frontiers | Frontiers in Neurorobotics

TYPE Original Research
PUBLISHED 22 February 2024
DOI 10.3389/fnbot.2024.1362359

OPEN ACCESS

EDITED BY
Alois C. Knoll,
Technical University of Munich, Germany

REVIEWED BY
Weida Li,
Soochow University, China
Chen Chen,
Harbin University of Science and Technology, China

*CORRESPONDENCE
Pengfei Wang
✉ wangpengfei@hit.edu.cn
Lining Sun
✉ lnsun@hit.edu.cn

RECEIVED 28 December 2023
ACCEPTED 05 February 2024
PUBLISHED 22 February 2024

CITATION
Liang K, Zha F, Guo W, Liu S, Wang P and Sun L (2024) Motion planning framework based on dual-agent DDPG method for dual-arm robots guided by human joint angle constraints. *Front. Neurorobot.* 18:1362359. doi: 10.3389/fnbot.2024.1362359

Motion planning framework based on dual-agent DDPG method for dual-arm robots guided by human joint angle constraints

Keyao Liang, Fusheng Zha, Wei Guo, Shengkai Liu, Pengfei Wang* and Lining Sun*

State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin, China

Introduction: Reinforcement learning has been widely used in robot motion planning. However, for multi-step complex tasks of dual-arm robots, the trajectory planning method based on reinforcement learning still has some problems, such as ample exploration space, long training time, and uncontrollable training process. Based on the dual-agent depth deterministic strategy gradient (DADDPG) algorithm, this study proposes a motion planning framework constrained by the human joint angle, simultaneously realizing the humanization of learning content and learning style. It quickly plans the coordinated trajectory of dual-arm for complex multi-step tasks.

Methods: The proposed framework mainly includes two parts: one is the modeling of human joint angle constraints. The joint angle is calculated from the human arm motion data measured by the inertial measurement unit (IMU) by establishing a human-robot dual-arm kinematic mapping model. Then, the joint angle range constraints are extracted from multiple groups of demonstration data and expressed as inequalities. Second, the segmented reward function is designed. The human joint angle constraint guides the exploratory learning process of the reinforcement learning method in the form of step reward. Therefore, the exploration space is reduced, the training speed is accelerated, and the learning process is controllable to a certain extent.

Results and discussion: The effectiveness of the framework was verified in the gym simulation environment of the Baxter robot's reach-grasp-align task. The results show that in this framework, human experience knowledge has a significant impact on the guidance of learning, and this method can more quickly plan the coordinated trajectory of dual-arm for multi-step tasks.

KEYWORDS

trajectory planning, reinforcement learning, dual-agent depth deterministic strategy gradient, human experience constrains guidance, motion parameter mapping

1 Introduction

In recent years, more and more researchers have paid attention to dual-arm robots, which are more similar to human beings in terms of configuration, joint freedom, and working space. They can better replace human tasks by imitating human arms (Wang et al., 2022). Compared with single-arm robots, dual-arm robots have significant advantages in precision assembly with high

coordination and multi-object assembly in unstructured environments. Because of the high degree of freedom and high coordination of the dual-arm robot, it requires high dimensions and strong coupling for its motion planning. In particular, for multi-step complex tasks, the motion process of the dual-arm robot can be divided into multiple sub-task processes, which increases the dimension of motion planning from the task planning level. The traditional motion planning method mainly solves dual-arm robots' obstacle avoidance motion planning problem using constraint model establishment and non-linear solutions (Vahrenkamp et al., 2012; Fang et al., 2015; Giftthaler et al., 2017). However, this method has little effect on dual-arm robots' multi-step coordination tasks, which limits the application of dual-arm robots.

With the development of machine learning methods, more and more researchers use intelligent learning methods to complete the motion planning of multi-step coordination tasks of dual-arm robots (Bing et al., 2022a, 2023b,d). Learning-based motion planning methods are mainly divided into imitation learning and reinforcement learning. The motion planning method based on imitation learning learns the motion features from the teaching demonstration and then reproduces the demonstration task on the robot. Maeda et al. (2020) proposed a demonstration programming method that automatically derives task constraints from data for constraint-based robot controllers using the Dirichlet Process Gaussian Mixture Model (DPGMM) and Gaussian Mixture Regression (GMR) method. In the study by Mronga and Kirchner (2021), phase portrait movement primitives (PPMPs), which can predict the dynamics of the low-dimensional phase space and then can be used to control the high-dimensional kinematics of the task, were proposed. In the study by Dong et al. (2022), a model-based learnable graph attention network (GAT) was used to learn task-level skills from human demonstration passively. It was validated in a humanoid robot task experiment of waving and grasping boxes. This category method can realize human imitation from the level of learning content and effectively learn human motion knowledge, but it can not optimize or learn new trajectories independently.

The motion planning method based on reinforcement learning enables the agent to explore learning motion strategies by interacting with the environment (Bing et al., 2022b, 2023a; Chu et al., 2022). For example, Ren and Ben-Tzvi (2020) proposed an advising reinforcement learning approach based on the depth deterministic strategy gradient (DDPG) and hindsight experience replay (HER), which applies the teacher-student framework to a continuous control environment with sparse rewards to solve the problem of extended agents. In the study by Jiang et al. (2021), a multiagent twin delayed deep deterministic policy gradient (MATD3) algorithm was proposed for the on-orbit acquisition mission of a space robot arm to generate a real-time inverse kinematics solution for the coordinated robot arm. In the study by Tang et al. (2022), the proximal policy optimization (PPO) algorithm with continuous rewards was used for trajectory planning of the two-arm robot, and the reward and punishment function was designed based on the artificial potential field (APF) method so that the dual-arm robot could approach and support patients in a complex environment. This category method imitates human beings from the level of learning style and mimics the human trial and error reward learning mechanism. However, the high-dimensional problem of trajectory planning brought by dual-arm multi-step tasks which will make the search space of reinforcement learning larger, the training process easily falls into local optimal, and the training results are difficult to converge.

In the previous study, the DADDPG algorithm proposed could reduce and decouple the dual-arm trajectory planning problem to a certain extent and successfully plan the dual-arm coordination trajectory for multi-objective tasks (Liang et al., 2023). Based on the DADDPG algorithm, this study proposes a motion planning framework guided by the human joint angle constraints, which simultaneously realizes the human-like learning content and learning style. By introducing human joint angle constraint, this method reduces the exploration space of reinforcement learning, rationalizes its exploration, makes its learning controllable to a certain extent, and speeds up the learning speed.

Building a robot's structure or control algorithm by imitating humans or animals has long been one of the potential means of improving robot performance (Bing et al., 2023c). The bionics-based human-like arm motion planning method extracts biomarkers and rules from recorded movements for simulating arm motion trajectory (Gulletta et al., 2020). For example, Kim et al. proposed a method to extract human arm movement features from the motion capture database, characterize human arm movement according to elbow elevation angle, and use this representation to generate human-like movements in real-time (Kim et al., 2006; Shin and Kim, 2014). In the study by Suárez et al. (2015), a motion planning method for a dual-arm anthropomorphic system was proposed, and a new basis vector of the dual-arm configuration space returned by principal component analysis (PCA) was used to characterize the dual-arm synergy. In this study, the human joint angle is calculated from the demonstration data collected by IMU and mapped to the robot model. Then, the joint angle constraint is extracted piecewise from the multi-group human demonstration and used to guide the autonomous learning of the multi-step coordination trajectory of dual-arm robots.

Three existing learning optimization methods use empirical knowledge to guide reinforcement learning: reward function optimization, exploration behavior optimization, and network parameter initialization (Taylor et al., 2011; Bougie et al., 2018; Xiang and Su, 2019). Among them, optimizing reward function is the most consistent with human behavior patterns. It models the reward function of the reinforcement learning method based on the empirical knowledge model, which can guide reinforcement learning intuitively and effectively (Tian et al., 2021). The segmented guided step reward of this study is designed to make the joint angle constraints guide the DADDPG method to quickly learn the dual-arm coordination trajectory for complex multi-step tasks.

2 Methodology

This study proposes a motion planning framework for dual-arm robots guided by human joint constraints based on the DADDPG method, as shown in Figure 1. In the proposed framework, the joint angle is calculated from the demonstration data collected by IMU and mapped to the robot model. Then, the joint angle constraint is extracted piecewise from multiple groups of human demonstration. The joint angle constraint is then

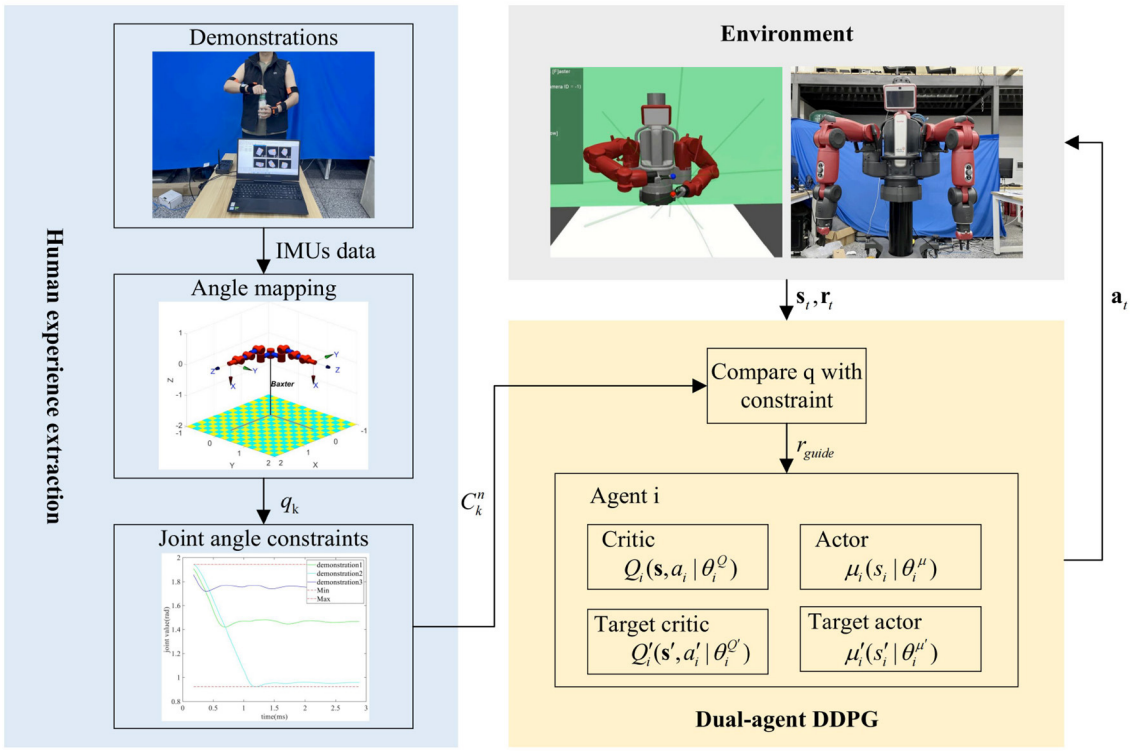

FIGURE 1
The proposed framework block diagram.

TABLE 1 D-H parameters of the human arm(right).

Item	θ_i	d_i	a_i	α_i	$offset_i$
1	q_1	0	0	$\frac{\pi}{2}$	π
2	q_2	0	0	$\frac{\pi}{2}$	$-\frac{\pi}{2}$
3	q_3	l_1	0	$\frac{\pi}{2}$	π
4	q_4	0	0	$\frac{\pi}{2}$	π
5	q_5	l_2	0	$\frac{\pi}{2}$	π
6	q_6	0	0	$\frac{\pi}{2}$	π
7	q_7	l_3	0	0	0

l_1 is the distance from the shoulder to the elbow, l_2 is the distance from the elbow to the wrist, and l_3 is the distance from the wrist to the palm.

TABLE 2 D-H parameters of Baxter arm(right).

Item	θ_i	d_i	a_i	α_i	$offset_i$
1	q_1	0.27	0.069	$-\frac{\pi}{2}$	0
2	q_2	0	0	$\frac{\pi}{2}$	$\frac{\pi}{2}$
3	q_3	0.364	0.069	$-\frac{\pi}{2}$	0
4	q_4	0	0	$\frac{\pi}{2}$	0
5	q_5	0.375	0.01	$-\frac{\pi}{2}$	0
6	q_6	0	0	$\frac{\pi}{2}$	0
7	q_7	0.28	0	0	0

used to guide the DADDPG method to quickly learn the dual-arm coordination trajectory for complex multi-step tasks through reward distribution. The following is a detailed introduction from four aspects: joint mapping model, joint angle constraint, reinforcement learning method, and reward guidance design.

2.1 Human-robot joint mapping model

The human arm has three joints: shoulder, elbow, and wrist. Among them, the shoulder joint has three rotational degrees of freedom, the elbow joint has two rotational degrees of freedom, and the wrist joint has two rotational degrees of freedom. The kinematics model of the human arm and Baxter is established using the standard D-H method (Denavit and Hartenberg, 1955). The parameters are shown in Tables 1, 2. q_1, q_2, q_3 is the rotation angle corresponding to the shoulder joint, q_4, q_5 is the rotation angle corresponding to the elbow joint, and q_6, q_7 is the rotation angle corresponding to the wrist joint. Input the same joint angle value, and the posture of the two models is consistent, as shown in Figure 2. Figure 2A is the simplified kinematics model of the human arms, and Figure 2B is the kinematics model of the Baxter robot.

The following describes how to solve the corresponding joint angle from the demonstration data measured by the IMU. The presenter wears six IMUs, as shown in Figure 3A, three for each

10.3389/fnbot.2024.1362359

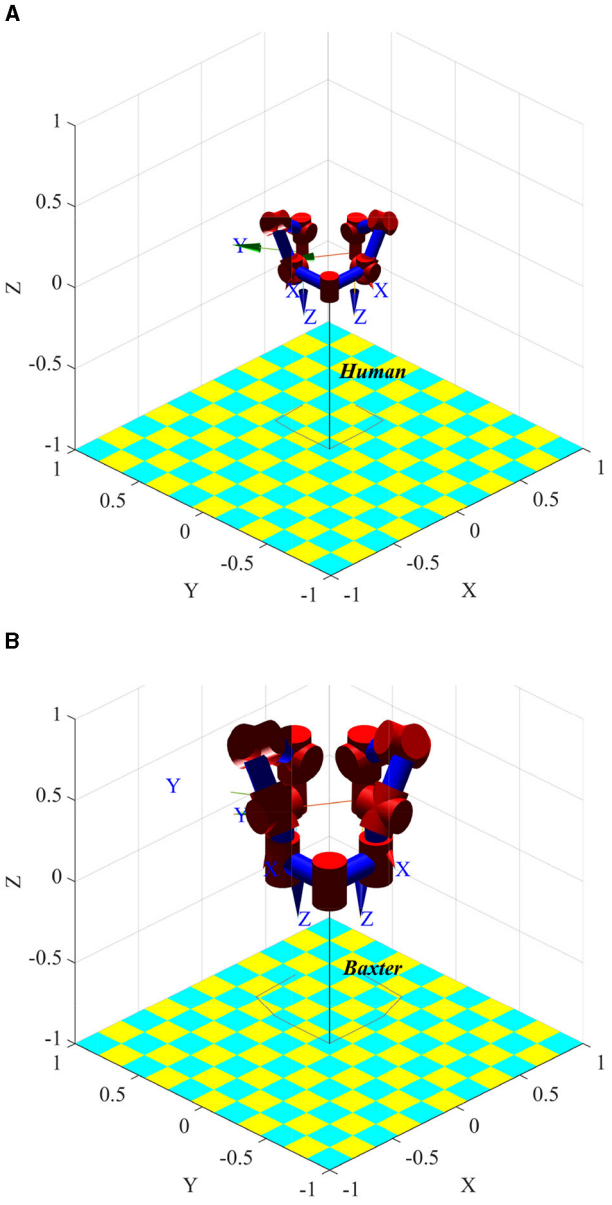

FIGURE 2
Comparison of human and robot pose when joint angle $ql = qr = [-0.078, -0.968, -1.150, 1.923, 0.648, 1.008, -0.483]$ is set. **(A)** Simplified kinematic model of human arms. **(B)** Kinematic model of Baxter robot.

arm, to measure the spatial orientation of the upper arm, forearm, and palm. Taking the right arm as an example, the coordinate system is presented in Figure 3B: G is the global coordinate frame, U is the upper arm coordinate frame, F is the forearm coordinate frame, P is the palm coordinate frame, I_U is the coordinate frame of the IMU on the upper arm, I_F is the coordinate frame of the IMU on the forearm, and I_P is the coordinate frame of the IMU on the palm.

The initial position of the right arm joint angle is specified as $qr_0 = [q_1, q_2, q_3, q_4, q_5, q_6, q_7] = [0, 0, 0, 0, 0, 0, 0]$. In the initial joint angle configuration, the orientation of each frame of the right arm with respect to the global frame is represented as: $R^0_{GU} = R^0_{GF} = R^0_{GP} = [0, 0, 1; 0, 1, 0; -1, 0, 0]$. The measured values of the IMU on the upper arm, forearm, and palm are $R^0_{GI_U}$, $R^0_{GI_F}$, and $R^0_{GI_P}$, respectively, that is, the orientation of the IMU's frame with respect to the global frame. The orientation of the IMU relative to the arm can be determined as shown in Equation (1):

$$\begin{cases} R_{UI_U} = \left(R^0_{GU}\right)^T R^0_{GI_U} \\ R_{FI_F} = \left(R^0_{GF}\right)^T R^0_{GI_F} \\ R_{PI_P} = \left(R^0_{GP}\right)^T R^0_{GI_P} \end{cases} \tag{1}$$

where R_{UI_U} is the orientation of the IMU's frame on the upper arm relative to the upper arm's frame, R_{FI_F} is the orientation of the IMU's frame on the forearm relative to the forearm's frame, and R_{PI_P} is the orientation of the IMU's frame on the palm relative to the palm's frame.

When the arm moves to a new position, the orientation of the IMU concerning the arm remains unchanged, assuming that the new orientations of the IMU's frame relative to the global frame are $R^{new}_{GI_U}$, $R^{new}_{GI_F}$, and $R^{new}_{GI_P}$. The orientation of the upper arm, forearm, and palm can be calculated using Equation (2):

$$\begin{cases} R^{new}_{GU} = R^{new}_{GI_U} \left(R_{UI_U}\right)^T \\ R^{new}_{GF} = R^{new}_{GI_F} \left(R_{FI_F}\right)^T \\ R^{new}_{GP} = R^{new}_{GI_P} \left(R_{PI_P}\right)^T \end{cases} \tag{2}$$

where R^{new}_{GU} is the orientation of the upper arm's frame with respect to the global frame in new position, R^{new}_{GF} is the orientation of the forearm's frame with respect to the global frame in new position, and R^{new}_{GP} is the orientation of the palm's frame with respect to the global frame in new position. The relationship between the orientation of the upper arm's frame with respect to the global frame and the joint angles q_1, q_2, and q_3 is shown in Equation (3):

$$R^{new}_{GU} = R_X\left(-q_1\right) R_Y\left(q_2\right) R_Z\left(q_3\right) \tag{3}$$

By substituting Equation (2) into Equation (3), the joint angles q_1, q_2, and q_3 corresponding to the shoulder joint can be calculated as Equation (4):

$$\begin{cases} q_1 = -atan2\left(-R^{new}_{GU}\left(2,3\right), R^{new}_{GU}\left(3,3\right)\right) \\ q_3 = atan2\left(-R^{new}_{GU}\left(1,2\right), R^{new}_{GU}\left(1,1\right)\right) \\ q_2 = atan2\left(-R^{new}_{GU}\left(1,3\right), \frac{R^{new}_{GU}(1,1)}{cos(q_3)}\right) \end{cases} \tag{4}$$

The orientation of the forearm relative to the upper arm can be calculated using Equation (5):

$$R^{new}_{UF} = \left(R^{new}_{GU}\right)^T R^{new}_{GF} \tag{5}$$

The relationship between the orientation of the forearm's frame with respect to the global frame and the joint angles q_4 and q_5 is shown in Equation (6):

$$R^{new}_{UF} = R_Y\left(q_4\right) R_Z\left(q_5\right) \tag{6}$$

By substituting Equation (5) into Equation (6), the joint angles q_4 and q_5 corresponding to the elbow joint can be calculated as Equation (7):

$$\begin{cases} q_4 = atan2\left(-R^{new}_{UF}\left(1,3\right), R^{new}_{UF}\left(3,3\right)\right) \\ q_5 = atan2\left(-R^{new}_{UF}\left(2,1\right), R^{new}_{UF}\left(2,2\right)\right) \end{cases} \tag{7}$$

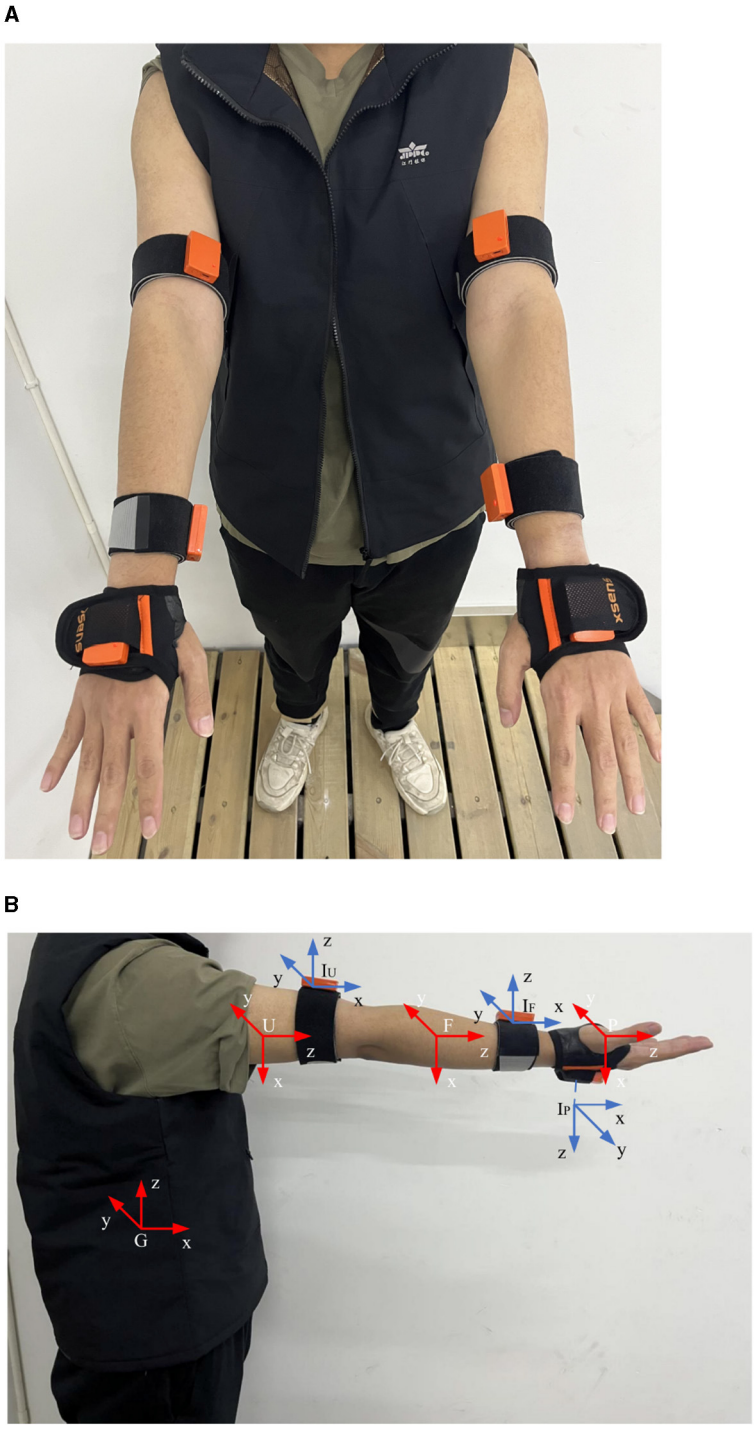

FIGURE 3
IMUs and coordinate frame schematic. **(A)** The position of the IMU on the presenter's arm. **(B)** Coordinate frame diagram.

The orientation of the palm relative to the forearm can be calculated using Equation (8):

$$R_{FP}^{new} = \left(R_{GF}^{new}\right)^{T} R_{GP}^{new} \tag{8}$$

The relationship between the orientation of the palm's frame with respect to the global frame and the joint angles q_6 and q_7 is shown in Equation (9):

$$R_{FP}^{new} = R_Y\left(q_6\right) R_Z\left(q_7\right) \tag{9}$$

By substituting Equation (8) into Equation (9), the joint angles q_6 and q_7 corresponding to the wrist joint can be calculated

as Equation (10):

$$\begin{cases} q_6 = atan2\left(-R_{FP}^{new}(1,3), R_{FP}^{new}(3,3)\right) \\ q_7 = atan2\left(-R_{FP}^{new}(2,1), R_{FP}^{new}(2,2)\right) \end{cases} \tag{10}$$

In this way, the right arm joint angle is calculated, and the left arm joint angle can also be calculated by the above method.

2.2 Joint angle constraint

A human demonstration can be divided into multiple trajectories for a complex multi-step task. For example, a bottle cap screwing task can be broken down into reaching, grabbing, aligning, and screwing steps. Suppose a multi-step task is artificially divided into N sub-tasks; in the sub-task n, the angle constraint of the k-th joint of the human arm can be expressed as Equation (11):

$$a_k^n \le q_k^n \le b_k^n \tag{11}$$

where $0 < n \le N$, $0 < k \le 14$, a_k^n is the lower limit for the k-th joint angle in sub-task n, b_k^n is the upper limit for the k-th joint angle in sub-task n.

When there are M groups of human demonstrations, the trajectories can be divided into $M * N$ sub-trajectories, where N is the number of sub-tasks. Then, for sub-task n, the angle constraint of the k-th human arm joint can be expressed as Equation (12):

$$C_k^n: \begin{cases} q_k^n \ge \min\left(a_{k,1}^n, a_{k,2}^n, \cdots, a_{k,m}^n\right) \\ q_k^n \le \max\left(b_{k,1}^n, b_{k,2}^n, \cdots, b_{k,m}^n\right) \end{cases} \tag{12}$$

where $0 < m \le M$, $a_{k,m}^n$ is the lower limit of the k-th joint angle in the sub-task n of the demonstration m, $b_{k,m}^n$ is the upper limit of the k-th joint angle in the sub-task n of the demonstration m.

2.3 Reinforcement learning method

The DADDPG method proposed in previous study can plan the coordinated trajectories of dual-arm robots for multi-objective tasks (Liang et al., 2023). In this study, the DADDPG algorithm is chosen as the algorithm of reinforcement learning, which uses two agents to plan the coordinated trajectory of the left arm and the right arm simultaneously. Each agent contains four networks: Actor $\mu_i(s_i|\theta_i^{\mu})$, Critic $Q_i(\mathbf{s}, a_i|\theta_i{}^{Q})$, Target Actor $\mu_i'(s_i'|\theta_i^{\mu'})$, and Target Critic $Q_i'(\mathbf{s}', a_i'|\theta_i^{Q'})$, where $i = 1, 2$.

For the agent i, the parameters of the Critic network are updated by minimizing MSBE loss Lc by the gradient descent method using Equation (13) (Liang et al., 2023):

$$L_c = (y_i - q_i)^2 = (Q_i(\mathbf{s}_j, a_{j,i}|\theta_i^Q) - r_{j,i} + \gamma(1 - done) Q_i'(\mathbf{s}_{j+1}, \mu_i'(s_{j+1,i}|\theta_i^{\mu'})|\theta_i^{Q'}))^2 \tag{13}$$

The parameters of the Actor network are updated by maximizing the cumulative expected return J of agent i by the gradient ascent method using Equation (14) (Liang et al., 2023):

$$\nabla_{\theta_i^{\mu}} J = \underset{\mathbf{s}\sim\mu_1,\mu_2}{E}\left[\nabla_{a_i} Q_i(\mathbf{s}, a_i|\theta^Q)|_{\mathbf{s}=\mathbf{s}_j, a_i=\mu_i(s_{j,i})} \nabla_{\theta_i^{\mu}} \mu_i(s_i|\theta_i^{\mu})|s_{j,i}\right] \tag{14}$$

The parameters of the target networks are updated by way of soft update using Equation (15) (Liang et al., 2023):

$$\begin{cases} \theta_i^{Q'} \leftarrow \tau\theta_i^{Q} + (1-\tau)\theta_i^{Q'} \\ \theta_i^{\mu'} \leftarrow \tau\theta_i^{\mu} + (1-\tau)\theta_i^{\mu'} \end{cases} \tag{15}$$

2.4 Guided reward design

Based on the reward function designed in the previous study (Liang et al., 2023), this study develops the segmented guided step reward term so that the human joint angle constraint can guide the learning process of the reinforcement learning method and narrow its exploration space. The reward of agent i can be calculated using the reward function. The segmented guided step reward term r_{guide} is shown in Equation (16):

$$r_{guide} = \sum_k r_{n,k} \tag{16}$$

where

$$n = \begin{cases} 1 & \text{if } goal_1 = False \\ 2 & \text{if } goal_1 = True, goal_2 = False \\ \cdots & \\ N & \text{if } goal_1, goal_2, \cdots, goal_{N-1} = True, goal_N = False \end{cases},$$

$goal_n$ is the target of sub-task n. $r_{n,k} = \begin{cases} c_0 & \text{if } q_k \text{ satisfies the constraint } C_k^n \\ -c_0 & \text{else} \end{cases}$, c_0 is a positive constant.

The step reward term of the reward function is shown as Equation (17):

$$r_{step} = -distance(pos\_gripper_i, pos\_finalgoal_i) + r_{guide} \tag{17}$$

The reward function is the sum of the three items shown in Equation (18) (Liang et al., 2023):

$$R = r_{step} + r_{goal} + r_{coordinate} \tag{18}$$

3 Experiment

3.1 Validation of joint angle mapping method

This experiment set up a human arm movement trajectory, and the arm posture data were measured using IMUs. The arm joint angle was calculated using the method in Section 2.1 and input into a simplified D-H model of the human arm. The human arm pose sampled at five positions was compared with the D-H visualization model pose, as shown in Figure 4.

In the D-H visualization model, the X-axis of the coordinate frame at the end of the right arm corresponds to the direction of the right hand, and the Z-axis corresponds to the direction of the fingers when the right hand is opened. The X-axis of the coordinate frame at the end of the left arm corresponds to the direction of the left hand, and the Z-axis corresponds to the direction of the fingers when the left hand is opened. As can be observed from the comparison in Figure 4, the orientation of the two end

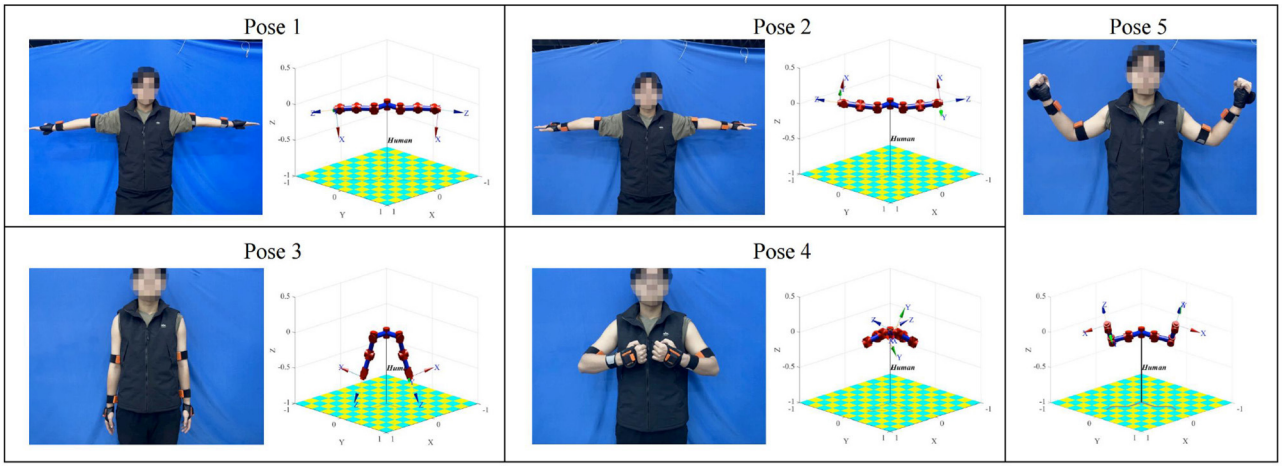

FIGURE 4
Comparison of human arm sampling pose and D-H visual model pose.

coordinate frames of the model is consistent with the orientation of the IMUs worn on the demonstrator's palms, and the posture of the demonstrator's arm and the model's arm is also very similar. Therefore, the joint angle calculation and mapping method in this study are effective.

3.2 Validation of joint angle constraint guidance method

In this section, the effectiveness of the proposed joint angle constraint guidance method was verified on a Baxter robot in a GYM simulation environment for multi-step tasks. The set multi-step task consists of three sub-steps: reach, grasp, and align. Ten groups of demonstration data of human execution of the reach-grasp-align task were collected, calculated, and processed to obtain joint angle constraints. Then, the constraint guidance is added to each time step of DADDPG algorithm learning trajectory. To verify the framework's effectiveness proposed in this study, the learning effects of single-segment constraint introduction, multi-segment constraint introduction, and unconstrained introduction are compared.

3.2.1 Experiment settings

The reach-grasp-align task scene of the Baxter robot in the GYM simulation environment is shown in Figure 5. The black block is the operating object of the left arm, the yellow block is the operating object of the right arm, the red sphere is the target position of the left arm, and the blue sphere is the target position of the right arm. The parameters of the DADDPG algorithm were set the same as in the study by Liang et al. (2023), except that the dimension of observation was changed.

3.2.1.1 Constraints

This experiment used two types of constraints: single-segment and multi-segment constraints. When the calculated joint angle data were not segmented, the entire motion process included reach, grasp, and align sub-steps, and a single segment constraint $C1$ was obtained. When the calculated joint angle data were segmented, it was divided into reach, grasp, and align sub-steps. Because the joint movement of the grasp substep was tiny, constraint $C2$ was obtained from the joint angle data of the reach substep and the grasp substep, and constraint $C3$ was obtained from the joint angle data of the align substep. $C2$ and $C3$ form a multi-segment constraint.

3.2.1.2 Observation

Joint angle constraint guidance must introduce the state of the robot's joint angle in the observation to guide the agent's learning. The observation was therefore set as: $\mathbf{s} = (s_l, s_r)$, s_l is the state of the robot's left arm and its target, including the position of the left gripper, the position of the left arm joint angle, the position of the left object, the relative position of the left object and the left gripper, the state of the two fingers of the left gripper, the orientation of the left object, the linear velocity of the left object, the angular velocity of the left object, the linear velocity of the left gripper, the speed of the two fingers of the left gripper, and the position of the left target. s_r is defined as the variable corresponding to the right arm.

When no constraints are introduced, the state of the robot's joint angle is not required. The observation was the same as in the study by Liang et al. (2023).

3.2.1.3 Reward

This experiment used the reward function designed in Section 2.4. Set the guiding reward constant to 0.01 for the reach and grasp stages and 0.05 for the align stage. Set the reward of subgoal reaching $rg_1 = 1$, the reward of subgoal grasping $rg_2 = 20$, the reward of coordinate $rc = 4000$.

3.2.2 Results

The comparison of training curves of the DADDPG algorithm with single-segment constraint introduction, multi-segment constraint introduction, and no constraint introduction in

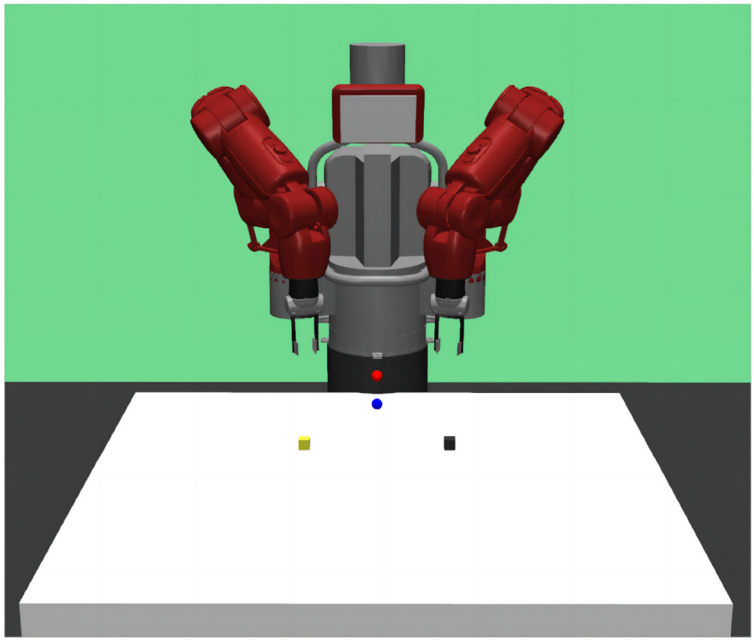

FIGURE 5
GYM simulation environment for Baxter robot's reach-grasp-align task. Baxter's arms reach the position of their respective object box, then grasp their respective object box, and finally align the two object boxes.

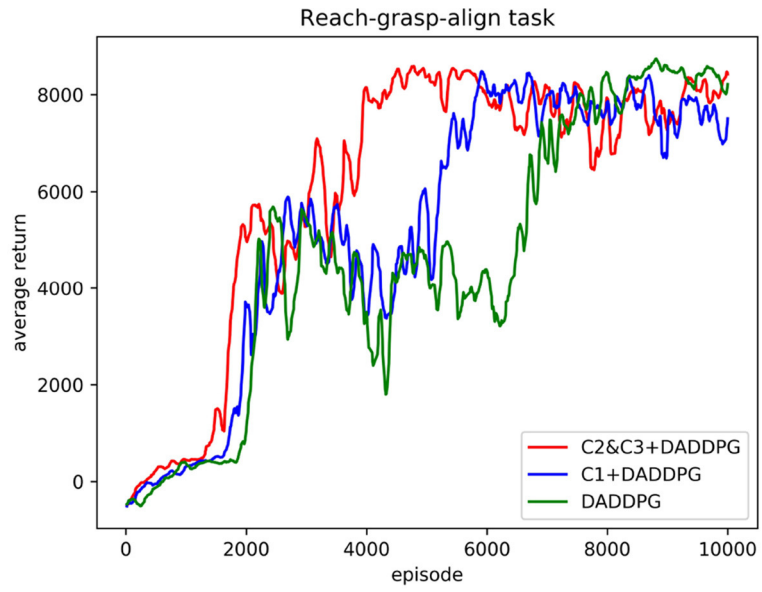

FIGURE 6
Comparison of the average cumulative return curves of the DADDPG algorithm with single-segment constraint introduction, multi-segment constraint introduction, and no constraint introduction trained in the reach-grasp-align task of the dual-arm robot.

the reach-grasp-align task of the dual-arm robot is shown in Figures 6, 7.

Figure 6 shows the curve comparison of the average cumulative return as the number of training increases. The DADDPG algorithm with single-segment constraint introduction, multi-segment constraint introduction, and no constraint introduction was trained 10,000 episodes. As can be observed from the figure, the average cumulative return curve of an agent guided by multi-segment constraints *C*2&*C*3 converges at approximately 4,000 episodes, that of an agent guided by single-segment constraints *C*1 converges at approximately 5,600 episodes, and that of an agent guided by no constraints converges

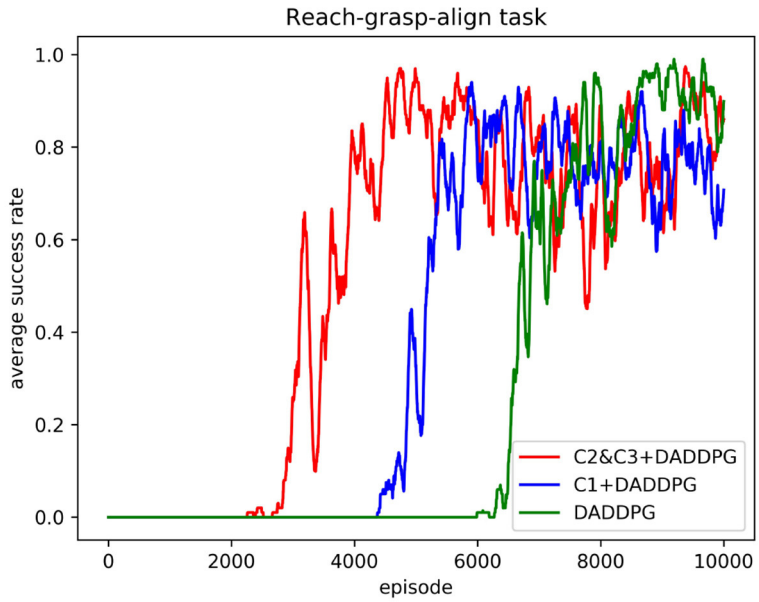

FIGURE 7
Comparison of average success rate curves of the DADDPG algorithm with single-segment constraint introduction, multi-segment constraint introduction, and no constraint introduction in the reach-grasp-align task of the dual-arm robot.

at approximately 7,000 episodes. Regarding the number of training episodes for the average cumulative return curve convergence, the number of training episodes required for converging multi-segment constrained guided agents is 71% for single-segment constrained guided agents and 57% for unconstrained guided agents. The number of necessary training episodes for single-segment constrained guided agent convergence is 80% of that needed for unconstrained guided agents. The results show that adding single-segment constraints to DADDPG can significantly improve the speed of training convergence. Improving the rate of training convergence by introducing constraints in segments is more prominent. This verifies the validity of the motion planning framework for two-arm robots based on the DADDPG method, which is guided by human joint constraints.

Figure 7 shows the curve comparison of the average success rate as the number of training increases. As can be observed from the figure, the average success rate of the agent guided by multi-segment constraint *C*2&*C*3 is 0.85 or above approximately 4,480 episodes, and the average success rate of the agent guided by single-segment constraint *C*1 is 0.85 or above approximately 5,810 episodes. The average success rate of agents without constraint guidance is 0.85 and above approximately 7,685 episodes. Regarding the number of training episodes required to achieve a success rate of 0.85 and above, the multi-stage constraint is 77% of the single-stage constraint guidance and 58% of the unconstrained guidance. In addition, the maximum average success rate for multi-segment constrained booting is 0.98, the maximum average success rate for single-segment constrained booting is 0.94, and the maximum average success rate for unconstrained booting is 0.98. The results show that using joint Angle constraint to guide the learning of the agent can improve the learning speed without sacrificing the success rate. The results demonstrate the superiority of the proposed framework in the performance of multi-step tasks.

Figure 8 compares the actual training time of DADDPG algorithm with single-segment constraint introduction, multi-segment constraint introduction, and no constraint introduction in the reach-grasp-align task of the dual-arm robot. The DADDPG algorithm with single-segment constraint introduction, multi-segment constraint introduction, and unconstrained constraint introduction was trained in 10,000 episodes on the same device. As can be observed from the figure, the average cumulative return curve of an agent guided by multi-segment constraints *C*2&*C*3 converges at approximately 48590s, that of an agent guided by single-segment constraints *C*1 converges at approximately 75830s, and that of an agent guided by no constraints converges at approximately 96393s. Regarding actual training time for average cumulative return curve convergence, the training time required for multi-stage constrained guided agent convergence is 64% of that for single-stage constrained guided agent and 50% for unconstrained guided agent. The results show that although the joint angle constraint guidance needs to increase the observation dimension and improve the computational complexity to a certain extent, the actual training time is still significantly reduced. The results further demonstrate the effectiveness of the proposed framework in reducing training time.

4 Discussion

Improving the learning efficiency of dual-arm robots' motion planning is always a primary concern, as it can save the investment in time and hardware. In this article, we introduced human

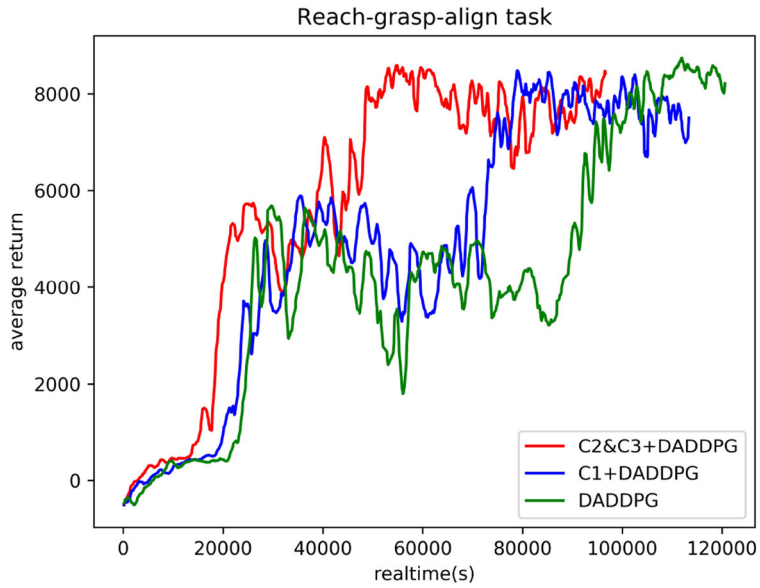

FIGURE 8
Comparison of actual training time curves of the DADDPG algorithm with single-segment constraint introduction, multi-segment constraint introduction, and no constraint introduction in the reach-grasp-align task of the dual-arm robot.

joint angle constraints into the DADDPG method for improving the motion planning framework of dual-arm robots. We tested the improved framework on a Baxter dual-arm robot in the Gym simulation environment. The performance of the proposed motion planning framework was evaluated in multi-step tasks (reaching, grasping, and aligning). The results show that the introduced human angle constraints effectively guide robots to learn tasks faster.

Human movement patterns are energy consumption optimal solutions learned through life experience. Inspired by human movement patterns during reaching, grasping, and aligning objects in random places, we extracted human motion features from joint angle curves for faster learning and a higher task completion rate. These features are transformed into constraints in each step during learning. In Section 3.2, three operations were planned using the same constraints. As shown in Figures 6, 8, the real learning time and the number of iterations were reduced. This phenomenon demonstrates that constraining the angle range of the robot joint can narrow the exploration space of the end trajectory, thus improving the learning efficiency. To further enhance learning efficiency, smaller constraints were defined for reaching/grasping and aligning, respectively. The real learning time and the number of iterations were shorter. The effectiveness of the introduced human joint angle constraints is verified.

This study defined the constraints for the joint angle of dual-arm robots. Thus, a smaller search space of the end is obtained. Some other motion parameters, such as joint angular velocity or acceleration, can also be constrained to improve the learning efficiency. The constraints can be defined more strictly according to the human motion features with respect to these motion parameters. Therefore, the proposed constraint-based dual-arm robot motion planning framework has a scalability potential.

In our future studies, the performance of the proposed planning framework will be verified on the objects unseen in the Gym simulation learning. A more complicated task pool will also be developed to show the potential of this work. The additional tasks are mainly focused on three application scenarios: a) dynamic assembly of parts in the factory, b) valve screwing in the space environment, and c) multi-objects sequentially screwing in housekeeping and healthcare.

5 Conclusion

This study proposes a motion planning framework based on reinforcement learning guided by human joint angle constraints, and the DADDPG algorithm is selected as part of reinforcement learning. First, the human joint angle is calculated from the demonstration data collected by IMU and mapped to the robot model. The joint angle constraint is extracted piecewise from multiple groups of human demonstrations. Then, the segmented step guidance reward is designed, and the joint angle constraint is introduced into the reinforcement learning algorithm to guide the autonomous learning of the multi-step coordination trajectory of both arms. Finally, for the reach-grasp align task of the two-arm robot, the effectiveness of the proposed framework was verified in terms of training convergence speed, success rate, and training duration under the GYM simulation environment of the Baxter robot. The method will be extended to more complex multi-step tasks and applied to bottle cap screwing scenarios for home services in future studies.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding authors.

Author contributions

KL: Writing – original draft, Writing – review & editing. FZ: Funding acquisition, Resources, Writing – review & editing. WG: Visualization, Writing – review & editing. SL: Writing – review & editing. PW: Formal analysis, Writing – review & editing. LS: Supervision, Writing – review & editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported in part by the National Key R&D Program of China (2020YFB13134), National Natural Science Foundation of China (U2013602, 52075115, 51521003, 61911530250), Self-Planned Task (SKLRS202001B, SKLRS202110B) of State Key Laboratory of Robotics and System (HIT), Shenzhen Science and Technology Research and Development Foundation (JCYJ20190813171009236), and Basic Scientific Research of Technology (JCKY2020603C009).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

Bing, Z., Brucker, M., Morin, F. O., Li, R., Su, X., Huang, K., et al. (2022a). Complex robotic manipulation via graph-based hindsight goal generation. *IEEE Trans. Neural Netw. Learn. Syst.* 33, 7863–7876. doi: 10.1109/TNNLS.2021.3088947

Bing, Z., Cheng, L., Huang, K., and Knoll, A. (2022b). Simulation to real: learning energy-efficient slithering gaits for a snake-like robot. *IEEE Robot. Automat. Maga.* 29, 92–103. doi: 10.1109/MRA.2022.3204237

Bing, Z., Knak, L., Cheng, L., Morin, F. O., Huang, K., and Knoll, A. (2023a). Meta-reinforcement learning in nonstationary and nonparametric environments. *IEEE Trans. Neural Netw. Learn. Syst.* 2023, 1–15. doi: 10.1109/TNNLS.2023.3270298

Bing, Z., Lerch, D., Huang, K., and Knoll, A. (2023b). Meta-reinforcement learning in non-stationary and dynamic environments. *IEEE Trans. Pattern Anal. Mach. Intell.* 45, 3476–3491. doi: 10.1109/TPAMI.2022.3185549

Bing, Z., Rohregger, A., Walter, F., Huang, Y., Lucas, P., Morin, F. O., et al. (2023c). Lateral flexion of a compliant spine improves motor performance in a bioinspired mouse robot. *Sci. Robot.* 8, eadg7165. doi: 10.1126/scirobotics.adg7165

Bing, Z., Zhou, H., Li, R., Su, X., Morin, F. O., Huang, K., et al. (2023d). Solving robotic manipulation with sparse reward reinforcement learning via graph-based diversity and proximity. *IEEE Trans. Indust. Electr.* 70, 2759–2769. doi: 10.1109/TIE.2022.3172754

Bougie, N., Cheng, L. K., and Ichise, R. (2018). Combining deep reinforcement learning with prior knowledge and reasoning. *ACM SIGAPP Appl. Comput. Rev.* 18, 33–45. doi: 10.1145/3243064.3243067

Chu, Z., Wang, F., Lei, T., and Luo, C. (2022). Path planning based on deep reinforcement learning for autonomous underwater vehicles under ocean current disturbance. *IEEE Trans. Intell. Vehicl.* 8, 108–120. doi: 10.1109/TIV.2022.3153352

Denavit, J. and Hartenberg, R. S. (1955). A kinematic notation for lower-pair mechanisms based on matrices. *J. Appl. Mech.* 22, 215–221. doi: 10.1115/1.4011045

Dong, Z., Li, Z., Yan, Y., Calinon, S., and Chen, F. (2022). Passive bimanual skills learning from demonstration with motion graph attention networks. *IEEE Robot. Automat. Lett.* 7, 4917–4923. doi: 10.1109/LRA.2022.3152974

Fang, C., Rocchi, A., Hoffman, E. M., Tsagarakis, N. G., and Caldwell, D. G. (2015). "Efficient self-collision avoidance based on focus of interest for humanoid robots," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)* (Seoul: IEEE), 1060–1066.

Giftthaler, M., Farshidian, F., Sandy, T., Stadelmann, L., and Buchli, J. (2017). "Efficient kinematic planning for mobile manipulators with non-holonomic constraints using optimal control," in *2017 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE), 3411–3417.

Gulletta, G., Erlhagen, W., and Bicho, E. (2020). Human-like arm motion generation: a review. *Robotics* 9, 102. doi: 10.3390/robotics9040102

Jiang, D., Cai, Z., Peng, H., and Wu, Z. (2021). Coordinated control based on reinforcement learning for dual-arm continuum manipulators in space capture missions. *J. Aerosp. Eng.* 34, 04021087. doi: 10.1061/(ASCE)AS.1943-5525.0001335

Kim, S., Kim, C., and Park, J. H. (2006). "Human-like arm motion generation for humanoid robots using motion capture database," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Beijing: IEEE), 3486–3491.

Liang, K., Zha, F., Sheng, W., Guo, W., Wang, P., and Sun, L. (2023). "Research on target trajectory planning method of humanoid manipulators based on reinforcement learning," in *Intelligent Robotics and Applications*, eds. H. Yang, H. Liu, J. Zou, Z. Yin, L. Liu, G. Yang, et al. (Singapore. Springer Nature Singapore), 452–463.

Maeda, G., Koç, O., and Morimoto, J. (2020). Phase portraits as movement primitives for fast humanoid robot control. *Neural Netw.* 129, 109–122. doi: 10.1016/j.neunet.2020.04.007

Mronga, D. and Kirchner, F. (2021). Learning context-adaptive task constraints for robotic manipulation. *Rob. Auton. Syst.* 141, 103779. doi: 10.1016/j.robot.2021.103779

Ren, H. and Ben-Tzvi, P. (2020). Advising reinforcement learning toward scaling agents in continuous control environments with sparse rewards. *Eng. Appl. Artif. Intell.* 90, 103515. doi: 10.1016/j.engappai.2020.103515

Shin, S. Y. and Kim, C. (2014). Human-like motion generation and control for humanoid's dual arm object manipulation. *IEEE Trans. Ind. Electron.* 62, 2265–2276. doi: 10.1109/TIE.2014.2353017

Suárez, R., Rosell, J., and Garcia, N. (2015). "Using synergies in dual-arm manipulation tasks," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, (Seattle, WA: IEEE), 5655–5661.

Tang, W., Cheng, C., Ai, H., and Chen, L. (2022). Dual-arm robot trajectory planning based on deep reinforcement learning under complex environment. *Micromachines (Basel)* 13, 564. doi: 10.3390/mi13040564

Taylor, M. E., Suay, H. B., and Chernova, S. (2011). "Integrating reinforcement learning with human demonstrations of varying ability," in *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, 617–624. doi: 10.5555/2031678.2031705

Tian, Y., Cao, X., Huang, K., Fei, C., Zheng, Z., and Ji, X. (2021). Learning to drive like human beings: a method based on deep reinforcement learning. *IEEE Trans. Intell. Transp. Syst.*, 23, 6357–6367. doi: 10.1109/TITS.2021.3055899

Vahrenkamp, N., Asfour, T., and Dillmann, R. (2012). Simultaneous grasp and motion planning: humanoid robot armar-iii. *IEEE Robot. Autom. Mag.* 19, 43–57. doi: 10.1109/MRA.2012.2192171

Wang, Z., Gan, Y., and Dai, X. (2022). Assembly-oriented task sequence planning for a dual-arm robot. *IEEE Robot. and Automat. Lett.* 7, 8455–8462. doi: 10.1109/LRA.2022.3183786

Xiang, G. and Su, J. (2019). Task-oriented deep reinforcement learning for robotic skill acquisition and control. *IEEE Trans. Cybern.* 51, 1056–1069. doi: 10.1109/TCYB.2019.2949596