Berger Scientfic Press

Innovation & Technology Advances, 2026, 4(1), 1-16
https://bergersci.com/index.php/jta/index
DOI: 10.61187/ita.v4i1.296

Motion Control of Flexible-Joint Robotic Arms for Variable-Station Warehouse Sorting Based on Proximal Policy Optimization

Xiaogang Zhu[1,*], Xiangzhen Pan[2], Xiaojing Gao[3]

[1]School of Architecture and Engineering, Yantai Institute of Technology, Yantai, Shandong, 264000, China
[2]School of Management Science and Engineering, Shandong Technology and Business University, Yantai, Shandong, 264005, China
[3]School of Architecture and Engineering, Qingdao Binhai University, Qingdao, Shandong, 266555, China

*Correspondence: zxg198812@126.com

Abstract: In variable-station warehouse sorting scenarios, the motion control of flexible-joint robotic arms must simultaneously address the triple-coupled challenges of flexible joint characteristics, dynamic workstation environments, and sorting task requirements, making arm motion control of the robotic arm highly complex. To address this, a motion control method for flexible-joint robotic arms in variable-station warehouse sorting is proposed, based on Proximal Policy Optimization (PPO algorithm). After analyzing the variable-station warehouse sorting model and the robotic arm's control system architecture, a motion control model based on Proximal Policy Optimization is constructed. This model maps the robotic arm as an agent, by designing a multidimensional state space that encompassing station coordinates and cargo status. It divides the action space into overall arm movement and end-effector rotation, establishing a reward function incorporating continuous rewards, sparse rewards, and penalties. An LSTM is introduced to capture temporal motion correlations, predicting advantageous function values under different actions as workstation coordinates change. The PPO algorithm obtains the robotic arm motion control commands with the highest cumulative reward value—such as angular velocity and torque for each joint, along with gripper opening degree (gripping force)—for robotic arm motion control. Experiments demonstrate that this method achieves position control errors as low as 0.1 mm and gripping force errors reduced to 0.05N for flexible-joint robotic arms in variable-workstation warehouse sorting. Sorting speeds reach 30 items per minute, meeting the high-precision and high-robustness control demands of variable-workstation warehouse sorting.

Keywords: Proximal policy optimization; Variable workstation; Warehouse sorting; Flexible joint; Robotic arm; Motion control

How to cite this paper: Zhu, X., Pan, X., Gao, X. Motion Control of Flexible-Joint Robotic Arms for Variable-Station Warehouse Sorting Based on Proximal Policy Optimization. *Innovation & Technology Advances*, 2026, *4*(1), 1–16. https://doi.org/10.61187/ita.v4i1.296

1. Introduction

As a core link in the supply chain, warehousing and logistics is undergoing a transformation from fixed-process sorting to dynamic workstation adaptation. With the explosive growth of industries such as e-commerce retail and intelligent manufacturing, warehousing scenarios have shown characteristics including diversified cargo specifications (from small parts boxes to medium-sized commodity containers), dynamic workstation layouts (flexible adjustments of multiple shelf layers and multiple storage location zones), and high requirements for sorting efficiency. Traditional rigid robotic arms, due to issues such as insufficient flexibility, poor adaptability to variable loads, and low precision in multi-workstation switching, can no longer meet the flexible and intelligent demands of modern warehouse sorting systems [1].

Flexible-joint robotic arms have the advantages of multiple degrees of freedom and low motion impact [2]. They can maintain stable functions under different external environmental conditions, scene changes, or interference factors [3], and are currently the core equipment used in variable-workstation warehousing sorting. However, in scenarios with multiple workstation changes and multiple cargo specifications, ensuring the stability of robotic arm control and successful sorting of items at multiple workstations is a current difficulty in motion control for such robotic arms [4].

A large number of studies have been carried out in academia to address the motion control issues of flexible robotic arms. Hernandez-Sanchez et al. proposed a control method for the end-effector of robotic arms based on dynamic average subgradient integral sliding mode control. By designing the sliding surface to suppress external disturbances, it improved the acceleration control accuracy at the end-effector of rigid-joint robotic arms, but lacked effective compensation for the time-varying elastic deformation of flexible joints, leading to error oscillations during multi-workstation switching [5]. Mozhi et al. proposed a bidirectional position control method for prismatic joints of electric single-arm robots based on adaptive super-twisting sliding mode control. Although it optimized the bidirectional position control performance of prismatic joints, its robust control framework has limited decoupling ability for force-position coupled systems, making it difficult to adapt to the grasping force requirements of goods of different sizes [6]. Leanza et al. proposed a nose-inspired multimodal deformation and motion control method for soft robotic arms. The bionic structure enhanced the morphological adaptability of soft robotic arms, but the bionic design principle has insufficient generalizability in engineering scenarios, and the motion efficiency decreases significantly when facing large-span variable workstations [7]. Benyamin et al. proposed a multimodal-based robotic arm control method, which expanded the stable trajectory range of the flexible arm through geometric frustration design, while retaining load-carrying capacity and shape reversibility. However, this method is limited to steady-state control under a single mode and cannot meet the dynamic strategy requirements of the entire "grasping-transporting-placing" process in warehousing sorting [8].

To solve the above problems, this paper proposes a motion control method for flexible-joint robotic arms in variable-workstation warehousing sorting based on proximal policy optimization (PPO). The core goals are to address three major issues: nonlinear compensation of flexible joints, dynamic adaptation to multiple workstations, and stable full-process strategies. The innovations of this paper are reflected in three aspects: first, constructing a control framework for precise mapping of "workstation-joint-cargo", deeply integrating the PPO algorithm with the warehousing sorting scenario; second, introducing LSTM to enhance temporal modeling capabilities, using its memory units and gating structures to capture the temporal characteristics of robotic arm motion, accurately depict long-term dependencies such as the impact of current workstation coordinate changes on subsequent grasping actions, and optimize the evaluation accuracy of the advantage function; third, ensuring the stability of strategy updates and scene generalization through the strategy update amplitude constraint mechanism of the PPO algorithm.

2. Motion Control of Flexible-Jointed Robotic Arms for Variable-Station Warehouse Sorting

2.1. Variable-Station Warehouse Sorting Model Based on Flexible-Jointed Robotic Arms

Figure 1 is a schematic diagram of the variable workstation warehousing and sorting mode based on the flexible-joint robotic arm.

This mode is composed of the material conveying subsystem, the flexible-joint robotic arm mobile execution subsystem, and the variable workstation shelf storage subsystem working collaboratively [9]. After materials are transported to the to-be-grabbed area via the conveyor belt, the flexible-joint robotic arm mounted on the mobile cart completes

the material grabbing action with the help of laser ranging and network communication technologies [10]; then, according to the dynamic requirements of the sorting task, it autonomously moves to the variable workstations corresponding to multiple shelves to achieve precise placement of materials [11], forming a full-process operation link of "material conveying - flexible grabbing - multi-workstation dynamic sorting". **Figure 2** is the control system structure of the flexible-joint robotic arm for variable workstation warehousing and sorting.

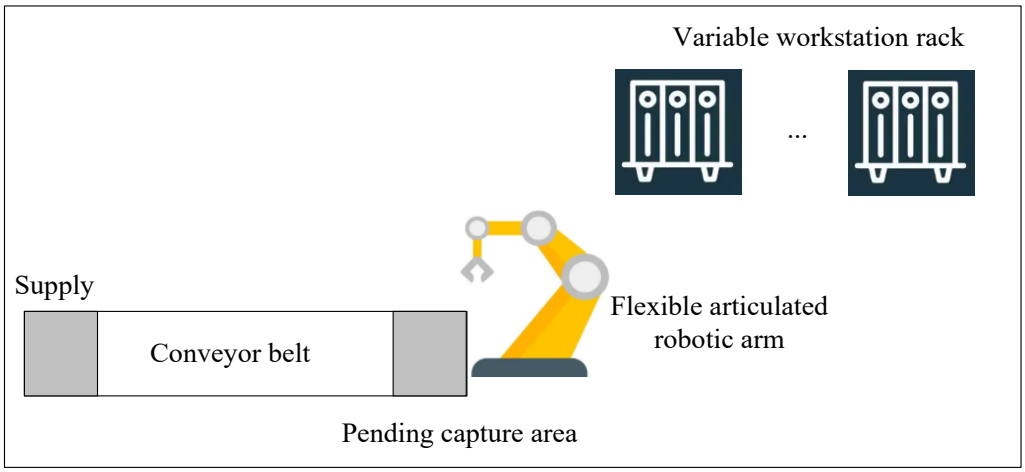

Figure 1. Variable workstation warehousing and sorting mode based on flexible-joint robotic arm.

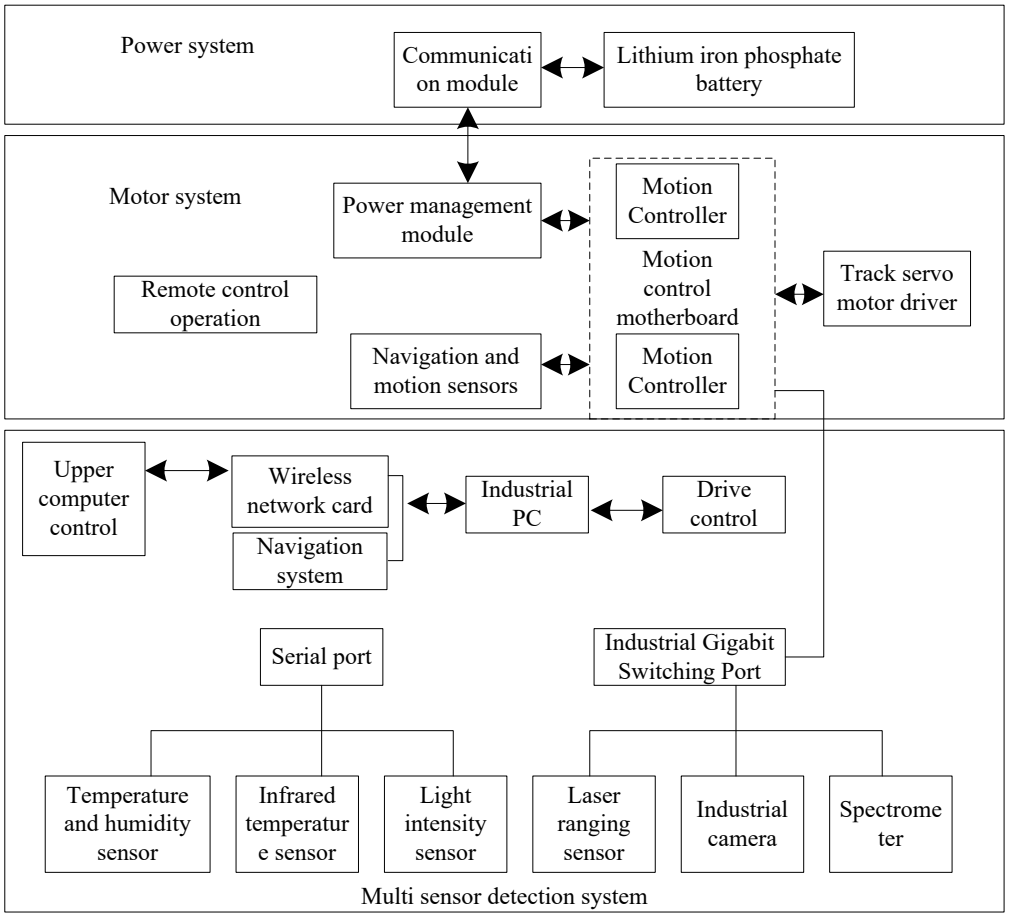

Figure 2. Control system structure of flexible-joint robotic arm for variable workstation warehousing and sorting.

The system adopts a hierarchical architecture comprising "multi-sensor detection," "motion control," and "power supply" layers. At the multi-sensor detection system layer, it integrates environmental sensors such as temperature and humidity, infrared temperature measurement, and light intensity (data summarized via serial port), as well as workstation and material detection equipment such as laser ranging sensors, visible light cameras, spectrometers, and infrared cameras (information transmitted through industrial gigabit switch ports); the industrial computer (Intel i5 processor, 8GB memory, 128GB hard disk) serves as the core control unit, realizing upper-computer control through a wireless network card and navigation system, and issuing instructions to the motion control module [12]. The motion system layer takes an ARM-core motion control motherboard as the core, matched with a motion controller, navigation and motion sensors, and realizes the motion drive of the robotic arm in combination with a track servo motor driver; it also supports a remote control operation mode, and the power management module supplies power to each unit [13]. The power system layer is powered by lithium iron phosphate batteries, which interact with the power management module through a communication module to ensure the continuous operation of the system.

Through multi-sensor fusion to perceive the environment and workstation status, this architecture realizes precise motion control of the flexible-joint robotic arm in variable workstation scenarios relying on hierarchical control logic, providing hardware support for the intelligence and flexibility of warehousing sorting [14].

2.2. Motion Control Model for Flexible Articulated Robotic Arms Based on Proximity Policy Optimization

2.2.1. Core concepts and scenario mapping

In the proximity policy optimization algorithm, the agent is defined as the flexible-jointed robotic arm. The state r represents sensor-collected information, including real-time coordinates of variable workstations, cargo dimensions/orientation, angles of each flexible joint, and sorting task progress (grabbed, pending placement). The action c denotes motion control commands for the robotic arm, such as angular velocity and torque for each joint, along with gripper opening degree (gripping force) [15]. $\left(\beta(c|r,\alpha)\right)$ is the "motion decision model" for the robotic arm, which combines the current state r to output the probability distribution of optimal actions c. A reword o is the feedback for sorting tasks, specifically whether items were successfully grasped and accurately placed at stations [16].

(1) State Space Design for Robotic Arm Variable-Station Warehouse Sorting

In the control of flexible-joint robotic arms for variable-station warehouse sorting, the multi-segmented and highly flexible nature of such arms results in a more complex state representation compared to rigid robotic arms [17]. However, leveraging the regularity of segmented structures (e.g., equal segment lengths), the coordinates of remaining joints can be derived based on the initial end-effector coordinates and the number of segments [18].

Based on a three-dimensional perspective analysis, during variable-station warehousing and sorting, the initial point coordinates of the flexible-joint robotic arm are set as follows: $Q_0(x_0, y_0, z_0)$. The number of segments and their respective lengths are m and l. Then, the coordinates of the j th joint node $Q_j(x_j, y_j, z_j)$ is:

$$\begin{cases} x_j = x_0 + jl\cos\varepsilon_{jx} \\ y_j = y_0 + jl\cos\varepsilon_{jy} \\ z_j = z_0 + jl\cos\varepsilon_{jz} \end{cases} \tag{1}$$

Where, $cos\,\varepsilon_{jx}$, $cos\,\varepsilon_{jy}$ and $cos\,\varepsilon_{jz}$ represent the cosines of the rotation angles of the joint around the x axis, y axis and z axis, respectively. The end-effector coordinates of the robotic arm are $Q_m(x_m, y_m, z_m)$.

During variable-station warehousing and sorting, the rotational directions of each joint encompass the x axis, y axis and z axis, resulting in three rotational direction angles per joint. These rotational direction angles correspond to the angles formed between the line connecting the starting point and the joint, and the coordinate axes [19]. Calculate the position of the connecting line and compute its angles with the x,y and z axes. If the coordinates of the flexible joint manipulator are $Q_j(x_j, y_j, z_j)$ and the origin coordinates are $Q_0(x_0, y_0, z_0)$, then the position vector of the connecting line between the two points is:

$$p_j = (x_j - x_0, y_j - y_0, z_j - z_0) \tag{2}$$

During variable-station warehousing and sorting, to obtain the position vector and the angle between the three coordinate axes and the positive direction, calculate the cosine value of the joint rotation angle:

$$\begin{cases} cos\,\varepsilon_{jx} = \dfrac{x_j - x_0}{\sqrt{(x_j - x_0)^2 + (y_j - y_0)^2 + (z_j - z_0)^2}} \\ cos\,\varepsilon_{jy} = \dfrac{y_j - y_0}{\sqrt{(x_j - x_0)^2 + (y_j - y_0)^2 + (z_j - z_0)^2}} \\ cos\,\varepsilon_{jz} = \dfrac{z_j - z_0}{\sqrt{(x_j - x_0)^2 + (y_j - y_0)^2 + (z_j - z_0)^2}} \end{cases} \tag{3}$$

To enable the PPO algorithm to effectively control the robotic arm and update the agent's state in real-time based on environmental rewards, a specific input state$r$must be designed: introduce the three-dimensional motor drive signal g to indicate state updates, and introduce the Boolean variable ∂ (1 for successful grasping, 0 otherwise) to determine the completion status of the sorting task. Ultimately, the state space is represented by Equation (4). This design enhances the robotic arm's responsiveness to changes in the warehouse environment and improves control precision, ensuring the PPO algorithm effectively evaluates and updates sorting strategies to achieve efficient sorting under variable workstation conditions.

$$r = [Q_0, Q_1, \ldots, Q_m, m, l, g, \partial] \tag{4}$$

(2) Action Space Design for Robotic Arm Variable-Station Warehouse Sorting

The motion actions of the flexible-joint robotic arm for variable-station warehouse sorting are composed of the axis vectors$x$,$y$and$z$. Assuming the positive half-axis motion vectors for$x$,$y$and$z$are$(x_m, 0,0)$,$(0, y_m, 0)$and$(0,0, z_m)$, respectively, the motion vector of the flexible-joint robotic arm is:

$$c_m = (x_m, y_m, z_m) \tag{5}$$

The motion space design for variable-work-cell sorting can be categorized into two types of movements:

Global movement c_N: Relocating the entire robotic arm near the target goods to position itself at an appropriate starting point, preparing for subsequent precise sorting. The expression is $c_N = (x_N, y_N, z_N)$.

End-effector rotation motion c_S: After the robotic arm reaches the vicinity of the goods, precise control of the end-effector rotation Q_m drives the rear segments to complete the grasping action, expressed as $c_S = (x_S, y_S, z_S)$.

For the end-effector rotation action c_S, the robotic arm's end-effector $Q_m(x_m, y_m, z_m)$ rotates counterclockwise. The rotation pivot point must be centered on the previous joint node. The rotation angle is set to ε, and the post-rotation coordinates are $\tilde{Q}_m(\tilde{x}_m, \tilde{y}_m, \tilde{z}_m)$. The rotation methods for the x axis, y axis and z axis are respectively:

$$\begin{pmatrix} \tilde{x}_m \\ \tilde{y}_m \\ \tilde{z}_m \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\,\varepsilon & -sin\,\varepsilon \\ 0 & sin\,\varepsilon & cos\,\varepsilon \end{pmatrix} \begin{pmatrix} x_m \\ y_m \\ z_m \end{pmatrix} \tag{6}$$

$$\begin{pmatrix} \tilde{x}_m \\ \tilde{y}_m \\ \tilde{z}_m \end{pmatrix} = \begin{pmatrix} cos\,\varepsilon & 0 & sin\,\varepsilon \\ 0 & cos\,\varepsilon & 0 \\ -sin\,\varepsilon & 0 & cos\,\varepsilon \end{pmatrix} \begin{pmatrix} x_m \\ y_m \\ z_m \end{pmatrix} \tag{7}$$

$$\begin{pmatrix} \tilde{x}_m \\ \tilde{y}_m \\ \tilde{z}_m \end{pmatrix} = \begin{pmatrix} cos\,\varepsilon & -sin\,\varepsilon & 0 \\ sin\,\varepsilon & cos\,\varepsilon & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_m \\ y_m \\ z_m \end{pmatrix} \tag{8}$$

The maximum rotation angle for each joint node is set to ε_{max}. If the absolute value of the rotation angle $|\varepsilon| = \varepsilon_{max}$ for any coordinate axis at the end effector Q_m, the end effector's active rotation terminates. The preceding joint node Q_{m-1} then rotates. Rotation ends when the rotation angle ε_{max} of the subsequent joint node Q_1 at the starting end reaches.

The resulting output motion of the flexible joint robotic arm is:

$$c = \{c_N, c_S\} \tag{9}$$

(3) Designing the reward function o

Set the total reward value o to include o_1, o_2 and o_3:

$$\begin{cases} o_1 = e^{-0.99\hbar} - 1 \\ o_2 = \begin{cases} 1 & \text{Sorting successful} \\ -1 & \text{Sorting failed} \end{cases} \\ o_3 = -12 \end{cases} \tag{10}$$

Where, o_1 is a distance-related continuous reward function related to the distance $\hbar$ (the arithmetic mean of the Euclidean distances between the robot arm's starting point and end point to the two target cargo pickup points); o_2 is a sparse reward function, set to 1 for successful sorting and -1 for failure; o_3 is a penalty function to prevent unproductive operations (e.g., moving in directions that increase distance from both ends).

2.2.2. Adversarial objective function and advantage function design

The adversarial objective function is defined as $A^c(\alpha)$:

$$A^c(\alpha) = E_t\left[min\left(o_t(\alpha) \cdot \hat{B}_t\right), clip(o_t(\alpha), 1-\chi, 1+\chi) \cdot \hat{B}_t\right] \tag{11}$$

Where, $o_t(\alpha) = \frac{\beta_\alpha(c_t|r_t)}{\beta_{\alpha o}(c_t|r_t)}$ denotes the probability ratio between the new strategy α and the old strategy αo under the state r_t and action c_t; $\hat{B}_t$ represents the advantage function, evaluating the quality of the current action relative to the average; $clip(o_t(\alpha), 1-\chi, 1+\chi)$ is the function limiting the magnitude of strategy updates, preventing the robotic arm from losing control due to abrupt strategy changes.

$$\hat{B}_t = \sum_{j=1}^{T} \delta^{j-t} o_j - U(r_t) \tag{12}$$

Where, $\sum_{j=1}^{T} \delta^{j-t} o_j$ is the cumulative discounted reward from the current time step t to the task completion time T; $U(r_t)$ is the state value function.

2.2.3. LSTM-based advantage function computation model

In variable-station warehouse sorting scenarios, the robot arm's state (station coordinates, joint deformations, cargo orientation) constitutes sequential continuous data. LSTMs capture dynamic temporal evolution through memory units and gating structures (input gate, forget gate, output gate), providing precise state representations with historical context for the advantage function.

Let the robot arm's state at time step t be r_t. Inputting r_t into the LSTM-based advantage function computation model calculates the hidden state and memory cell state k_t (the robot arm's hidden state, resulting from the memory cell state activated by the output gate and $tanh$, containing key temporal features at the current time step for subsequent reward or advantage function computation), and λ_t (fusing historical memory with new current information, storing long-term temporal state information):

$$\begin{cases} \phi_t = \eta\left(\omega_\phi r_t + \varpi_\phi k_{t-1} + \Gamma_\phi\right) \\ \Lambda_t = \eta(\omega_\Lambda r_t + \varpi_\Lambda k_{t-1} + \Gamma_\Lambda) \\ \lambda_t = \phi_t \otimes (\lambda_{t-1} + tan\hbar(\omega_\lambda r_t + \varpi_\lambda k_{t-1} + \Gamma_\lambda)) \otimes \Lambda_t \\ \Phi_t = \eta(\omega_\Phi r_t + \varpi_\Phi k_{t-1} + \Gamma_\Phi) \\ k_t = \lambda_t \cdot tan\hbar(\lambda_t) \end{cases} \tag{13}$$

Where, ϕ_t and Λ_t are the LSTM forget gate (controlling the retention ratio of the memory cell state λ_{t-1} at the previous time step) and input gate (controlling the input ratio of new information at the current time step); Φ_t is the output gate (controlling the output ratio of λ_t to k_t); η and $tanh$ are the Sigmoid activation function and hyperbolic tangent function; ω_ϕ, ω_Λ, ω_λ, ω_Φ are the weight matrices of each gate in LSTM, used for linear transformation of input or historical states to capture the correlation between features; ϖ_ϕ, ϖ_Λ, ω_λ, ϖ_Φ are the historical state weight matrices of each gate in LSTM, used for linear transformation of λ_t or k_t at the previous time step to transmit historical information; Γ_ϕ, Γ_Λ, Γ_λ, Γ_Φ are the bias terms of each gate in LSTM, used to adjust the result of linear transformation and increase the expressive flexibility of the model.

The advantage function $\hat{B}_t$ is used to evaluate "how good or bad the current action is relative to the average level". The reward sequence o_t of the robotic arm is time-series data. LSTM can encode the state sequence $\{r_t\}$ and the reward sequence $\{o_t\}$ simultaneously. Through the long-term memory characteristic of memory cells, it can more accurately predict the cumulative discounted reward "from the current moment to the end of the task". The state value function $U(r_t)$ adopts the LSTM structure, taking the hidden state k_t output by LSTM as input, and outputting the long-term value $U(r_t)$ of the current state. The gating mechanism of LSTM can effectively capture the long-term dependence of states (such as "the impact of current workstation changes on subsequent sorting actions"), making the evaluation of the value function more accurate.

The calculation process of the advantage function fused with LSTM is:

(1) Input the robot arm's state sequence $\{r_t\}$ into the LSTM to obtain a hidden state sequence k_t containing temporal information;

(2) Combining LSTM's temporal modeling of the reward sequence, compute the cumulative discounted reward $\sum_{j=1}^{T} \delta^{j-t} o_j$ from time steps t to T;

(3) Obtain the final advantage function value via Equation (12).

2.2.4. Motion control process for flexible-jointed robotic arms based on proximal policy optimization

In the motion control of flexible-joint robotic arms, the steps for proximal policy optimization are as follows:

Step 1: Collect samples. The robotic arm executes the current policy in the warehouse environment, gathering data on the arm's variable workstation coordinates and cargo status (r). The controller outputs joint motion and gripper open/close commands (c). Sensors

provide feedback on grasping results and collision events (o). Store this data as training samples ((r, c, o)).

Step 2: Use LSTM to predict the advantage function $\hat{B}_t$ for each sample.

Step 3: Calculate the objective by substituting into the adversarial objective function $A^c(\alpha)$, ensuring controllable divergence between the new and old policies [20]. If the robot arm's action probability under the "grabbing tilted cargo" state increases excessively with the new policy compared to the old one, the $clip$ function limits the update magnitude to prevent abrupt extreme movements.

Step 4: Update the policy by optimizing the LSTM network parameters via gradient descent to maximize $A^c(\alpha)$. If a set of output joint movements and gripper open/close commands yields a higher "cumulative reward for successful sorting at variable workstations," retain and progressively refine it. This ultimately yields an optimal (highest cumulative reward) robotic arm motion decision scheme.

3. Experimental Analysis

3.1. Experimental Environment

In the variable workstation warehousing and sorting environment shown in **Figure 3**, the motion control effect of the proposed method on the flexible-joint robotic arm is tested. In this environment, the industrial camera is responsible for collecting image information of sorted items and sending it to the computer through the image transmission link; the flexible-joint robotic arm undertakes the item sorting execution task; the conveyor belt realizes the transportation of items, on which sampling points are set for initial information collection and separation points are set for triggering item sorting actions; the PLC (Programmable Logic Controller) is responsible for connecting the control logic of the robotic arm, conveyor belt and computer, forming a closed-loop control architecture of "perception - decision-making - execution". Through this experimental environment, key performance indicators such as precision, real-time performance and robustness of the proposed method in motion control of the flexible-joint robotic arm in warehousing sorting scenarios can be systematically tested. **Figure 4**shows the actual operation scenario of the robotic arm, and **Table 1** presents the parameter information of the robotic arm.

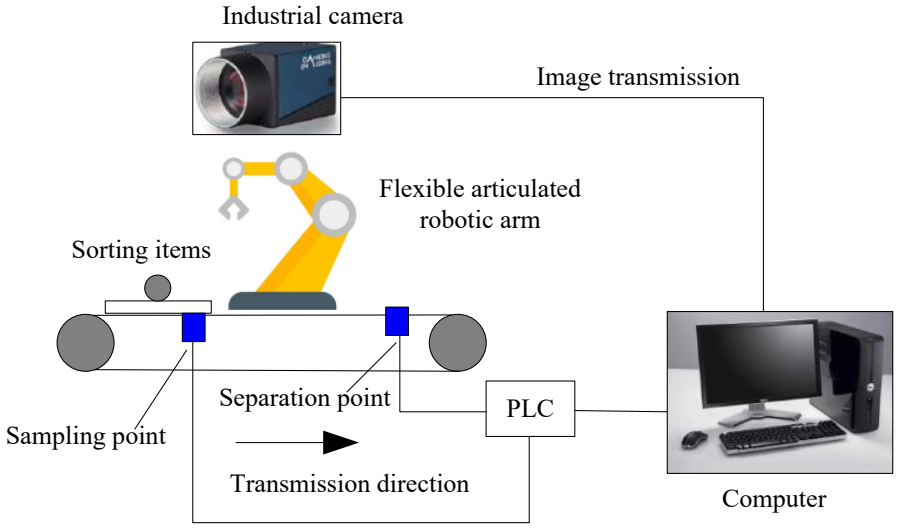

Figure 3. Experimental environment diagram of variable workstation warehouse sorting.

Figure 4. Actual working scenario of robotic arm.

Table 1. Parameters of the robotic arm.

Joint coding	Joint angle /°	Connecting rod offset /cm	Connecting rod length /cm	Torsion angle /°
1	180	9.86	0	0
2	-90	12.16	0	-90
3	0	0	40.9	180
4	-90	0	37.7	180
5	0	10.26	0	-90
6	0	9.5	0	90

3.2. Robotic Arm Motion Control Experiment

3.2.1. Motion control data and analysis for flexible-jointed robotic arm in variable-station warehouse sorting

In the experiment, the sorting area of a large e-commerce warehousing center was simulated, with 10 variable workstations set up. Their 3D coordinates range from (500, 300, 150) mm to (1400, 1200, 600) mm (details shown in **Table 2**), covering different shelf layers and storage location zones. Ten types of common e-commerce packages were selected, with sizes ranging from 100*80*50mm mm (small parts boxes) to 320*250*140mm mm (medium-sized commodity containers), simulating diverse sorting objects in warehousing. The proposed method was used for robotic arm motion control, and the proximal policy optimization (PPO) algorithm adopted in this method had a fixed strategy update amplitude constraint value of 0.2 to ensure the stability of strategy iteration. The robotic arm needed to move from the initial pose to 10 variable workstations in sequence to complete the full-process sorting of "grabbing-transporting-placing" goods of corresponding sizes, with each workstation repeating the task 10 times. To further evaluate the control performance of this method, the following metrics were defined for the motion control

scenario of a flexible-joint robotic arm in variable-station warehouse sorting: position control error: τ_1, gripping force control error: τ_2, sorting speed: τ_3.

$$\tau_1 = \left|x_s^{'} - x_s\right| + \left|y_s^{'} - y_s\right| + \left|z_s^{'} - z_s\right| \tag{14}$$

$$\tau_2 = \left|\Xi_s^{'} - \Xi_s\right| \tag{15}$$

$$\tau_3 = \frac{\ddot{N}}{t} \tag{16}$$

Where, (x_s', y_s', z_s') and (x_s, y_s, z_s) represent the desired sorting and grasping position and the position coordinates after control by the proposed method, respectively; Ξ_s' and Ξ_s represent the desired sorting and grasping force and the sorting and grasping force after control by the proposed method, respectively; $\ddot{N}$ denotes the number of sorted items.

The control performance of the proposed method is shown in **Table 2**.

Table 2. Motion control results of flexible joint robot arm for variable workstation warehousing and sorting.

Variable workstation coordinates (X-axis, Y-axis, Z-axis)/mm	Size of goods (length * width * height)/mm	Accumulated discount rewards	LSTM hidden state temporal correlation coefficient	Policy update amplitude constraint value	Position control error /mm	Grasping force control error /N	Sorting speed/(pieces/min)
(500,300,150)	100*80*50	125.6	0.62	0.2	2.5	1.2	8
(600,400,200)	120*90*60	189.3	0.71	0.2	1.8	0.9	10
(700,500,250)	150*100*70	256.8	0.78	0.2	1.3	0.7	12
(800,600,300)	180*120*80	321.5	0.83	0.2	0.9	0.5	15
(900,700,350)	200*150*90	385.2	0.87	0.2	0.6	0.3	18
(1000,800,400)	220*160*100	452.7	0.91	0.2	0.4	0.2	20
(1100,900,450)	250*180*110	518.3	0.93	0.2	0.3	0.15	22
(1200,1000,500)	280*200*120	586.9	0.95	0.2	0.2	0.1	25
(1300,1100,550)	300*220*130	653.4	0.96	0.2	0.15	0.08	28
(1400,1200,600)	320*250*140	721.8	0.98	0.2	0.1	0.05	30

Analysis of the data in **Table 2** shows that the position control error gradually decreases from 2.5 mm at the initial workstation (500, 300, 150) mm to 0.1 mm at the workstation (1400, 1200, 600) mm. This is because the Proximal Policy Optimization (PPO) algorithm can accurately compensate for the nonlinear elastic deformation of flexible joints. Under the iterative update of policy gradients, it can still move accurately to the ideal position in variable workstations with large spatial spans, meeting the core requirement of "precise goods placement" in warehousing sorting.

The grasping force control error gradually decreases from 1.2N to 0.05N. According to the logic shown in Formula (15) earlier, this result reflects the decoupling control capability of the PPO algorithm for the "force-position coupled system". It can avoid damage to goods (such as fragile items and lightweight packages) due to excessive grasping force, or goods falling off due to insufficient grasping force, ensuring sorting reliability.

The sorting speed increases from 8 pieces/min to 30 pieces/min, which is strongly correlated with the increasing trend of cumulative discounted reward (125.6 to 721.8). This reflects the optimization logic of the proposed method in the trade-off between "efficiency and precision": the sparse reward design of the PPO algorithm (such as giving positive rewards for completing sorting tasks and penalties for position deviations) promotes the iterative optimization of the robotic arm's action sequence. With the increase in training

steps, the proportion of invalid actions of the robotic arm in the "grabbing-transporting-placing" process continues to decrease, and the action fluency is significantly improved. In addition, the improvement of sorting speed also benefits from LSTM's ability to model temporal states. The temporal correlation coefficient of LSTM hidden states increases from 0.62 to 0.98, indicating that its accuracy in capturing temporal information such as "workstation coordinate changes - joint angle evolution - goods size adaptation" of the robotic arm continues to improve. This enables the robotic arm to predict the optimal path of the action sequence in advance and reduce motion delay.

The strategy update amplitude constraint value (fixed at 0.2) is the core guarantee for the "stability" of the PPO algorithm. By limiting the probability difference between the new strategy and the old strategy, it avoids the risk of robotic arm out of control caused by sudden strategy changes, ensuring the convergence of the algorithm in complex scenarios with 10 variable workstations and 10 types of goods sizes.

In terms of scene generalization ability, all indicators of the proposed method show a continuous optimization trend in the combined test of "large-span workstations + diversified goods". This indicates that the proposed method has good adaptability to dynamic changes in warehousing sorting scenarios (such as workstation layout adjustments and goods specification updates), providing feasibility for practical engineering deployment.

3.2.2. Analysis of position and speed of flexible joints of robotic arm

In the variable workstation warehousing sorting experimental environment, the expected flexible joint position curve (**Figure 5**) is set as the motion control target, covering the position change trajectory of each joint in the time dimension of 0-18s, simulating the joint motion requirements under multi-workstation switching such as item grabbing, transfer, and placement in warehousing sorting. Before the experiment, the joint speed curve (**Figure 6**) is collected. After control using the proposed method, the joint position curve is shown in **Figure 7**, and the joint speed curve after motion control is shown in **Figure 8**.

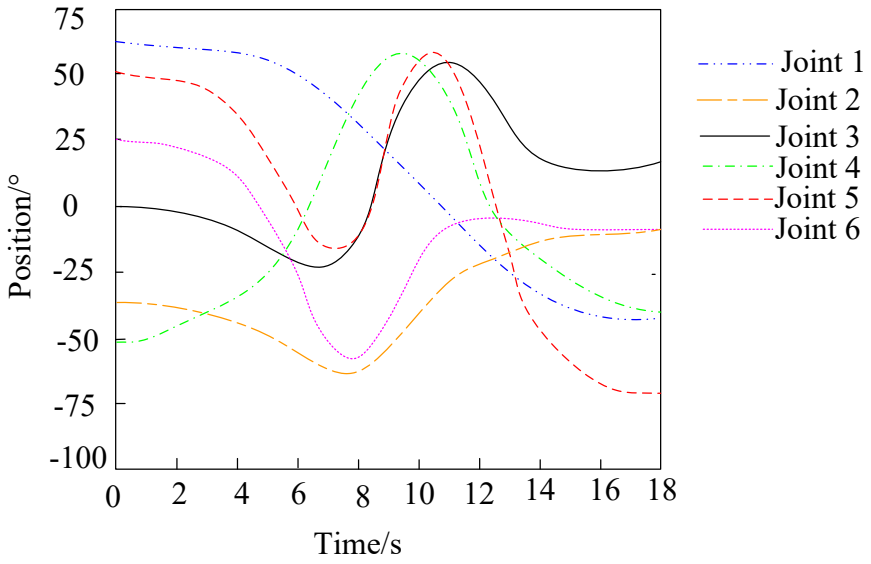

Figure 5. Expected flexible joint position curve.

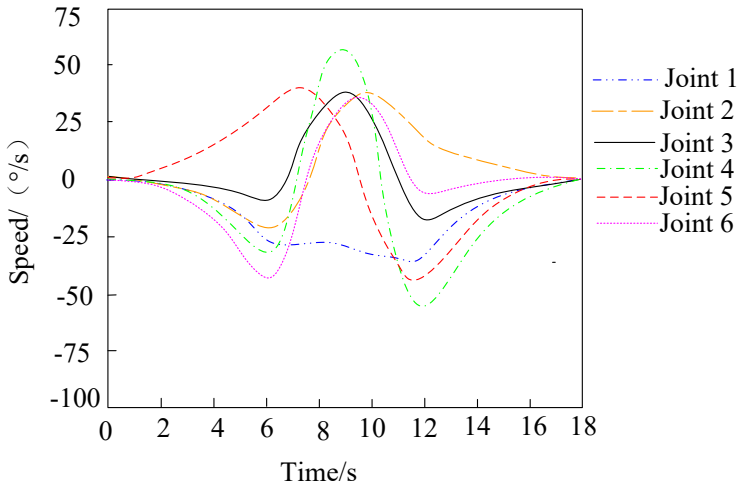

Figure 6. The velocity curve of the front joint controlled by the method described in this article.

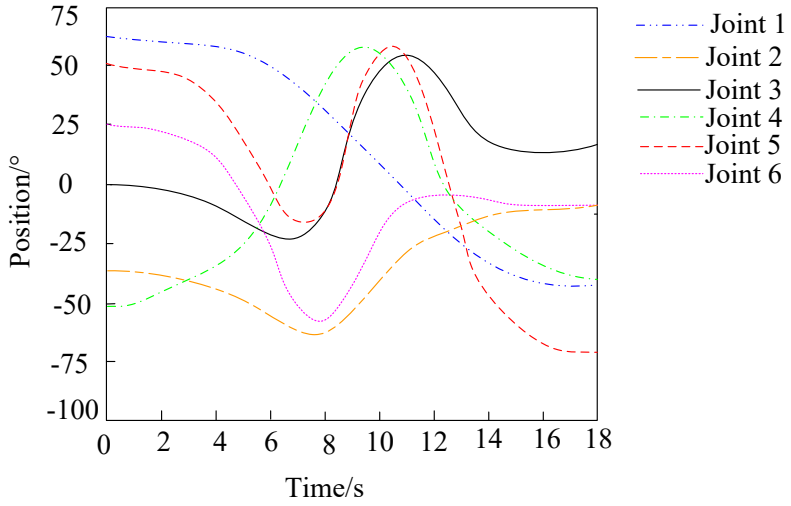

Figure 7. Joint position curve after motion control.

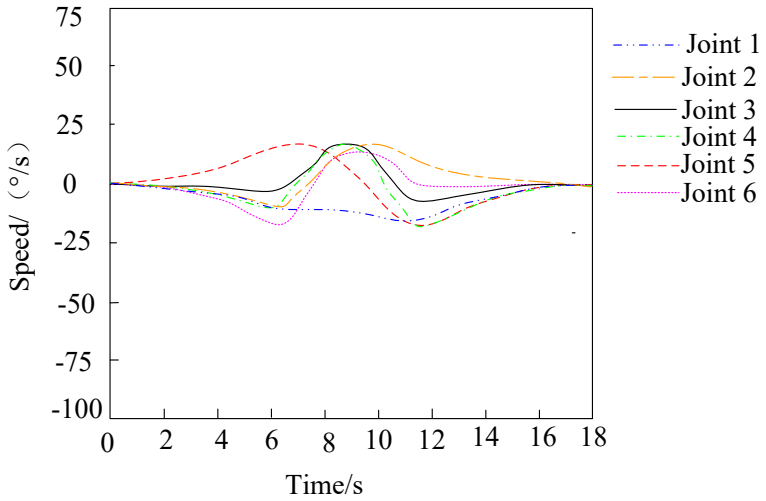

Figure 8. Joint velocity curve after motion control.

Comparing **Figure 5** (expected flexible joint position curve) with **Figure 7** (joint position curve after motion control), it can be seen that the actual position trajectory of each joint (Joints 1-6) almost coincides with the expected trajectory. For example, key nodes such as the position peak of Joint 3 at around 10s and the position valley of Joint 6 at around 8s are accurately fitted to the expected curve after control. This indicates that the proposed method has extremely high position tracking accuracy for multiple flexible joints, which can meet the position control requirements during multi-workstation switching such as item grabbing, transfer, and placement in variable workstation warehousing sorting.

Comparing **Figure 6** (joint speed curve before control) with **Figure 8** (joint speed curve after motion control), it can be seen that before control, the speed of each joint fluctuates sharply. For example, the speed peak of Joint 4 is greater than 50°/s, and there are obvious sudden changes in the speed curve. After control, the smoothness of the speed curve of each joint is greatly improved, the absolute values of peaks and valleys are significantly reduced, and the speed of all joints is controlled within a small fluctuation range. This reflects the smooth control ability of the proposed method for joint speed, which can effectively reduce the motion impact of the robotic arm and improve equipment life and sorting stability.

3.2.3. End-effector position control error of different methods

In the variable workstation warehousing sorting experimental environment (as shown in **Figure 9**), three comparative control methods are introduced for the robotic arm equipped with flexible joints: Comparative Method 1 (control method for robotic arm end-effector based on dynamic average subgradient integral sliding mode control), Comparative Method 2 (bidirectional position control method for prismatic joints of electric single-arm robots based on adaptive super-twisting sliding mode control), and Comparative Method 3 (nose-inspired multimodal deformation and motion control method for soft robotic arms). A comparative experiment on the motion control error of the robotic arm end position is conducted with the proposed method. The experiment takes the position control accuracy of the robotic arm end-effector within 0~18s as the core indicator, and the collected and analyzed motion control errors of the robotic arm end position under different methods are shown in the curves in **Figure 9**.

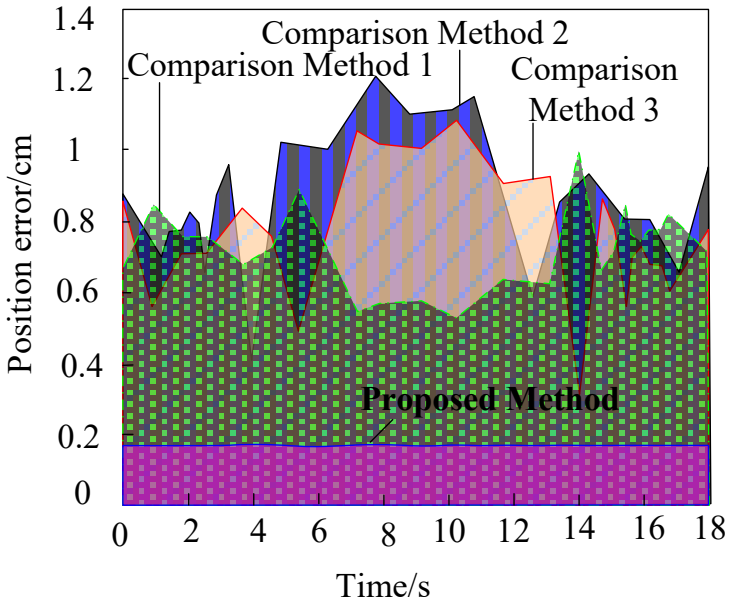

Figure 9. Comparison of motion control errors of robotic arm end positions using different methods.

It can be seen from the position error curves in **Figure 9** that the position error of the proposed method is always stable within 0.2 cm, while the error peaks of Comparative Method 1, Comparative Method 2, and Comparative Method 3 reach 1 cm, 1.2 cm, and 1.1 cm respectively, which are larger than that of the proposed method. This difference stems from the different modeling capabilities of each method for the nonlinear dynamic characteristics of flexible joints: Comparative Method 1 (dynamic average subgradient integral sliding mode control) and Method 2 (adaptive super-twisting sliding mode control) belong to the traditional robust control system. Although they can suppress disturbances through sliding surface design, they lack accurate characterization of the complex time-varying characteristics of "elastic deformation-inertial coupling" of flexible joints, leading to large error oscillations during the workstation switching phase of 6-12s (when the robotic arm motion trajectory changes sharply). Although Comparative Method 3 (nose-inspired soft robot multimodal control) is optimized for flexible bodies, the bionic control logic of "nose inspiration" has insufficient generalization in engineering scenarios. When facing the working conditions of "large-span workstations + multi-specification goods loads" in warehousing sorting, the error stability is significantly weaker than that of the proposed method.

The proposed method iteratively optimizes the motion strategy of the robotic arm in the high-dimensional action space through the PPO algorithm, and at the same time uses LSTM to deeply capture the dynamic correlation of "workstation coordinates-joint angles-time series", realizing adaptive compensation for the "time-varying and strongly coupled" characteristics of flexible joints. This data-driven approach enables it to maintain error convergence throughout the 0-18s period, especially in the high-frequency action phase of 14-16s (corresponding to the response demand for rapid sorting of small-sized goods), where the error does not fluctuate at all.

4. Conclusion

This paper proposes a motion control method for flexible-joint robotic arms in variable workstation warehousing sorting, which integrates the Proximal Policy Optimization (PPO) algorithm and Long Short-Term Memory (LSTM) network to realize the motion control of flexible-joint robotic arms in variable workstation warehousing sorting. Verified by experimental data, after using the proposed method, the robotic arm position control error is reduced to 0.1 mm, and the grasping force control error is reduced to 0.05N. Compared with Comparative Method 1 (control method for robotic arm end-effector based on dynamic average subgradient integral sliding mode control), Comparative Method 2 (bi-directional position control method for prismatic joints of electric single-arm robots based on adaptive super-twisting sliding mode control), and Comparative Method 3 (nose-inspired multimodal deformation and motion control method for soft robotic arms), the end position error of the robotic arm controlled by the proposed method is always stable within 0.2 cm. It is more valuable in the scenarios of multiple variable workstations and multi-specification goods loads in warehousing sorting. The proposed method overcomes the limitations of traditional methods with poor generalization and only applicable to a single scene, verifying its good adaptability to dynamic changes in warehousing sorting scenarios.

In the sorting tasks of 10 variable workstations and 10 types of goods of different sizes, the proposed method increases the sorting speed from 8 pieces/min at the initial workstation to 30 pieces/min at the farthest workstation, and the cumulative discounted reward shows a monotonically increasing trend (from 125.6 to 721.8). This indicates that the algorithm continuously optimizes the action sequence during training, reduces invalid movements, and improves task completion efficiency. At the same time, the strategy update amplitude constraint mechanism of the PPO algorithm (fixed at 0.2) effectively ensures the stability of the training process and avoids the risk of control failure caused by sudden strategy changes.

This study deeply integrates reinforcement learning (PPO) with deep learning (LSTM), constructing an intelligent decision-making framework for closed-loop control of "state-action-reward", realizing end-to-end optimization from perception to execution. This framework is not only applicable to warehousing sorting but also provides a referable technical path for the control problems of other flexible robotic arms in high-dynamic and multi-task environments.

References

1. Abdelghani, D., Mohamed, Z. A. Nouara, A. A Novel Stepper Motor Haptic Interface for Efficient Robotic Task Programming. Journal Europeen des Systemes Automatises, 2024, 57(5), 1369-1376. https://doi.org/10.18280/jesa.570512
2. Parvin, M., Jafar, M., Keyvan, A. V. Robotic date fruit harvesting using machine vision and a 5-DOF manipulator. Journal of Field Robotics, 2023, 40(6), 1408-1423. https://doi.org/10.1002/rob.22184
3. Steafan, E. K. Zachary, C. D. Continuous Gesture Control of a Robot Arm: Performance Is Robust to a Variety of Hand-to-Robot Maps. IEEE Transactions on Biomedical Engineering, 2024, 71(3), 944-953. https://doi.org/10.1109/TBME.2023.3323601
4. Sinha, A. K., Thalmann, N. M. Cai, Y. Measuring Anthropomorphism of a New Humanoid Hand-Arm System. International Journal of Social Robotics, 2023, 15(8), 1341-1363. https://doi.org/10.1007/s12369-023-00999-x
5. Hernandez-Sanchez, A., Chairez, I., Poznyak, A., et al. Cueing end-effector acceleration of a two-link robotic arm by dynamic averaged sub-gradient integral sliding mode control. Asian Journal of Control: Affiliated with ACPA, the Asian Control Professors Association, 2023, 25(4), 2577-2587. https://doi.org/10.1002/asjc.2994
6. Mozhi, G. T., Sundareswari, M. B. Dhanalakshmi, K. Dhanalakshmi. Bidirectional Position Control of a Prismatic joint for Motorized Single Link Robotic Arm Using Adaptive Super- Twisting Sliding Mode Control. Journal of The Institution of Engineers (India), Series B. Electrical eingineering, electronics and telecommunication engineering, computer engineering, 2023, 104(5), 1035-1042. https://doi.org/10.1007/s40031-023-00908-w
7. Leanza, S., Juliana, L. Y., Kaczmarski, B., et al. Elephant Trunk Inspired Multimodal Deformations and Movements of Soft Robotic Arms. Advanced functional materials, 2024, 34(29),2400396.1-2400396.10. https://doi.org/10.1002/adfm.202400396
8. Benyamin, S., Hossein, M., Arian, S., et al. Programmable Shape-Preserving Soft Robotics Arm via Multimodal Multistability. Advanced functional materials, 2025, 35(6), 2407651.1-2407651.16. https://doi.org/10.1002/adfm.202407651
9. Lancaster, P., Mavrogiannis, C., Srinivasa, S., et al. Electrostatic brakes enable individual joint control of underactuated, highly articulated robots. The International Journal of Robotics Research, 2024, 43(14), 2204-2220. https://doi.org/10.1177/02783649241250362
10. Miroslav, M., Milena, K., Vladimir, P., et al. Optimizing the Position of a Robotic Arm Using Statistical Methods. Manufacturing Technology, 2024, 24(4),618-625. https://doi.org/10.21062/mft.2024.073
11. Marco, B., Gianluca, R., Luca, Z., et al. Lightweight Human-Friendly Robotic Arm Based on Transparent Hydrostatic Transmissions. IEEE Transactions on Robotics: A publication of the IEEE Robotics and Automation Society, 2023, 39(5),4051-4064. https://doi.org/10.1109/TRO.2023.3290310
12. Hichame, T., Mohamed, E. H., Daachi, T. M. Real-time adaptive super twisting algorithm based on PSO algorithm: application for an exoskeleton robot. Robotica: International journal of information, education and research in robotics and artificial intelligence, 2024, 42(6),1816-1841. https://doi.org/10.1017/S0263574724000547
13. Schorr, L., Cobilean, V., Mavikumbure, H. S., et al. Industrial workspace detection of a robotic arm using combined 2D and 3D vision processing. The International Journal of Advanced Manufacturing Technology, 2025, 136(3/4), 1317-1326. https://doi.org/10.1007/s00170-024-14901-0
14. Islem, K., Abdelhak, B., Mohamed, T. Enhancing pose estimation for mobile robots: A comparative analysis of deep reinforcement learning algorithms for adaptive Extended Kalman Filter-based estimation. Engineering Applications of Artificial Intelligence, 2025, 150(Jun.),110548.1-110548.22. https://doi.org/10.1016/j.engappai.2025.110548
15. Kim, D., Choi, M., Um, J. Digital twin for autonomous collaborative robot by using synthetic data and reinforcement learning. Robotics and Computer Integrated Manufacturing: An International Journal of Manufacturing and Product and Process Development, 2024, 85(Feb.),102632.1-102632.13. https://doi.org/10.1016/j.rcim.2023.102632
16. Moslem, M., Abbas, Z. K., Mahdi, B., et al. Sustainable Robotic Joints 4D Printing with Variable Stiffness Using Reinforcement Learning. Robotics and Computer Integrated Manufacturing: An International Journal of Manufacturing and Product and Process Development, 2024, 85(Feb.), 102636.1-102636.12. https://doi.org/10.1016/j.rcim.2023.102636
17. Mani, A., Reza, A. Intelligent ergonomic optimization in bimanual worker-robot interaction: A Reinforcement Learning approach. Automation in construction, 2024, 168(Dec. Pt.A),105741.1-105741.13. https://doi.org/10.1016/j.autcon.2024.105741
18. Anh, V. L., Dinh, T. V., Nguyen, T. D., et al. Complete coverage planning using Deep Reinforcement Learning for polyiamonds-based reconfigurable robot. Engineering Applications of Artificial Intelligence, 2024, 138(Dec. Pt.B), 109424.1-109424.14. https://doi.org/10.1016/j.engappai.2024.109424
19. Pablo, R. B., Matteo, S., Angeliki, K., et al. Neutrons Sensitivity of Deep Reinforcement Learning Policies on EdgeAI Accelerators. IEEE Transactions on Nuclear Science, 2024, 71(8 Pt.1),1480-1486. https://doi.org/10.1109/TNS.2024.3387087

20. Chen, G., Peng, Y., Zhang, M. An adaptive clipping approach for proximal policy optimization. arXiv preprint arXiv:1804.06461, 2018. https://doi.org/10.48550/arXiv.1804.06461