Hindawi
Complexity
Volume 2022, Article ID 5648826, 10 pages
https://doi.org/10.1155/2022/5648826

Research Article

A Robot Human-Like Learning Framework Applied to Unknown Environment Interaction

Xianfa Xue,[1] Lei Zuo ,[2] and Ning Wang [3]

[1] *Key Laboratory of Autonomous Systems and Networked Control, School of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China*
[2] *School of Electronic and Control Engineering, Chang'an University, Xi'an 710064, China*
[3] *Bristol Robotics Laboratory, University of the West of England, Frenchay, Coldharbour Ln, Bristol BS34 8QZ, UK*

Correspondence should be addressed to Ning Wang; nwang0902@gmail.com

Received 13 July 2021; Accepted 4 December 2021; Published 1 March 2022

Academic Editor: Atila Bueno

Learning from demonstration (LfD) is one of the promising approaches for fast robot programming. Most learning systems learn both movements and stiffness profiles from human demonstrations. However, they rarely consider the unknown environment interaction. In this paper, a robot human-like learning framework is proposed, where it can learn human skills through demonstration and complete the interaction task with an unknown environment. Firstly, the desired trajectory was generated by dynamic movement primitive (DMP) based on human demonstration. Then, an adaptive optimal admittance control scheme was employed to interact with environments with the reference adaptation method. Finally, the experimental study was conducted, and the effectiveness of the framework proposed in this paper was verified via a group of curved surface wiping experiments on a balloon with unknown model parameters.

1. Introduction

Robot learning from demonstration (LfD) has recently drawn much attention because of its high efficiency in robot programming [1]. Thus, robots can quickly program the robots to perform operating variable skills and replace human tutors from such tasks in a complex industrial environment [2]. Compared to conventional programming methods using a teaching pendant, LfD is an easier and more intuitive way for people who are unfamiliar with programming. Besides, human characteristics involved in the demonstrations are available for robots to further improve the flexibility and compliance of motions [3, 4].

After the demonstration, how to use the information of the human tutor is very important. Dynamic movement primitive (DMP) is a kind of common method in human-robot skill transfer tasks [5]. DMP has many advantages; for example, the DMP model is so simple that we only need to adjust a few parameters to achieve trajectory modeling. Besides, we can use regression algorithm to quickly learn model parameters in the online trajectory planning of robots [6]. In addition, the DMP model is also easy to generalize; we can quickly generalize a trajectory with the same style as the original trajectory by simply adjusting the starting and ending coordinates of the trajectory [7, 8]. Because of the above advantages, the DMP has been widely used in human-robot skill transfer tasks [9].

Appropriate control strategies help robots reproduce human skills more accurately and stably. In some specific tasks such as surface cleaning, cargo handling, and environment identification, robots are required to track a task trajectory and achieve compliance in the interaction with environments [10]. In the previous literature on interaction control, two main methods have been studied widely: impedance control [11] and hybrid position/force control [12]. The admittance control, which is regarded as the position-based impedance control, can achieve good interaction performance by trajectory adaptation [13–15]. According to admittance models, the external forces received by the robot will be transformed to the position of the end-effector, and then, the desired interaction performance can be ensured by trajectory adaptation and tracking [16]. The control strategies mainly include proportion integration differentiation (PID) control, adaptive control, adaptive control using neural networks, and fuzzy control [17–20]. When robots perform different tasks in an unknown, complex, and dynamic environment, it is usually difficult to obtain accurate task models and environmental information, and various errors may have a serious impact on the final control results [21]. In recent years, control methods based on neural network learning have shown better adaptability to the system and environmental uncertainty, but this method requires a large amount of system data samples, and it is difficult to integrate various constraints in unknown environments in real-time [22, 23].

In this paper, firstly, the desired trajectory is generated by human demonstration, and then, an adaptive admittance control scheme is applied to interact with environments using the reference adaptation method. The contributions can be summarized as follows:

(1) An adaptive optimal admittance controller is developed to take into account the unknown interaction environment dynamics. Combining with the generalization ability of the DMP model and the compliance control ability of the adaptive optimal admittance model, the interaction performance between the robot and the unknown environment is improved.

(2) A complete human-like learning framework is developed. In the beginning, the desired trajectory can be obtained quickly and accurately by human teaching and generalization. And then, the online adaptive controller recalculates and updates the originally desired trajectory to obtain a new reference trajectory. The framework can update the new reference trajectory combined with different interaction environments, which greatly enhances the interaction accuracy

The rest of the article is organized as follows: In Section 2, the methods of desired trajectory generation and adaptive optimal admittance controller used in this paper are introduced. In Section 3, the experimental study is presented and then the effectiveness of the framework proposed in this paper is verified via balloon surface wiping experiments. Finally, Section 4 summarizes the whole paper.

2. Preliminaries and Methods

2.1. Overview of the Framework. The scheme of the proposed framework is shown in Figure 1. In the proposed learning framework, the human tutor presents a demonstration at first. The trajectory learned from the DMP model is regarded as the desired trajectory. Then, the desired trajectory and the interaction force measured by the force sensor are input into an adaptive admittance controller to obtain the modified reference trajectory. Here, x and $\dot{x}$ represent the current position and velocity, respectively. x_d and x_d represent the desired position and velocity, respectively. x_r represents the reference position. q, $\dot{q}$, and τ represent the current angle, angular velocity, and torque, respectively. q_r and τ_r represent the reference angle and reference torque, respectively. f_{int} represents the interaction force. Finally, new manipulation motions are implemented by the robot joint controller, and the collected new data are taken as a new demonstration for the repetitive training.

2.2. Dynamic Movement Primitives (DMPs). In this paper, motion DMP can be obtained by using the DMP model to fit the motion trajectory. The principles of motion DMP used in this paper are stated as follows [24, 25].

The essence of DMP is a second-order nonlinear dynamical system including spring and damper. A single-degree-of-freedom motion can be expressed by the following equations:

$$\tau\dot{\beta}_2 = a\left(g - \beta_1\right) - b\beta_2 + f\left(s;\omega\right), \tag{1}$$

$$\tau\dot{\beta}_1 = \beta_2, \tag{2}$$

$$\tau\dot{s} = -k_1 s, \tag{3}$$

where we ignore the time variable for the sake of simplicity. For example, $\beta_1(t)$ is represented by β_1; a and b represent the damping coefficient and spring constant of the system, respectively. And a is usually set as $a = b^2/4$; g is the target value of the motion trajectory; and τ represents the time scaling constant. β_1 and β_2 represent the position and velocity of motion trajectories, respectively. And the relationship between these two variables is shown in equation (2). ω means the weight of the Gaussian model. s is the phase variable of the system, which is calculated by the regular system of equation (3). And k_1 is a positive constant. The nonlinear function $f(s;\omega)$ is defined as

$$\begin{gathered} f\left(s;\omega\right) = \frac{\sum_{i=1}^{N}\varphi_i\omega_i}{\sum_{i=1}^{N}\varphi_i}\left(g - \beta_0\right)s, \\ \varphi_i = \exp\left(-d_i\left(s - c_i\right)^2\right), \end{gathered} \tag{4}$$

where c_i, di, and ω_i are the centre, width, and weight of the i-th kernel function, respectively. β_0 is the initial value of the motion trajectory. N is the total amount of Gaussian models.

In general, the initial value of s is set to 1, which gradually decays to zero. Since the value of s tends to zero, the nonlinear function $f\left(s;\omega\right)$ is bounded, and the model becomes a stable second-order spring-damped system.

In general, supervised learning algorithms such as the local weighted regression (LWR) algorithm are used to determine model parameters ω [26]. Given the teaching trajectory $\beta(t)$, where $t = [1, 2, \ldots, T]$, and $g = \beta(T)$, the force function can be determined according to the following equation

$$f_{\text{target}} = \tau\dot{\beta}_2 - K\left(g - \beta_1\right) - D\beta_2, \tag{5}$$

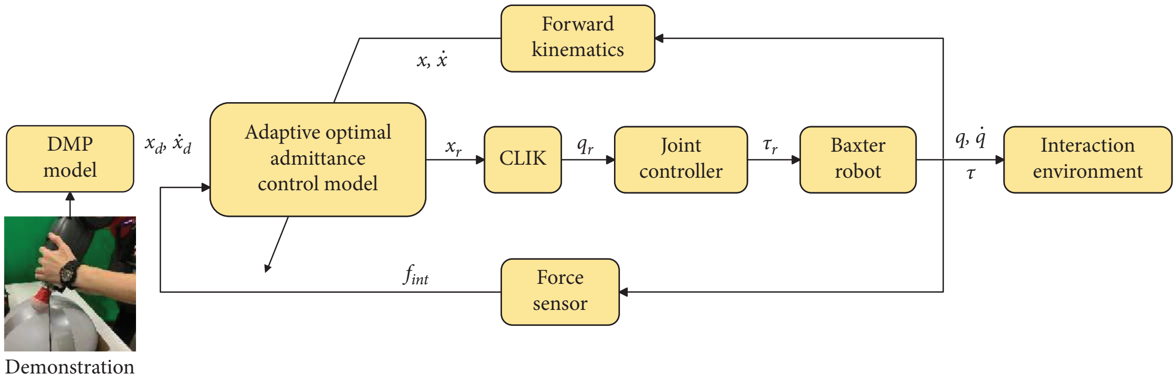

FIGURE 1: Overview of the proposed framework.

where K and D represent the stiffness and damping of the system, respectively. ω can be determined by the following equation:

$$\text{argmin}_{\omega} = \sum \left(f_{\text{target}} - f(s; \omega) \right)^2. \tag{6}$$

2.3. Adaptive Optimal Admittance Control. In this section, an adaptive task-specific admittance controller is developed. This adapts the parameters of the prescribed robot admittance model so that the robot system assists the human to achieve task-specific objectives. The task information is modeled by DMP so that the controller can adapt to the human tutor characteristics. The designed adaptive admittance controller will be used in the reproduction phase.

As shown in Figure 1, the process of adaptive admittance control in this article is as follows: the robot obtains the desired trajectory x_d, $\dot{x}_d$, $\ddot{x}_d$ through LfD and DMP generalization; then, the force sensor collects the interaction force between the robot end-effector and the environment in real-time. They are used as the input of the adaptive admittance model, and the expected trajectory x_d is improved according to the admittance model. At this time, a new reference trajectory x_r is obtained, and x_r is transmitted to the controller as the control signal to ensure the fast and accurate tracking of the actual trajectory to the reference trajectory. Among them, the core of adaptive admittance control is that the model parameters are not fixed but can be optimized online with the help of an adaptive algorithm according to the real-time position and interaction force information, in order to minimize the quadratic cost function [13].

The prescribed admittance model is defined as follows:

$$M_E(\ddot{x} - \ddot{x}_d) + C_E(\dot{x} - \dot{x}_d) + K_E x = -f_{\text{int}}, \tag{7}$$

where x, $\dot{x}$, and $\ddot{x}$ represent the current position, velocity, and acceleration, respectively. $\dot{x}_d$ and $\ddot{x}_d$ represent the desired velocity and acceleration, respectively. M_E, C_E, and K_E represent the unknown mass, damping, and stiffness matrices in the model, respectively. However, the mass matrix M_E is usually high nonlinear. In this study, the mass-damping-stiffness model is simplified as the damping-stiffness model, which is used to interact with a balloon as a kind of flexible object. The simplified model is as follows:

$$C_E(\dot{x} - \dot{x}_d) + K_E x = -f_{\text{int}}. \tag{8}$$

Suppose the following continuous-time linear system:

$$\dot{\xi} = A\xi + Bu(t), \tag{9}$$

where $\xi = [x^T, {x_d}^T]^T$, $A = \text{diag}\{-{C_E}^{-1}K_E, I_n\}$, $B = [-{C_E}^{-1}, 0]^T$, and $u(t) = f_{int}$ is system input variable, which is related to the dynamic model of the interactive environment (8).

The optimal control input of the system is designed as $u = -K\,\xi$, and the control objective is to minimize the cost function by designing the control system. The cost function is defined as follows:

$$\begin{aligned} V(\text{t}) &= \int_0^{\infty} \left((x - x_d)^T Q (x - x_d) + f_{\text{int}}^T R f_{\text{int}} \right) \text{dt}, \\ &= \int_0^{\infty} \left(\xi^T Q' \xi + u^T R u \right) \text{dt}, \end{aligned} \tag{10}$$

where Q is a constant matrix, $Q' = [Q, -Q]^T$ [1, −1], which represents the weight matrix of tracking error. R represents the weight matrix of the external force. In this paper, the design of the cost function takes into account the both robot system state and the external environment to evaluate the interactive control effect.

In the case that A and B are unknown constant matrices, an algorithm to obtain the optimal control signal by online learning is proposed. First, some variables are defined as follows:

$$\begin{aligned} \overline{\xi} &= \left[\xi_1^2, \xi_1\xi_2, \ldots, \xi_1\xi_m, \xi_2^2, \xi_2\xi_3, \ldots., \xi_m^2 \right]^T, \\ \Delta_{\xi\xi} &= \left[\overline{\xi}(t_1) - \overline{\xi}(t_0), \overline{\xi}(t_2) - \overline{\xi}(t_1), \ldots \overline{\xi}(t_h) - \overline{\xi}(t_{h-1}) \right]^T, \\ I_{\xi\xi} &= \left[\int_{t_0}^{t_1} \xi \otimes \xi, \int_{t_1}^{t_2} \xi \otimes \xi, \ldots, \int_{t_{h-1}}^{t_h} \xi \otimes \xi \right]^T, \\ I_{\xi u} &= \left[\int_{t_0}^{t_1} \xi \otimes u, \int_{t_1}^{t_2} \xi \otimes u, \ldots, \int_{t_{h-1}}^{t_h} \xi \otimes u \right]^T, \end{aligned} \tag{11}$$

where ξ, $\Delta_{\xi\xi}$, $I_{\xi\xi}$, and $I_{\xi u}$ are the intermediate variable matrices used to calculate the state feedback gain. h is the degree of integration and ⊗ represents the Kronecker product. We define that P_K which transforms matrix into vector form, and P_K is a symmetric matrix:

$$P_K = \left[P_{11}, 2P_{12} \ldots 2P_{1m} \ldots P_{mm}\right]^T. \tag{12}$$

The principle of the adaptive optimal admittance scheme is summarized in Algorithm 1 [27, 28].

$$\begin{aligned} \Theta_k &= \left[\delta_{\xi\xi}, -2I_{\xi\xi}\left(I_m \otimes K_k^T R\right) - 2I_{\xi u}\left(I_m \otimes R\right)\right], \\ \Xi_k &= -I_{\xi\xi}\mathrm{vec}\left(Q_k\right), \\ Q_k &= Q + K_k^T R K_k. \end{aligned} \tag{13}$$

where I_m is the m-dimension identity matrix and $vec(*)$ is the function that transforms the matrix into a vector. Through equation (17), we can obtain the optimal feedback control gain K_K*. Substituting K_K* into u = -K ξ, the optimal feedback control signal u can be obtained.

2.4. Inverse Kinematics Using CLIK. The closed-loop inverse kinematics (CLIK) algorithm is employed to resolve the Cartesian reference trajectory x_r into q_r in joint space [29–31]. The solution error is $e = k(q_r) - x_r$, where $k(*)$ denotes the forward kinematics and e is given by

$$\dot{e} = -K_e e. \tag{14}$$

where K_e is a positive user-defined matrix that decides the convergent rate of e. Expanding the above equations and combining with $x = J_i\ q$ and $J_i = \partial k(q)/q$, Ji is the Jacobian matrix of the robot. The following equation is obtained:

$$\dot{q}_r = J_i^+\left(\dot{x}_r - K_e\left(k\left(q_r\right) - x_r\right)\right). \tag{15}$$

Furthermore, we obtain the CLIK method:

$$q_r = \int_0^t \left(J_i^+\dot{x}_r - J_i^+ K_e\left(k\left(q_r\right) - x_r\right)\right)\mathrm{d}t, \tag{16}$$

where $J_i^+ = J_i^T\left(J_i J_i^T + \delta I_n\right)^{-1}$.

3. Experiment and Analysis

In this section, the performance of the proposed learning framework was validated by conducting experiments on a 7-DOF Baxter robot, as shown in Figure 2. The manipulator was equipped with the ATI Mini45 Force/Torque force sensor. The end effector was wrapped in a towel used to wipe the drawn curve on the surface of the balloon. The force sensor and the system controller are communicated by the UDP protocol whose sampling rate and control rate are set as 100 and 50 Hz, respectively. To prevent the displacement of the balloon from affecting the experimental results, the balloon to be wiped was fixed in the fixing box. The box is a paper carton with a size just big enough to hold the balloon. The paper carton is of size 43 cm × 32 cm × 18 cm and is fixed on the test bench with adhesive tape.

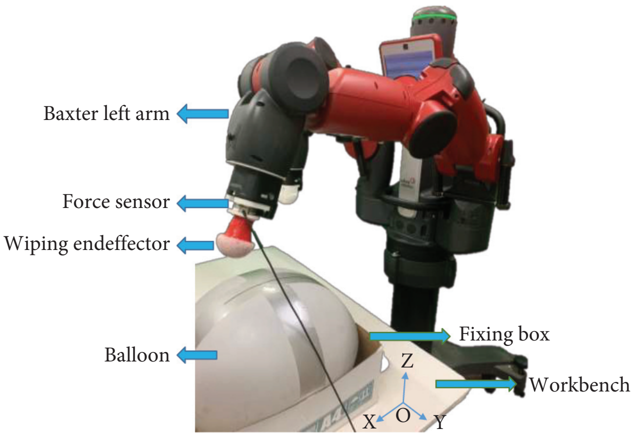

Figure 2: Experimental platform based on Baxter robot.

3.1. Demonstration Stage. In the teaching stage, first, a curve was drawn on the surface of the balloon with a whiteboard pen, and then, the human tutor dragged the left arm of the Baxter robot to complete the teaching task—wiping. In the meantime, the teaching trajectory information was recorded and input into the DMP model through the program. Then, the system learnt and generalized to obtain the desired trajectory x_d. At the same time, the force sensor recorded the interactive forces in the X, Y, and Z directions for subsequent analysis.

3.2. Reproduction of the Wiping Task. In the beginning, a new curve was drawn on the surface of the balloon. The robot end-effector was controlled to move to the starting point of the desired trajectory at [0.992, 0.280, 0.227] m. At this time, the end of the robot arm had interacted with the environment and changed from free space motion to constrained space motion. Since balloon was used as an interactive environment, its parameters are unknown. So an adaptive optimal admittance control was proposed to solve this problem. According to the set cost function, online adaptive learning of the interactive environment model parameters could help to achieve our desired control effects and complete the wiping task of the new curve.

The first wiping experiment process was that the trajectory obtained by a demonstration was directly used as the reference trajectory, and the admittance model parameters were specified as $C_E = [-0.5,\ 0.01,\ -0.8]$ and $K_E = [7,\ 2,\ 10]$. In the second experiment, the trajectory obtained by teaching was input into the DMP model for learning and generalization. The generated trajectory was used as the reference trajectory and then applied to the admittance model of the first experiment. In the third experiment, the expected trajectory obtained by DMP learning generalization of the teaching trajectory was used as the input of the adaptive optimal admittance controller, and finally, a new reference trajectory was obtained and then input to the Baxter joint controller. The initial value of state feedback gain in the X, Y, and Z directions was set to [−10, 1], and the weight matrix of cost function was Q = [200, −200], R = 5. Finally, the optimal state feedback gains

```
Input: the set of initial feedback gain $K_0$ and state variable $\xi$;
Output: optimal feedback gain $K_K{}^*$;
phase 1: set $f_{int0} = K_0\xi$ as the initial input while the manipulator is contacting with the environment;
Repeat: compute $\Delta_{\xi\xi}$, $I_{\xi\xi}$, and $I_{\xi u}$;
Until: rank$[I_{\xi\xi}, I_{\xi u}] = m(m+1)/2 + mr$;
Repeat:
phase 2: solve $P_K$ and $K_{K+1}$ according to:
    $\begin{bmatrix} \widehat{P}_K \\ \text{vec}(K_{K+1}) \end{bmatrix} = (\Theta_k^T\Theta_k)^{-1}\Theta_k^T\Xi_k.$
Until: $||P_K - P_{K-1}|| < \varepsilon$;
Return: $K_K{}^*$;
```

Algorithm 1: Framework of the adaptive optimal admittance scheme.

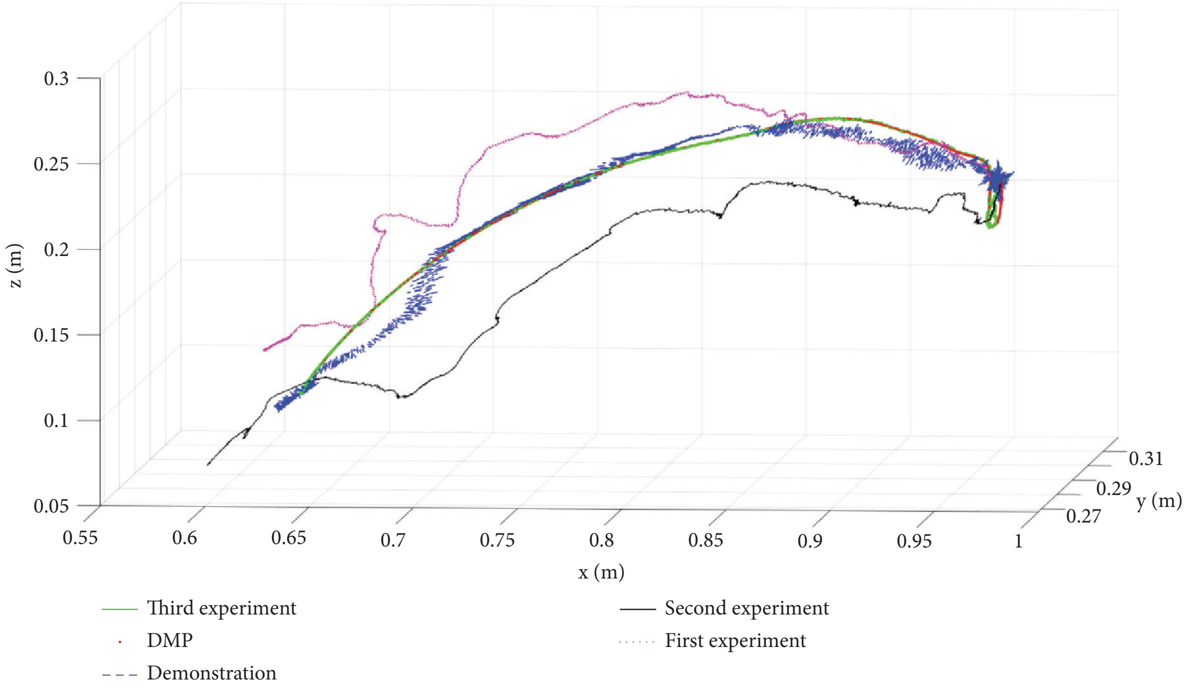

Figure 3: Three-dimensional curves of teaching trajectory, DMP generalization trajectory, and three times experimental trajectory.

in the X, Y, and Z directions were $K^*_{Kx} = [-16.1083, 10.7866]$, $K^*_{Ky} = [2.3554, 9.6812]$, and $K^*_{Kz} = [10.7259, 80.5781]$, respectively. Next, to verify the effectiveness of the proposed framework, the effects of the above three experiments were compared and the trajectory tracking error and interaction force changes were analyzed.

3.3. Experimental Results and Discussion. First of all, we give the three-dimensional curves of teaching trajectory, DMP generalization trajectory, and three experimental trajectories in the same space rectangular coordinate system, as shown in Figure 3. As can be seen from the figure, the unprocessed teaching trajectory has serious jitter, and the expected trajectory after DMP generalization becomes smoother.

In the first experiment, the teaching trajectory is used as the reference input of the joint controller of the Baxter robot. The wiping effect is shown in Figure 4(b). It can be seen that the wiping task is not successfully completed under these experimental conditions. The time-varying curve of the interaction force during this process is shown in Figure 5. It shows that the robot performed the wiping task between 6 s and 18 s. However, its interaction force is not large enough and later becomes smaller and smaller close to 0. So, it fails to wipe the handwriting clean.

In the second experiment, although the curve is able to be erased from the surface of the balloon, it can be seen from the force trajectory that the interaction force in this case is very large. As shown in Figure 6, the maximum force in the Z direction is up to 25 N. The actual task figure (Figure 4(c)) also shows that a very serious inward depression occurs on the balloon at this time. Assuming that the interactive environment is not a flexible object as a balloon and the rigidity is very large, it may cause some damage to the robot arm or the interactive object. Therefore, it is obvious that this experiment fails to complete the wiping task well.

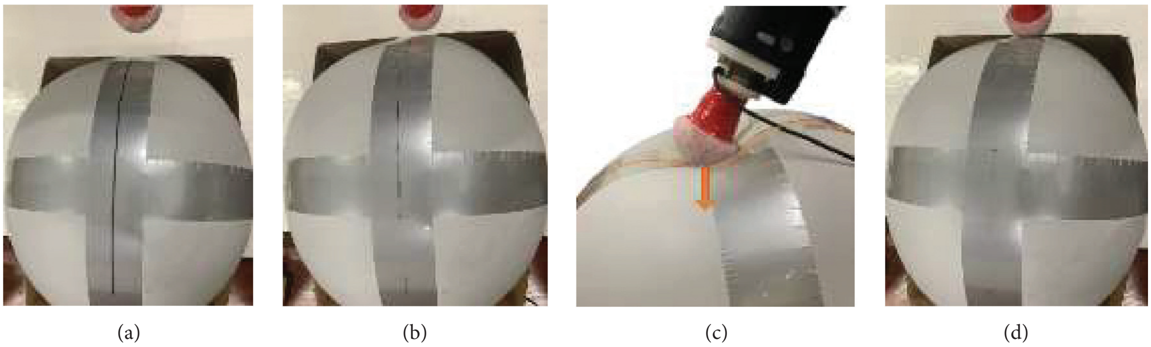

Figure 4: The actual effect of the wiping experiment. (a) Object to be wiped. (b) The wiping effect of the first experiment. (c) The wiping effect of the second experiment. (d) The wiping effect of the third experiment with the proposed framework.

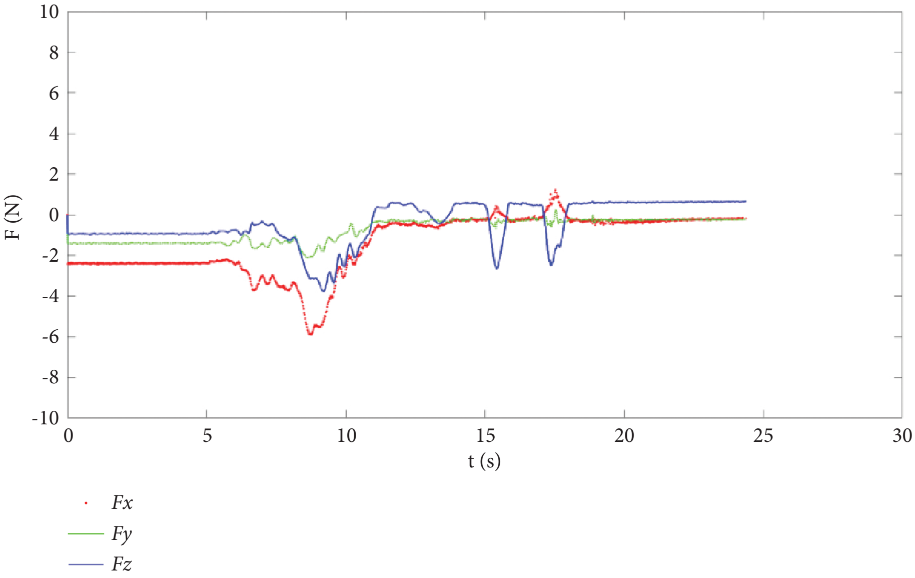

Figure 5: Change curve of interaction force in the first experiment. *Fx*, *Fy*, and *Fz* represent interaction forces in the *X*, *Y*, and *Z* directions.

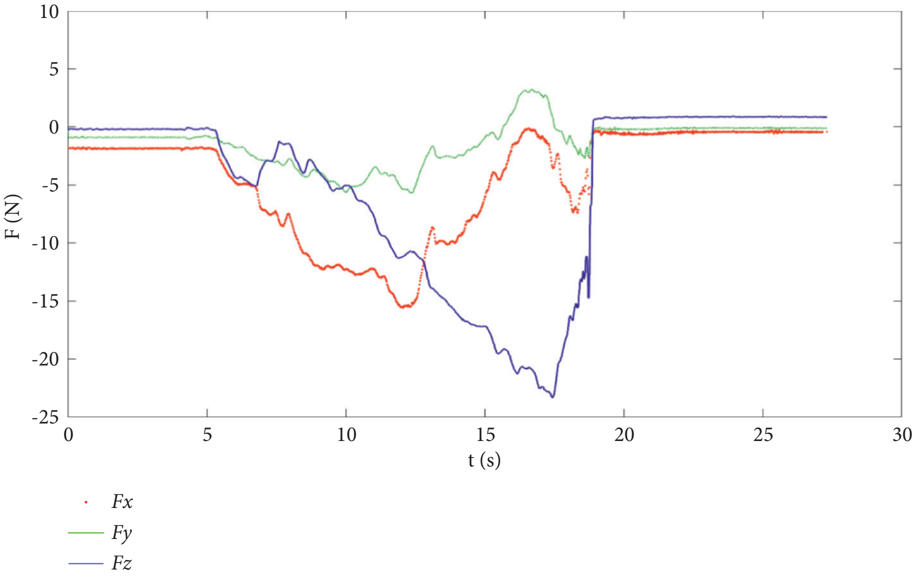

Figure 6: Change curve of interaction force in the second experiment. *Fx*, *Fy*, and *Fz* represent interaction forces in the *X*, *Y*, and *Z* directions.

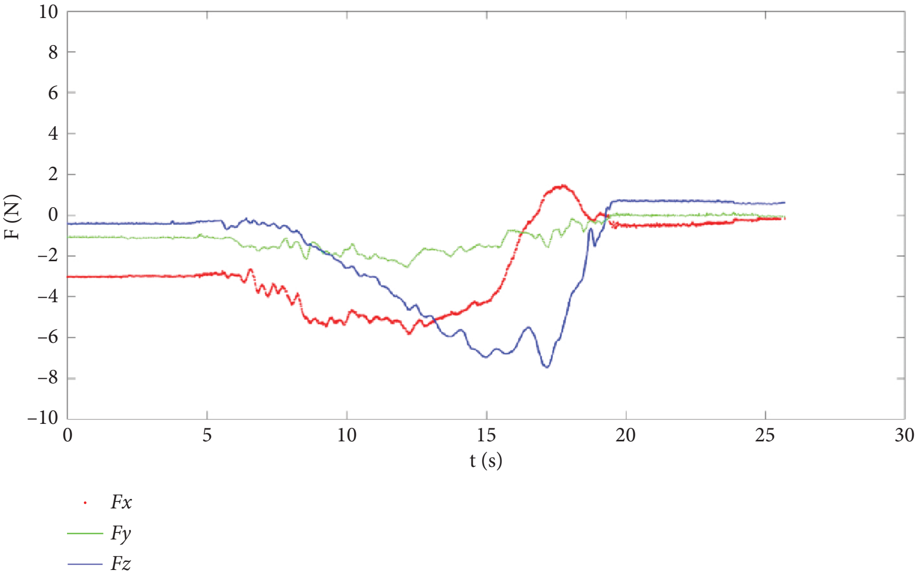

FIGURE 7: Change curve of interaction force in the third experiment. *Fx*, *Fy*, and *Fz* represent interaction forces in the *X*, *Y*, and *Z* directions.

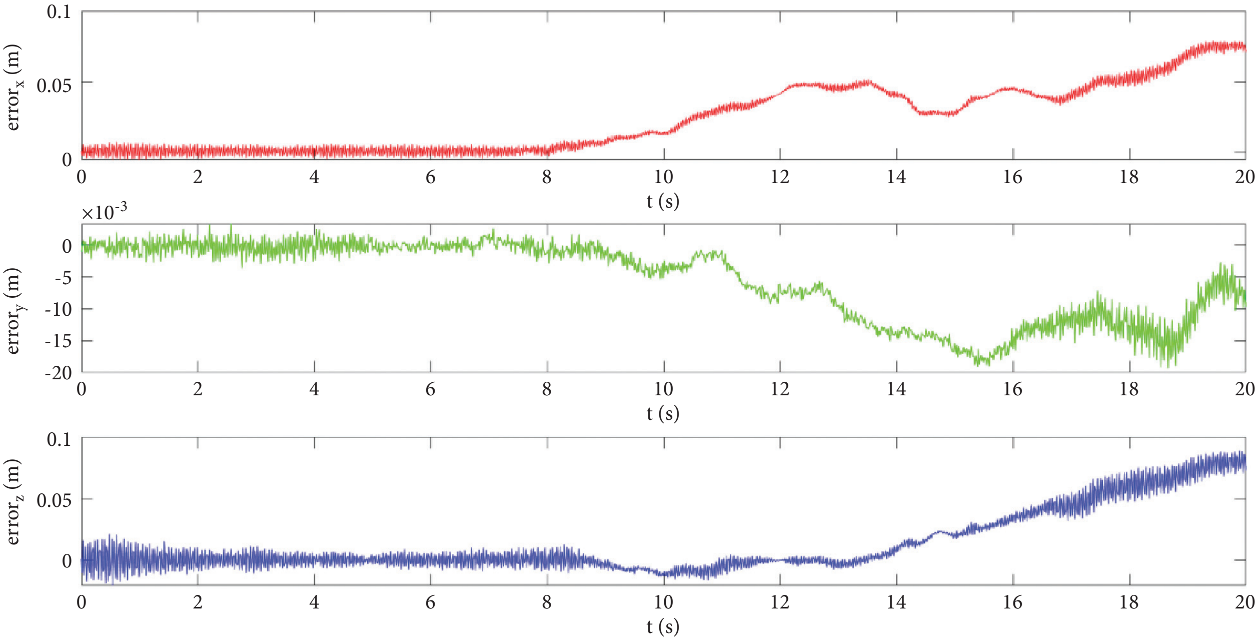

FIGURE 8: Position tracking error curve of the first experiment. The error_x, error_y, and error_z represent the position errors in the *X*, *Y*, and *Z* directions, respectively.

The third experiment is based on the learning framework proposed in this paper. It can be seen from Figure 4(d) that the wiping effect is greatly improved compared with the previous two, and the handwriting curve can be basically wiped clean. The interactive force graph (Figure 7) shows that the robot interacted with the balloon in about 6–20 s, during which the force in the *Z* direction, as the dominant force of the wiping task, changes smoothly between 0 N and 8 N. Compared with the previous two experiments, it is obvious that the interactive force is relatively optimal.

From the three-dimensional trajectory in Figure 3, it can be seen intuitively that there are large errors between the first experiment and the third experiment in terms of the expected trajectory. The interaction between the manipulator end effector and the balloon is basically completed within 20 seconds. The following is to analyze and discuss the change of trajectory tracking error in this stage. As shown in Figure 8, in the first experiment, the tracking error reaches the maximum in the later stage of the wiping task, and the maximum value in the *Z* direction has exceeded 0.1 m. Likewise, the trajectory tracking error of the second experiment (Figure 9) is also relatively large. Only the third experiment can track the expected trajectory well. The tracking error curves in the *X*, *Y*, and *Z* directions

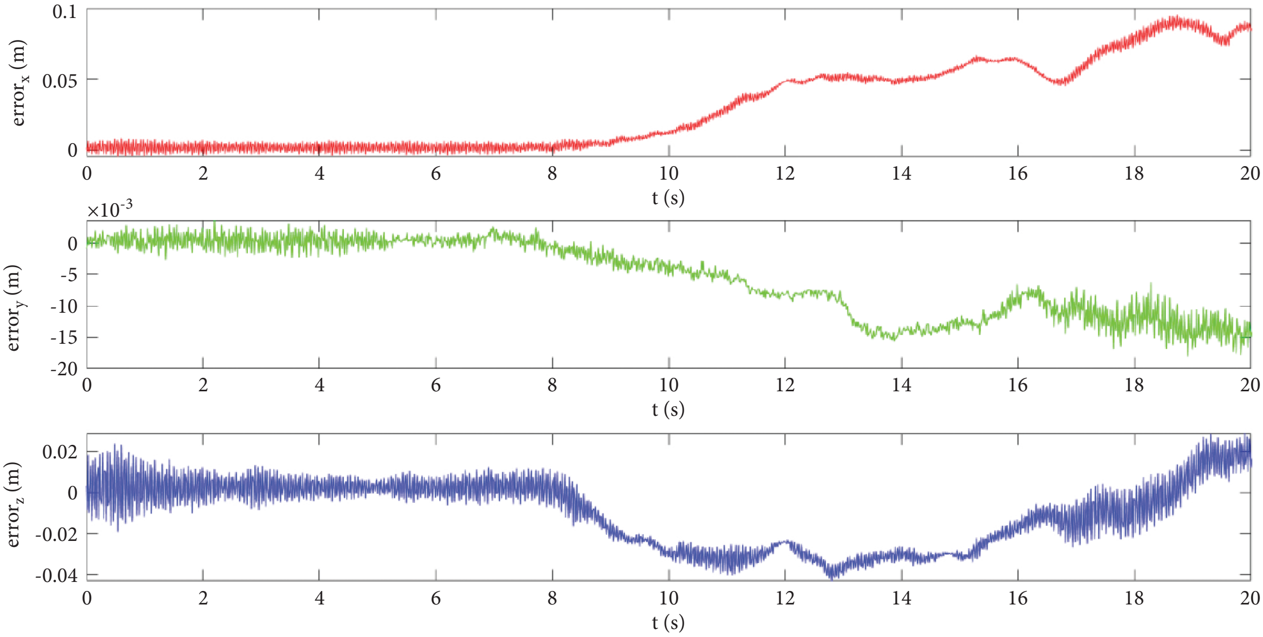

FIGURE 9: Position tracking error curve of the second experiment. The error$_x$, error$_y$, and error$_z$ represent the position errors in the *X*, *Y*, and *Z* directions respectively.

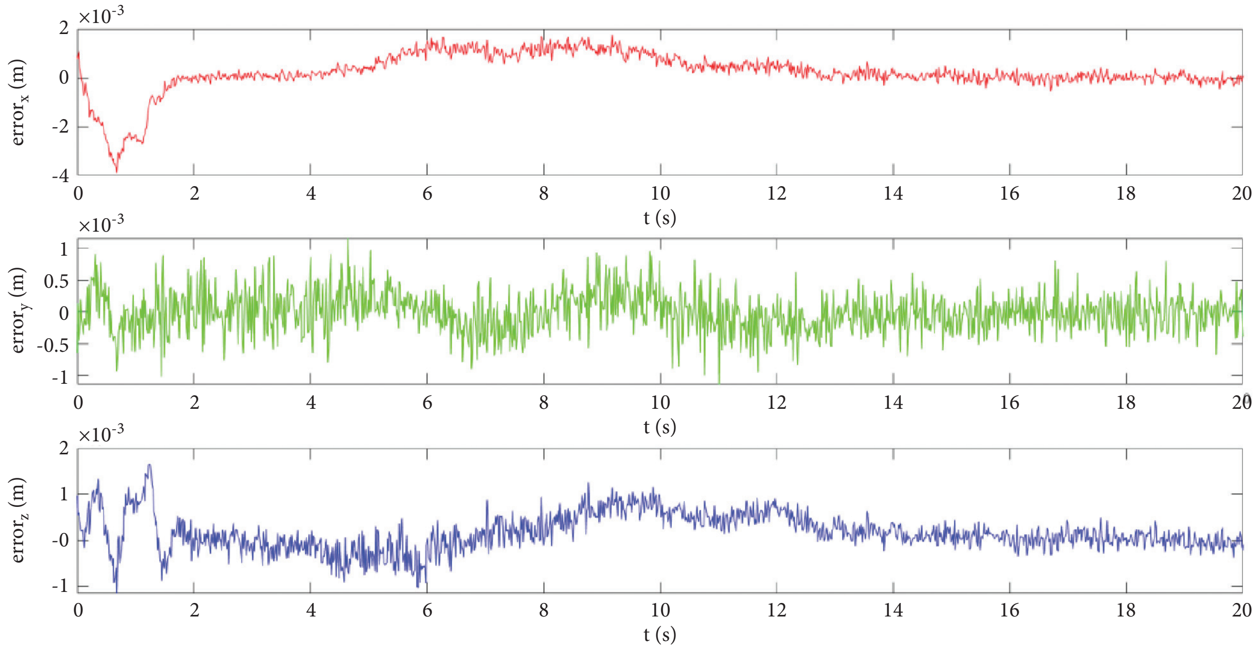

FIGURE 10: Position tracking error curve of the third experiment. The error$_x$, error$_y$, and error$_z$ represent the position errors in the *X*, *Y*, and *Z* directions, respectively.

in the third experiment are shown in Figure 10. The error values all lie within the range of ±0.04 m and, in the process of wiping, the error is basically stable around 0. It also proves that the learning framework proposed in this paper is able to well track the reference trajectory.

4. Conclusions

In this paper, a robot human-like learning framework based on robot and unknown environment interaction was proposed. The LfD approach can make the robot obtain the reference input more quickly and accurately. At the same time, combined with the generalization ability of the DMP model and the compliance control ability of the adaptive optimal admittance model, the interaction performance between the robot and the unknown environment was enhanced. At last, the effectiveness of the proposed framework was verified by the wiping experiment of the balloon surface. Our future work will apply the proposed framework to different complex tasks and environments such as writing on an unknown curved surface, and the learning and generalization of interaction force will also be considered in force control.

Data Availability

The detailed parameters of used model and controller are given in the article. The results are computed on the Matlab 2018a software, while the relevant results are also given in the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partially supported by the Industrial Key Technologies R&D Program of Foshan (2020001006308) and the National Natural Science Foundation of China under Grant no. 61803039.

References

[1] J. Aleotti and S. Caselli, "Robust trajectory learning and approximation for robot programming by demonstration," *Robotics and Autonomous Systems*, vol. 54, no. 5, pp. 409–413, 2006.

[2] C. Yang, C. Chen, N. Wang, Z. Ju, J. Fu, and M. Wang, "Biologically inspired motion modeling and neural control for robot learning from demonstrations," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 11, no. 2, pp. 281–291, 2018.

[3] R. Dillmann, T. Asfour, M. Do et al., "Advances in robot programming by demonstration," *KI - Künstliche Intelligenz*, vol. 24, no. 4, pp. 295–303, 2010.

[4] M. G. Han and J. C. Koo, "DMP accuracy improvement to facilitate learning from demonstration for industrial cooperative robots," *Journal of Institute of Control*, vol. 26, no. 12, pp. 1062–1066, 2010.

[5] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: learning attractor models for motor behaviors," *Neural Computation*, vol. 25, no. 2, pp. 328–373, 2013.

[6] Y. Cohen, O. Bar-Shira, and S. Berman, "Motion adaptation based on learning the manifold of task and dynamic movement primitive parameters," *Robotica*, vol. 39, no. 7, pp. 1299–1315, 2020.

[7] M. Karlsson, A. Robertsson, and R. Johansson, "Autonomous interpretation of demonstrations for modification of dynamical movement primitives," in *Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA)*, Singapore, May 2017.

[8] A. B. Pehlivan and E. Oztop, "Dynamic movement primitives for human movement recognition," in *Proceedings of the IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society IEEE*, Yokohama, Japan, November 2015.

[9] Z. Chao, C. Yang, Z. Chen, and S. L. Dai, "Robot learning human stiffness regulation for hybrid manufacture," *Assembly Automation*, vol. 38, no. 5, pp. 539–547, 2018.

[10] Z.-Y. Liu and H. Qiao, "Multiple ellipses detection in noisy environments: a hierarchical approach," *Pattern Recognition*, vol. 42, no. 11, pp. 2421–2433, 2009.

[11] N. Hogan, "Impedance control: an approach to manipulation: Part I-theory," *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, no. 1, pp. 1–7, 1985.

[12] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 103, no. 2, pp. 126–133, 1981.

[13] K. P. Tee, R. Yan, and H. Li, "Adaptive Admittance Control of a Robot Manipulator under Task Space Constraint," in *Proceedings of the IEEE International Conference on Robotics & Automation*, Anchorage, AK, USA, May 2010.

[14] H. Seraji, "Adaptive Admittance Control: An Approach to Explicit Force Control in Compliant Motion," in *Proceedings of the IEEE International Conference on Robotics & Automation*, Washington, DC, USA, May 2002.

[15] K. Yamane, "Admittance control with unknown location of interaction," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 4079–4086, 2021.

[16] I. Ranatunga, F. L. Lewis, D. O. Popa, and S. M. Tousif, "Adaptive admittance control for human–robot interaction using model reference design and adaptive inverse filtering," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 1, pp. 278–285, 2016.

[17] G. Lai, Z. Liu, Y. Zhang, and C. L. Philip Chen, "Adaptive fuzzy tracking control of nonlinear systems with asymmetric actuator backlash Based on a new smooth inverse," *IEEE Transactions on Cybernetics*, vol. 46, no. 6, pp. 1250–1262, 2017.

[18] H. Huang, T. Zhang, C. Yang, and C. L. P. Chen, "Motor learning and generalization using broad learning adaptive neural control," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 10, pp. 8608–8617, 2020.

[19] C. Yang, X. Wang, L. Cheng, and H. Ma, "Neural-learning-based telerobot control with guaranteed performance," *IEEE Transactions on Cybernetics*, vol. 47, no. 10, pp. 3418–3159, 2017.

[20] H. Huang, C. Yang, and C. L. P. Chen, "Optimal robot-environment interaction under broad fuzzy neural adaptive control," *IEEE Transactions on Cybernetics*, vol. 51, no. 7, pp. 3824–3835, 2021.

[21] H. Zhan, D. Huang, Z. Chen, M. Wang, and C. Yang, "Adaptive dynamic programming-based controller with admittance adaptation for robot-environment interaction," *International Journal of Advanced Robotic Systems*, vol. 17, no. 3, 11 pages, 2020.

[22] C. Mei, J. Yuan, and R. Guan, "Adaptive Unified Impedance and Admittance Control Using Online Environment Estimation," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Kuala Lumpur, Malaysia, December 2018.

[23] L. Roveda, N. Iannacci, F. Vicentini, N. Pedrocchi, F. Braghin, and L. M. Tosatti, "Optimal impedance force-tracking control design with impact formulation for interaction tasks," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 130–136, 2016.

[24] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic

movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.

[25] A. Ude, B. Nemec, T. Petrić, and J. Morimoto, "Orientation in Cartesian Space Dynamic Movement Primitives," in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China, May 2014.

[26] C. G. Atkeson, "Using locally weighted regression for robot learning," in *Proceedings of the - IEEE International Conference on Robotics and Automation*, vol. 2, pp. 958–963, Sacramento, CA, USA, April 1991.

[27] C. Yang, G. Peng, Y. Li, R. Cui, L. Cheng, and Z. Li, "Neural networks enhanced adaptive admittance control of optimized robot-environment interaction," *IEEE Transactions on Cybernetics*, vol. 49, no. 7, pp. 2568–2579, 2018.

[28] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, 2012.

[29] J. Marti-Saumell, A. Santamaria-Navarro, C. Ocampo-Martinez, and J. Andrade-Cetto, "Multi-task closed-loop inverse kinematics stability through semidefinite programming," in *Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA) IEEE*, Paris, May 2020.

[30] D. Huang, H. Zhan, and C. Yang, "Impedance model-based optimal regulation on force and position of bimanual robots to hold an object," *Complexity*, vol. 2020, no. 3, pp. 1–13, 2020.

[31] D. Siciliano, "A closed-loop inverse kinematic scheme for online joint-based robot control," *Robotica*, vol. 8, no. 3, pp. 231–243, 1990.