2020 5th International Conference on Advanced
Robotics and Mechatronics (ICARM)

Fuzzy Adaptive Control-based Real-time Obstacle Avoidance under Uncertain Perturbations

Yingbai Hu[1], Guang Chen[2], Longbin Zhang[3]*, Hang Su[4], Mengyao Li[5]
Yunus Schmirander[4], Hu Cao[1], Alois Knoll[1]

Abstract— **Dynamic Movement Primitives (DMPs) framework is a powerful approach to imitate motor skills, which has outstanding characteristics, such as convergence to the goal position and good imitation performance. Considering complex motion scenes of manipulators, such as changing the goal position or adding obstacles, the original DMPs framework is not sufficient for the requirements. In this paper, we propose a learning control-based hierarchical control strategy to adapt to new goal positions and avoid obstacles: the high-level learning scheme is targeted at imitating the motor skill and generating the optimization trajectory for obstacle avoidance; the lower-level control scheme focuses on the safety and stability of the robot's movement with unknown disturbances. Firstly, the enhanced DMPs framework is presented to imitate the trajectory from human demonstrations, where the novel DMPs can adapt to new goal position with the changing goal, and avoid single or multiple obstacles. Then, the fuzzy adaptive control method is employed to control redundant manipulators, where the fuzzy logic system (FLS) is incorporated to approximate an unknown nonlinear function term of the unknown disturbance. Finally, the effectiveness of the proposed learning-control strategy is demonstrated with simulation results. The results show that the developed hierarchical strategy has good performance for new goal adaptation and obstacle avoidance.**

I. INTRODUCTION

Manipulator autonomous obstacle avoidance is a challenging topic prominent in the past decade. Generally, the main issue is how to plan a smooth trajectory of the end-effector which adapts as the task changes during the presence of many obstacles. In [1], the continuous genetic algorithm (CGA) is proposed for trajectory planning of redundant manipulators, where CGA can solve the singularity problem. In [2], the authors present an reinforcement learning-based double neural networks algorithm to manage the obstacle avoidance problem, where the reinforcement learning method is used to compute the optimization path. In [3], the particle swarm optimization (PSO) is employed for motion planning with multi-obstacle constraints, while PSO is used to obtain the optimal trajectory with the condition constraints.

However, the methods mentioned above only consider fixed goal positions, and cannot solve the problem of changing goal positions. We want to design a powerful method to solve more complex problems, such as adaption to new goal positions with multi-obstacle constraints. Imitation learning has a pivotal role in robotic technology and can learn new motor skills from human demonstrations and interactions with the environment. In robotic learning, a complex movement can be deemed as a combination of simple movements that can be easily encoded by learning methods. Dynamical Movement Primitives (DMPs) framework is a well-established method that can model nonlinear motions [4], [5], and provide powerful learning mechanisms, such as combining with machine learning methods, e.g. reinforcement learning.

In [6], dual-arm robots based on DMPs cooperate to cut a vegetable by human demonstrations. In [7], reinforcement learning fusion with DMPs is employed to learn the grasping actions with a number of iterations in the external disturbance environment. In this paper, DMPs are applied to learning human movements for a mobile robot. The learning scheme is a novel human-robot interaction method that can imitate the motion from human demonstrations. Therefore, the mobile robot is able to reproduce the demonstrations by DMPs, which lead to the reduction of complexity for motion planning tasks. In [8], DMPs are utilized to learn the robot's joint trajectory fusion with sensor signals. However, the original DMPs framework has limitations that cannot solve the obstacle avoidance problem, and its adaptability is not sufficient for changing goal positions [9], [10]. In this paper, we modify the original DMPs method and apply it to the manipulator movement learning with multi-obstacle constraints. Compared with the original DMPs framework, the enhanced DMPs are more powerful and can be adapted to changing goal positions with more natural trajectories.

In addition, for robotic control, various methods are applied to dynamic systems. In [11], [12], the global asymptotic PID is proposed for a robot system with input constraints.

This work was supported by the Human Brain Project SGA2, under the Specific Grant Agreement No. 785907, and also supported by the Zhejiang Lab's International Talent Fund for Young Professionals. This work was supported by the Youth Program of National Natural Science Foundation of China (Grant No. 61906138) and Shanghai AI Innovation Development Program 2018.

[1]Yingbai Hu, Hu Cao and Alois Knoll are with Chair of Robotics, Artificial Intelligence and Realtime Systems, Technical University of Munich, Munich, 85748, Germany. yingbai.hu@tum.de, knoll@in.tum.de, caohu@hnu.edu.cn.

[2]Guang Chen is with College of Automotive Engineering, Tongji University, Shanghai, 201804, China. He is also with Chair of Robotics, Artificial Intelligence and Real-time Systems Technical University of Munich, Munich, 85748, Germany. guangchen@tongji.edu.cn.

[3]Longbin Zhang is with department of Engineering Mechanics and KTH MoveAbility Lab, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden. *corresponding author, longbin@kth.se.

[4]Hang Su and Yunus Schmirander are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133, Milano, Italy. hang.su@polimi.it, yunus.schmirander@mail.polimi.it.

[5]Mengyao Li is with Guangdong Provincial Key Laboratory of Robotics and Intelligent System, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, 518055, China. my.li@siat.ac.cn.

In [13], [14], [15], the authors explore the adaptive control system to solve the input saturation problem. In [16], [13], the closed-loop based adaptive control algorithm is applied to a exoskeleton robot. However, the robot system is highly nonlinear and the disturbance needs to be compensated. Inspired by [17], [18], [19], the fuzzy adaptive control strategy is presented to control the manipulator to compensate for the external disturbance. Overall, the learning-control system is proposed, where the high-level learning scheme is targeted at imitating the motor skill and generating the optimal trajectory for obstacle avoidance; while the lower-level control scheme focuses on the safety and stability of the robot's movement with unknown disturbances.

II. Adaptive Trajectories: Goal Adaptation and Obstacle Avoidance

A. Dynamical Movement Primitives

DMPs framework is a dynamic system that can be applied to imitation learning, even combined with machine learning methods for motor skills learning. The dynamic system consists of a linear spring-damper and a nonlinear item, which are defined as follows:

$$\ddot{X} = K_p (X_g - X) - K_d \dot{X} + (X_g - X_0) f(s, \rho) \tag{1}$$

$$f(s, \rho) = \frac{\sum_{j=1}^{N} \varphi_j(s) \rho_j}{\sum_{j=1}^{N} \varphi_j(s)} s \tag{2}$$

$$\dot{s} = -\lambda s \tag{3}$$

$$\varphi_j(s) = \exp\left(-\frac{1}{2d_j}(s - c_j)^2\right). \tag{4}$$

Equation (1) describes the transformation system, where $\left\{\ddot{X}, \dot{X}, X\right\}$ denote the robot's position, velocity and acceleration in Cartesian space; K_p and K_d are the corresponding stiffness matrix and damping matrix, respectively; X_0 and X_g are the initial and target values of the position; $f(s, \rho)$ is the nonlinear forcing item. Equation (3) is the canonical system acting as the decay factor, and $\lambda > 0$ is the constant parameter; d_j and c_j in equation (4) are the bandwidth and center of the Gaussian kernel function. Obviously, when $f(s, \rho) = 0$, the dynamic system becomes linear.

B. Enhanced DMPs for Goal Adaptation

The traditional framework of DMPs cannot guarantee the generation of a smooth trajectory when the start or target positions are altered. In addition, it cannot ensure the adaption of the robot to a new goal position. Hence, we introduce the novel framework of a nonlinear dynamic system, which is an extended form of the traditional DMPs.

Firstly, the canonical system in equation (3) is modified as,

$$\dot{s} = \begin{cases} -\lambda_1, & s \geq 0 \\ 0, & s < 0 \end{cases} \tag{5}$$

where $\lambda_1 > 0$ is the decay rate of the canonical system. The nonlinear forcing item $f(s, \rho)$ depends only on s, which acts as a global clock for the entire system. Secondly, the transformation system in (1) is reformulated as,

$$\ddot{X} = K_p (X_g - X) - K_d \dot{X} - K_p (X_g - X_0) s + K_p f(s, \rho). \tag{6}$$

Apart from the modified canonical system (5) and transformation system (6), the function $f(s, \rho)$ is identical to (2).

The novel transformation system, (6), is a forward procedure. To model the motion from human demonstrations, we need to obtain the shape parameter ρ in a backward procedure using regression methods.

$$f_{demo} = \frac{1}{K_p}\ddot{X} - \frac{1}{K_p}\left[K_p (X_g - X) - K_d \dot{X} - K_p (X_g - X_0) s\right] \tag{7}$$

Finally, to obtain the shape parameter ρ from the dataset, the Locally Weighted Regression (LWR) method is used to learn ρ by regression. The objective function is defined as

$$\Phi_j = \sum_{t=1}^{N} \varphi_j(t) \left\| f_{demo} - (\rho_j s + b_j) \right\|^2. \tag{8}$$

The solution of regression is calculated as,

$$\begin{bmatrix} \rho_j \\ b_j \end{bmatrix} = \begin{bmatrix} R_{s^2} & R_s \\ R_s & R_\rho \end{bmatrix}^{-1} \begin{bmatrix} R_{sy} \\ R_y \end{bmatrix} \tag{9}$$

$$\begin{aligned} R_\rho &= \sum_{t=1}^{N} \varphi_j(t) \\ R_s &= \sum_{t=1}^{N} \varphi_j(t) s(t) \\ R_{s^2} &= \sum_{t=1}^{N} \varphi_j(t) s^2(t) \\ R_{sy} &= \sum_{t=1}^{N} \varphi_j(t) s(t) f_{demo}. \end{aligned}$$

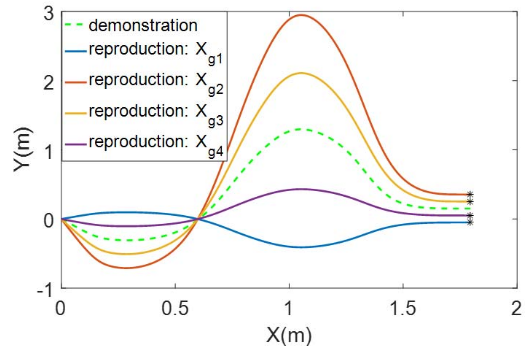
To test the proposed methods, we change the target position and compare the original DMPs with our enhanced DMPs.

The comparison results of the original DMPs and our enhanced DMPs for target position adaptation are shown in Fig. 1. In Fig. 1(a), the original DMPs model the trajectory and converge to a new target position, but the shape of the motion is adjusted overshoot.

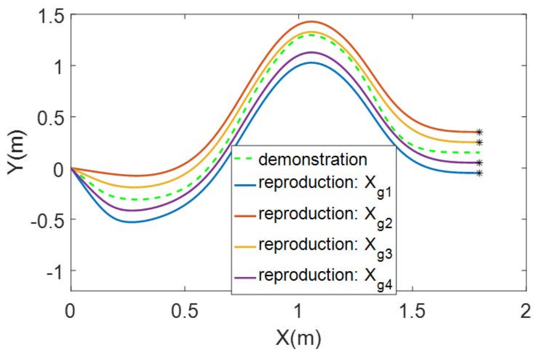
In Fig. 1(b), it is suitable for natural movements with changing target positions. Obviously, the enhanced DMPs can improve movement adaptation.

C. Obstacle Avoidance

The modified DMPs in (6) can adapt to the new target position but are still unable to deal with complex requirements, such as autonomous obstacle avoidance. Hence, in this part, we further extend the novel DMPs in (6) as a new formulation that can avoid a single obstacle and even many obstacles.



(a) Learning with the original DMPs.



(b) Learning with the enhanced DMPs.

Fig. 1. The original DMPs vs the enhanced DMPs for changing targets.

The new formulation of DMPs is reformulated as

$$\ddot{X} = K_p\left(X_g - X\right) - K_d\dot{X} - K_p\left(X_g - X_0\right)s + K_pf\left(s,\rho\right) + P\left(X,\dot{X}\right). \tag{10}$$

For the single-obstacle scene, the relationship between steering angle θ and velocity $\dot{\theta}$ can be defined as,

$$\dot{\theta} = \mu\theta\exp\left(-\alpha\left|\theta\right|\right) \tag{11}$$

where $\mu > 0$ and $\alpha > 0$ are constants; $\dot{\theta}$ is the steering angle velocity.

$$\ddot{X} = T_R\dot{X}\dot{\theta} \tag{12}$$

where T_R is the rotation matrix of axis $L = (M - X) \times \dot{X}$ with rotation angle of $\pi/2$. $M = [M_x, M_y, M_z]$ is the position of the obstacle, X is the position of robot's end-effector, while $\dot{X}$ and $\ddot{X}$ correspond to its velocity and acceleration, respectively.

$$P\left(X,\dot{X}\right) = \mu T_R\dot{X}\theta\exp\left(-\alpha\theta\right) \tag{13}$$

$$\theta = \cos^{-1}\left(\frac{\left(M - X\right)^T\dot{X}}{\left(\left|M - X\right|\left|\dot{X}\right|\right)}\right)$$

So, the modified DMPs equation in (10) can be rewritten as,

$$\ddot{X} = K_p\left(X_g - X\right) - K_d\dot{X} - K_p\left(X_g - X_0\right)s + K_pf\left(s,\rho\right) + \mu T_R\dot{X}\theta\exp\left(-\alpha\theta\right) \tag{14}$$

where the state variable $[X, \dot{X}] = [X_g, 0]$ is the stationary point, to which all the states converge from random initial states.

Proof: To prove global convergence of the modified DMPs in (14), the Lypunov function is set as,

$$V\left(X,\dot{X}\right) = \frac{1}{2}\left(X_g - X\right)^TK_p\left(X_g - X\right) + \frac{1}{2}\dot{X}^T\dot{X}. \tag{15}$$

Then, the time derivative of $V\left(X,\dot{X}\right)$ is obtained as

$$\begin{aligned}\dot{V} &= \nabla_X V^T\dot{X} + \nabla_{\dot{X}}V^T\ddot{X} \\ &= -\left(X_g - X\right)^TK_p\dot{X} + \dot{X}^T\ddot{X} \\ &= -\dot{X}^TK_p\left(X_g - X\right) + \dot{X}^TK_p\left(X_g - X\right) - \dot{X}^TK_d\dot{X} \\ &\quad - K_p\dot{X}^T\left(X_g - X\right)s + \dot{X}^TK_pf\left(s,\rho\right) \\ &\quad + \mu\dot{X}^TT_R\dot{X}\theta\exp\left(-\alpha\theta\right) \\ &= -\dot{X}^TK_d\dot{X} - \underbrace{K_p\dot{X}^T\left(X_g - X\right)s}_{C_1} + \underbrace{\dot{X}^TK_pf\left(s,\rho\right)}_{C_2} \\ &\quad + \underbrace{\mu\dot{X}^TT_R\dot{X}\theta\exp\left(-\alpha\theta\right)}_{C_3}.\end{aligned} \tag{16}$$

Obviously, if $t \to \infty$, $X \to X_g$ and $s \to 0$, then $C_1 \to 0$; again when $f\left(s,\rho\right) \to 0$, then $C_2 \to 0$; since T_R is the rotation matrix with rotation angle of $\pi/2$, then $\dot{X}^TT_R\dot{X} = 0$, so $C_3 \to 0$. Therefore, we can conclude

$$\dot{V} \to -\dot{X}^TK_d\dot{X} \leq 0.$$

The proof is finished. ■

However, we hope that the function in (13) is more powerful i.e. has better obstacle avoidance and is adaptive to a more complex environment. If there exist more obstacles, the added item in (13) needs to be further modified. The details of the novel modified item can be reformulated as,

$$P\left(X,\dot{X}\right) = \mu\sum_{i=1}^{O}T_{R_i}\dot{X}\theta_i\exp\left(-\alpha\theta_i\right) \tag{17}$$

$$\theta_i = \cos^{-1}\frac{\left(\left(M_i - X\right)^T\dot{X}\right)}{\left|M_i - X\right|\left|\dot{X}\right|} \tag{18}$$

where O denotes the number of obstacles.

To test the obstacle avoidance performance of the DMPs, we randomly put several obstacles on the trajectory after the demonstration. From Fig. (2(a))–Fig. (2(d)), it can be seen that the modified DMPs can avoid the single obstacle and even multiple obstacles well.

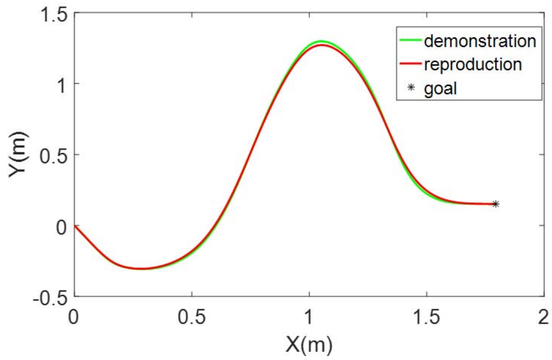
III. Fuzzy Adaptive Control for Manipulator

A. Dynamic System of Manipulator

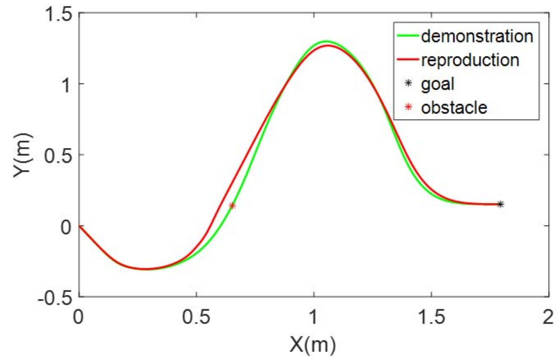
Considering the dynamic model of n-DOFs (n=7) Kuka manipulator shown in Fig. 3,

$$B\left(q\right)\ddot{q} + C\left(q,\dot{q}\right)\dot{q} + G\left(q\right) = \tau_c + \tau_e \tag{19}$$

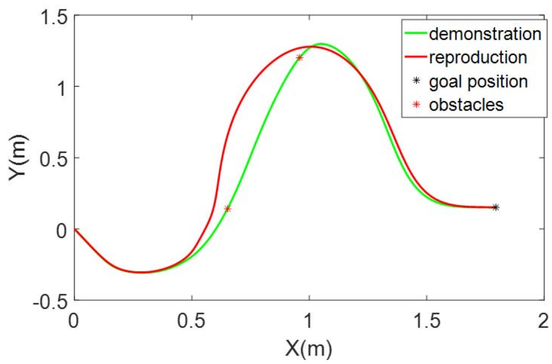
where $q \in R^{n\times n}$ and $\dot{q} \in R^{n\times n}$ are the joint position vector and velocity vector, respectively; $B\left(q\right) \in R^{n\times n}$



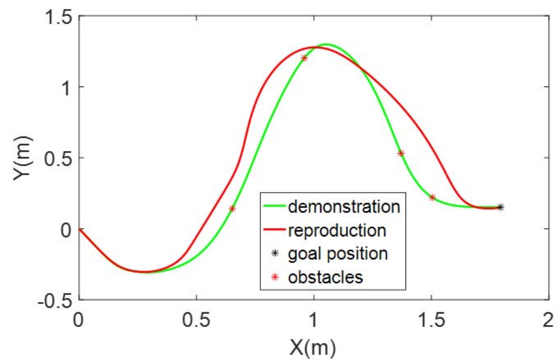
(a) Learning without obstacles.



(b) Learning with a single obstacle.



(c) Learning with many obstacles.



(d) Learning with more obstacles.

Fig. 2. Imitation learning in 2D space for obstacle avoidance.

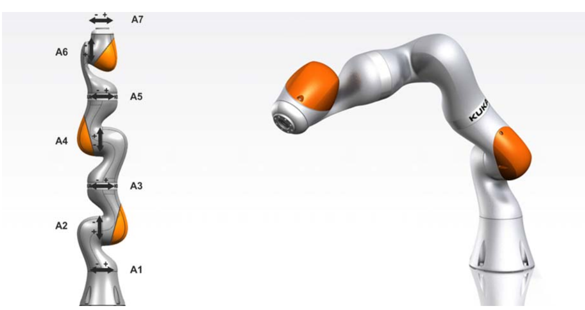


Fig. 3. 7-DOFs kuka robot's model.

denotes the inertia matrix; $C(q, \dot{q}) \in R^{n\times n}$ is the Coriolis and Centrifugal effects; $G(q) \in R^n$ represents the gravity matrix; τ_c is the joint torque variable, and τ_e is the torque of external unknown disturbances.

The forward kinematics model of a redundant manipulator is formulated as,

$$\dot{X} = J(q)\dot{q} \tag{20}$$

where $J(q)$ is the Jacobian matrix of the redundant manipulator.

To describe the robot system conventionally, the dynamic system in (19) should be reformulated in a standard nonlinear control affine form. We define,

$$\begin{aligned} &z_1 = q, z_2 = \dot{q} \\ &f(z_1, z_2) = -B^{-1}(C\dot{q} + G) \\ &g(z_1) = B^{-1} \\ &u = \tau = \tau_c + \tau_e. \end{aligned}$$

Therefore, the dynamic system in (19) can be rewritten as,

$$\dot{z}_1 = z_2 \tag{21}$$

$$\dot{z}_2 = f(z_1, z_2) + g(z_1)u. \tag{22}$$

B. Fuzzy Approximation system

It is difficult to obtain the accurate dynamic model of the manipulator due to unknown disturbances. In this paper, a fuzzy logic system is employed for unknown items. Firstly, the fuzzy rules are given as,

$$\Lambda_i: \text{if } \ Z_1 \text{ is } K_1^i \text{ and } \ldots \text{and } Z_l \ \text{ is } K_l^i, \text{ then } h \text{ is } \ h_i \tag{23}$$

where h_i is the i-item fuzzy rule.

$$h(Z) = \frac{\sum_{l=1}^{m} h^l \left(\prod_{j=1}^{n} \varphi(Z_j)\right)}{\sum_{l=1}^{m} \left(\prod_{j=1}^{n} \varphi(Z_j)\right)} = \Theta^T S(Z) \tag{24}$$

where m is the number of fuzzy rules; $Z = [Z_1, Z_2, \ldots, Z_n] \in R^n$ is the input variable; $S(Z)$ is the known fuzzy basis function; Θ denotes the adaptable weight parameters.

Therefore, the approximate function $h(Z)$ can be defined as,

$$\tilde{h}(Z) = \tilde{\Theta}^T S(Z) \tag{25}$$

where $Z \in R^n$, $\tilde{\Theta} \in \Lambda_m$, $S(Z) \in R^m$.

The objective function of optimization parameters Θ^* is defined as,

$$\Theta^* = \arg\min \left\{ \sup \left| h(Z) - \tilde{h}\left(Z|\tilde{\Theta}\right) \right| \right\} \tag{26}$$

$$M_\Theta = \left\{ \tilde{\Theta} \left| \left\| \tilde{\Theta} \right\| \leq M_\Theta \right. \right\} \tag{27}$$

where the parameter M_Θ corresponds to the bounds of $\tilde{\Theta}$ which are defined by the user.

According to the optimization result in (26), the unknown function $h(Z)$ can be rewritten as,

$$h(Z) = \Theta^{*T} S(Z) + \varepsilon \tag{28}$$

where ε presents the minimum approximation error. Actually, the minimum approximation error ε has an upper bound $\varepsilon^+ > 0$ according to the theory analysis of the fuzzy logical system,

$$|\varepsilon| \leq \varepsilon^+. \tag{29}$$

For the robot system,

$$\Theta^* = \lambda \mathcal{E} \varpi S(Z) \tag{30}$$

where $\lambda \in R^{n \times n}$ is the coefficient diagonal matrix which is used to scale the updating rate; $\mathcal{E} = [X_r - X(q)]$ is the position error in Cartesian space; $\varpi \in R^2$ is the last column of a symmetric positive definite matrix according to Lyapunov theory.

The torque approximation of the unknown disturbance using the fuzzy system can be defined as,

$$\tau_h = -J^T \Theta S(Z) \tag{31}$$

where τ_h is used to compensate for the effects of disturbances. Finally, the input torque can be reformulated as,

$$\tau_d = \tau_c + \tau_h. \tag{32}$$

C. Sliding Mode Control

In this part, sliding mode control is applied to the robot system and has better performance for modeling errors and anti-disturbance. The disturbance $\delta \in R^n$ is considered in the state space form, where δ is bounded such that $|S_i| \leq \xi_i (\xi_i > 0)$.

Thus the equations in (21)–(22) can be reformulated as,

$$\dot{z}_1 = z_2 \tag{33}$$

$$\dot{z}_2 = f(z_1, z_2) + g(z_1)(u + \delta). \tag{34}$$

The tracking error of the end-effector is defined as: $e_c = z_1 - z_{1d}$. The sliding mode surface is defined as,

$$S = \dot{e}_c + \alpha e_c \tag{35}$$

where $\alpha = diag(\alpha_1, \alpha_2, \ldots, \alpha_n)$, and $\alpha_i > 0$.

According to sliding mode theory, we aim to design an input signal u to ensure that $S = 0$. Firstly, the Lyapunov function is set as,

$$L = \frac{1}{2} S^T S. \tag{36}$$

So, the time derivative of L can be concluded as,

$$\begin{aligned} \dot{L} &= S^T \dot{S} \\ &= S^T (\ddot{e}_c + \alpha \dot{e}_c) \\ &= S^T (f(z_1, z_2) + g(z_1) u + g(z_1) \delta - \ddot{z}_{1d} + \alpha (z_2 - \dot{z}_{1d})). \end{aligned} \tag{37}$$

To ensure the stability of the control system, according to the equation of $\dot{L}$, the actual control law is defined as,

$$u = g(z_1)^{-1} (-f(z_1, z_2) + \ddot{z}_{1d} - \alpha (z_2 - \dot{z}_{1d}) - \kappa \mathrm{sgn}(S)) \tag{38}$$

where $\kappa = diag\{k_1, k_2, \ldots, k_n\}$, and $k_i > 0$ are constants. Substituting (38) into (37), the time derivative of L is obtained as,

$$\begin{aligned} \dot{L} &= S^T (g(z_1) \delta - \kappa \mathrm{sgn}(S)) \\ &= -\sum_{i=1}^{n} k_i |S_i| + S^T B^{-1} \delta \\ &\leq -\sum_{i=1}^{n} |S_i| (k_i - \eta \xi_i). \end{aligned} \tag{39}$$

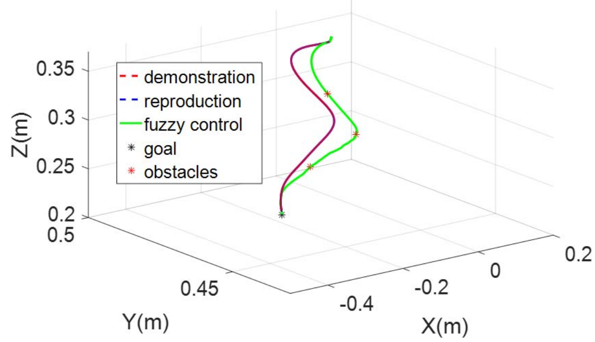
If the gain parameters satisfy that $k_i > \eta \xi_i$, then $\dot{L} < 0$ which can conclude that the system is asymptotically stable. Finally, the control law is presented as,

$$u = C\dot{q} + G + B\left(\ddot{X}_d - \alpha\left(z_2 - \dot{X}_d\right) - \kappa \mathrm{sgn}(S)\right). \tag{40}$$

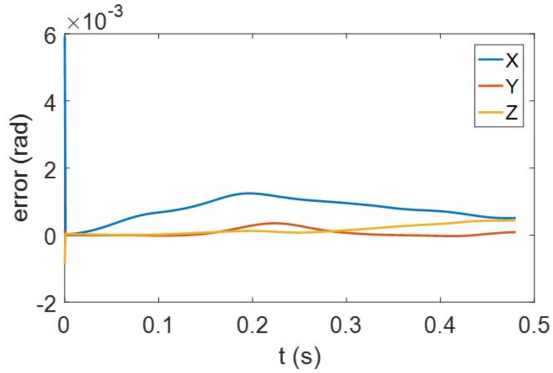
IV. Simulation

In our simulation scenarios, some obstacles are randomly placed in the environment. It is a challenging scene that lets the robot achieve new skills of adaption to the new target and obstacle avoidance. Firstly, the motor skills are encoded using the enhanced DMPs by human demonstrations, where a human holds the robot to teach the skills by demonstrations. After the collection of the dataset, the regression procedure is conducted using LWR, thus the shape parameter ρ is obtained, and then the imitation results are achieved via the forward DMPs. Finally, the sliding mode control-based fuzzy adaptive control is applied to the 7-DOFs Kuka model, which shows good performance in terms of trajectory tracking.

The tracking results using fuzzy adaptive control are shown in Fig. 4. Obviously, we can see that the real trajectory can successfully avoid obstacles. From Figs. 4(a)–4(b), the actual trajectory almost coincides with the learned trajectory, and the tracking errors of the X-Y-Z coordinates gradually converge to zero. Therefore, the proposed learning-control strategy can not only learn the skills, adapt new goal positions and avoid obstacle, but also tracking the desired trajectory in Cartesian space.



(a) Tracking results in Cartesian space.



(b) Tracking errors.

Fig. 4. Tracking results in Cartesian space using fuzzy adaptive control.

V. CONCLUSIONS

In this paper, we proposed a learning control-based hierarchical control strategy to adapt to new goal position and avoid obstacles: the high-level learning scheme is targeted at imitating the motor skill and generating the optimization trajectory for obstacle avoidance; the lower-level control scheme focuses on the safety and stability of robot's movement with unknown disturbances.

For the learning system, the modified DMPs have better performance regarding movement adaptation than the original DMPs. In addition, we further modify the DMPs to avoid a single obstacle and even multiple obstacles. To ensure the safety and stability of the robot, a sliding mode control-based fuzzy adaptive controller is proposed. From the learning-control results, we can conclude that the proposed method can stably and safely avoid multiple obstacles. In further work, we will test the proposed method with physical experiments, even more complex scenarios.

REFERENCES

[1] Z. S. Abo-Hammour, O. M. Alsmadi, S. I. Bataineh, M. A. Al-Omari, and N. Affach, "Continuous genetic algorithms for collision-free cartesian path planning of robot manipulators," *International Journal of Advanced Robotic Systems*, vol. 8, no. 6, p. 74, 2011.

[2] M. Duguleana, F. G. Barbuceanu, A. Teirelbar, and G. Mogan, "Obstacle avoidance of redundant manipulators using neural networks based reinforcement learning," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 2, pp. 132–146, 2012.

[3] J.-J. Kim and J.-J. Lee, "Trajectory optimization with particle swarm optimization for manipulator motion planning," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 620–631, 2015.

[4] Y. Hu, G. Chen, X. Ning, J. Dong, S. Liu, and A. Knoll, "Mobile robot learning from human demonstrations with nonlinear model predictive control," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 5057–5062.

[5] Y. Hu, X. Wu, P. Geng, and Z. Li, "Evolution strategies learning with variable impedance control for grasping under uncertainty," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 10, pp. 7788–7799, 2018.

[6] R. Lioutikov, O. Kroemer, G. Maeda, and J. Peters, "Learning manipulation by sequencing motor primitives with a two-armed robot," in *Intelligent Autonomous Systems 13*. Springer, 2016, pp. 1601–1611.

[7] Z. Li, T. Zhao, F. Chen, Y. Hu, C.-Y. Su, and T. Fukuda, "Reinforcement learning of manipulation and grasping using dynamical movement primitives for a humanoidlike mobile manipulator," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 1, pp. 121–131, 2017.

[8] 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, 2018.

[9] S. Schaal, J. Peters, J. Nakanishi, and A. Ijspeert, "Learning movement primitives," in *Robotics research. the eleventh international symposium*. Springer, 2005, pp. 561–572.

[10] D.-H. Park, H. Hoffmann, P. Pastor, and S. Schaal, "Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields," in *Humanoids 2008-8th IEEE-RAS International Conference on Humanoid Robots*, 2008, pp. 91–98.

[11] Y. Su, P. C. Muller, and C. Zheng, "Global asymptotic saturated pid control for robot manipulators," *IEEE Transactions on Control systems technology*, vol. 18, no. 6, pp. 1280–1288, 2009.

[12] Y. Hu, Z. Li, G. Li, P. Yuan, C. Yang, and R. Song, "Development of sensory-motor fusion-based manipulation and grasping control for a robotic hand-eye system," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1169–1180, 2016.

[13] Z. Li, J. Li, S. Zhao, Y. Yuan, Y. Kang, and C. P. Chen, "Adaptive neural control of a kinematically redundant exoskeleton robot using brain-machine interfaces," *IEEE transactions on neural networks and learning systems*, 2018.

[14] Y. Liu, W. Su, Z. Li, G. Shi, X. Chu, Y. Kang, and W. Shang, "Motor-imagery-based teleoperation of a dual-arm robot performing manipulation tasks," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 11, no. 3, pp. 414–424, 2018.

[15] Z. Li, Y. Yuan, L. Luo, W. Su, K. Zhao, C. Xu, J. Huang, and M. Pi, "Hybrid brain/muscle signals powered wearable walking exoskeleton enhancing motor ability in climbing stairs activity," *IEEE Transactions on Medical Robotics and Bionics*, vol. 1, no. 4, pp. 218–227, 2019.

[16] T.-F. Ding, M.-F. Ge, Z.-W. Liu, Y.-W. Wang, and H. R. Karimi, "Discrete-communication-based bipartite tracking of networked robotic systems via hierarchical hybrid control," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 4, pp. 1402–1412, 2020.

[17] Z. Li, C.-Y. Su, G. Li, and H. Su, "Fuzzy approximation-based adaptive backstepping control of an exoskeleton for human upper limbs," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 3, pp. 555–566, 2014.

[18] Y. Wang, H. R. Karimi, H.-K. Lam, and H. Yan, "Fuzzy output tracking control and filtering for nonlinear discrete-time descriptor systems under unreliable communication links," *IEEE Transactions on Cybernetics*, 2019.

[19] Z. Li, B. Huang, A. Ajoudani, C. Yang, C.-Y. Su, and A. Bicchi, "Asymmetric bimanual control of dual-arm exoskeletons for human-cooperative manipulations," *IEEE Transactions on Robotics*, vol. 34, no. 1, pp. 264–271, 2017.