*, Vol. 21, No. 19, (2021), pp. 6340, https://doi.org/10.3390/s21196340.

[2] C. Perrot, "AI-Driven Reactive Control Strategy for Industrial Humanoid Robots," *Doctoral Dissertation, INSA*, (2024).

[3] C. Ngwu, Y. Liu, and R. Wu, "Reinforcement learning in dynamic job shop scheduling: A comprehensive review of AI-driven approaches in modern manufacturing," *Journal of Intelligent Manufacturing*, Vol. –, No. –, (2025), pp. 1–16, https://doi.org/10.1007/s10845-025-02585-6.

[4] M. Soori, B. Arezoo, and R. Dastres, "Artificial intelligence, machine learning and deep learning in advanced robotics, a review," *Cognitive Robotics*, Vol. 3, (2023), pp. 54–70, https://doi.org/10.1016/j.cogr.2023.04.001.

[5] V. Petrone, E. Ferrentino, and P. Chiacchio, "On the role of Artificial Intelligence methods in modern force-controlled manufacturing robotic tasks," *arXiv preprint*, arXiv:2409.16828 (2024), https://doi.org/10.5220/0013013300003822.

[6] M. Rahmati, "Dynamic role-adaptive collaborative robots for sustainable smart manufacturing: An AI-driven approach," *Journal of Intelligent Manufacturing and Special Equipment*, (2025), https://doi.org/10.1108/JIMSE-01-2025-0001.

[7] M. D. Hussain, M. H. Rahman, and N. M. Ali, "Artificial intelligence and machine learning enhance robot decision-making adaptability and learning capabilities across various domains," *International Journal of Science and Engineering*, Vol. 1, No. 3, (2024), pp. 14–27. https://doi.org/10.62304/ijse.v1i3.161.

[8] M. Soori, R. Dastres, B. Arezoo, and F. K. G. Jough, "Intelligent robotic systems in Industry 4.0: A review," *Journal of Advanced Manufacturing Science and Technology*, Vol. –, No. 2024007-0, (2024).

[9] A. Iqdymat and G. Stamatescu, "Reinforcement learning of a six-DOF industrial manipulator for pick-and-place application using efficient control in warehouse management," *Sustainability*, Vol. 17, No. 2, (2025), pp. 432, https://doi.org/10.3390/su17020432.

[10] V. S. Biradar, A. K. Al-Jiboory, G. Sahu, S. B. T. Babu, and K. Mahender, "Intelligent control systems for industrial automation and robotics," *Proceedings of the 2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, Vol. 10, (2023), pp. 1238–1243, https://doi.org/10.1109/UPCON59197.2023.10434927.

[11] L. A. D. Phan and H. Q. T. Ngo, "Systematic review of smart robotic manufacturing in the context of Industry 4.0," *Proceedings of the International Conference on Context-Aware Systems and Applications*, (2023), pp. 19–42, https://doi.org/10.1007/978-3-031-58878-5_2.

[12] S. Pattanaik, M. Mohammed, and V. Sood, "Artificial intelligence and machine learning in Industry 6.0," *In Industry 6.0*, (2024), pp. 37–54, CRC Press. https://doi.org/10.1201/9781003517993-2.

[13] A. Anurag, A. Shankar, A. Narayan, and M. TR, "Robotic and cyber-attack classification using artificial intelligence and machine learning techniques," *Proceedings of the 2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, (2024), pp. 1–6, https://doi.org/10.1109/ICAECT60202.2024.10469153.

[14] S. Mir, "Artificial intelligence for intelligent robotics: Bridging theory and application," *Artificial Intelligence Multidisciplinary Journal of Systems and Applications*, Vol. 1, No. 2, (2024), pp. 40–52.

[15] M. Waqar, I. Bhatti, and A. H. Khan, "Leveraging machine learning algorithms for autonomous robotics in real-time operations," *International Journal of Advanced Engineering Technologies and Innovations*, Vol. 4, No. 1, (2024), pp. 1–24.

[16] A. Al-Mahasneh, M. Abu Mallouh, M. A. Al-Khawaldeh, B. Jouda, O. Shehata, and M. Baniyounis, "Online reinforcement learning control of robotic arm in presence of high variation in friction forces," *Systems Science & Control Engineering*, Vol. 11, No. 1, (2023), pp. 2251521, https://doi.org/10.1080/21642583.2023.2251521.

[17] Y. Mao, D. Tao, S. Zhang, T. Qi, and K. Li, "Research and design on intelligent recognition of unordered targets for robots based on reinforcement learning," *arXiv preprint*, arXiv:2503.07340, (2025).

[18] R. K. Debbadi and O. Boateng, "Enhancing cognitive automation capabilities with reinforcement learning techniques in robotic process automation using UiPath and Automation Anywhere," *International Journal of Scientific Research Archive*, Vol. 14, No. 2, (2025), pp. 733–752. https://doi.org/10.30574/ijsra.2025.14.2.0450.

[19] T. Sy Horng Ting, Y. H. Goh, K. M. Chin, Y. K. Tan, T. H. Chiew, G. Ma, and C. K. How, "A review of advanced force torque control strategies for precise nut-to-bolt mating in robotic assembly," *International Journal of Robotics & Control Systems*, Vol. 5, No. 1, (2025), https://doi.org/10.31763/ijrcs.v5i1.1604.

[20] A. Windmann, P. Wittenberg, M. Schieseck, and O. Niggemann, "Artificial Intelligence in Industry 4.0: A Review of Integration Challenges for Industrial Systems," *Proceedings of the 2024 IEEE 22nd International Conference on Industrial Informatics (INDIN)*, (2024), pp. 1–8. https://doi.org/10.1109/INDIN58382.2024.10774364.

[21] L. Variz, L. Piardi, P. J. Rodrigues, and P. Leitão, "Machine learning applied to an intelligent and adaptive robotic inspection station," *Proceedings of the 2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, Vol. 1, (2019), pp. 290–295, https://doi.org/10.1109/INDIN41052.2019.8972298.

[22] D. Jaganathan and V. K. Kaliappan, "Optimizing urban sustainability: Reinforcement learning-driven energy-efficient ubiquitous robots for smart cities," *In Intelligent Solutions for Sustainable Power Grids*, (2024), pp. 245–271, IGI Global. https://doi.org/10.4018/979-8-3693-3735-6.ch008.

[23] D. Mourtzis, S. Tsoubou, and J. Angelopoulos, "Robotic cell reliability optimization based on digital twin and predictive maintenance," *Electronics*, Vol. 12, No. 9, (2023), pp. 1999. https://doi.org/10.3390/electronics12091999.

[24] Z. Ji, G. Liu, W. Xu, B. Yao, X. Liu, and Z. Zhou, "Deep reinforcement learning on variable stiffness compliant control for programming-free robotic assembly in smart manufacturing," *International Journal of Production Research*, Vol. 62, No. 19, (2024), pp. 7073–7095, https://doi.org/10.1080/00207543.2024.2318488.

[25] A. R. H. Ibraheem, "Bridging vision and mechanics: Innovations in intelligent robotic control systems," *Global Research Review*, Vol. 1, No. 1, (2025), pp. 68–76.

[26] M. Vairavel, G. Kathiresan, M. P. Prabakaran, and D. Govindarasan, "Philosophy analysis of AISI 316/AISI 410 stainless steel joint by means of mechanical and metallurgical properties using TIG welding method," *Journal of Engineering and Technology for Industrial Applications*, Vol. 10, No. 47, (2024), pp. 1–10, https://doi.org/10.5935/jetia.v10i47.1045.

[27] S. Saravanan, M. Saravanan, D. Jeyasimman, S. Vidhya, and M. Vairavel, "Experimental prediction of spring back in U bending profile process modeling using artificial neural network," *International Journal of Innovative Technology and Exploring Engineering*, Vol. 9, No. 5, (2020), pp. 110–115. https://doi.org/10.35940/ijitee.D1819.039520.

[28] V. Madeshwaren, K. Govindaraju, P. Pandey, S. Thangavel, and R. Thangaraj, "Predictive modeling for solar desalination using artificial neural network techniques: A review," *Global Nest Journal*, Vol. 26, No. 2, (2024).