DMPs-based skill learning for redundant dual-arm robotic synchronized cooperative manipulation

Zhenyu Lu, Ning Wang*, Donghao Shi

Abstract

Dual-arm robot manipulation is applicable to many domains, such as industrial, medical, and home service scenes. Learning from demonstrations (LfD) is a highly effective paradigm for robotic learning, where a robot learns from human actions directly and can be used autonomously for new tasks, avoiding the complicated analytical calculation for motion programming. However, the learned skills are not easy to generalize to new cases where special constraints such as varying relative distance limitation of robotic end effectors for human-like cooperative manipulations exist. In this paper, we propose a dynamic movement primitives (DMPs) based skills learning framework for redundant dual-arm robots. The method, with a coupling acceleration term to the DMPs function, is inspired by the transient performance control of Barrier Lyapunov Functions (BLFs). The additional coupling acceleration term is calculated based on the constant joint distance and varying relative distance limitations of end effectors for object approaching actions. In addition, we integrate the generated actions in joint space and the solution for a redundant dual-arm robot to complete a human-like manipulation. Simulations undertaken in Matlab and Gazebo environments certify the effectiveness of the proposed method.

Introduction

With the development of artificial intelligence (AI) and modern engineering technology, robots are widely used in the industry and military domains [1], [2] to complete dexterous manipulation such as grasping and holding [3] objects etc. Compared with single-arm robot, the dual-arm robot can complete more complex tasks and take heavier objects through the cooperative actions of two robot arms [4], which draws a great of attention in academic and industrial areas [5]-[7].

Different from traditional analytical methods spending a lot of time for system modelling, trajectory planning, and force control, learning from demonstration (LfD) is a new paradigm that robots acquire new skills by learning to imitate an expert that is easy to be extended and adapted to novel situations and draws more and more attention in recent years [8]-[11]. Some researchers used a dual-arm robot for robotic skill learning and training which benefits

the isomorphic structure of the two arms: the operator use one arm to record robotic motions and the other robot arm is used as an actuator [12]-[15]. There are a great series of robotic skill learning methods, such as Hidden Markov model (HMM), Gaussian mixture model (GMM) and Gaussian mixture regression (GMR), and Dynamic movement primitives (DMPs) . Compared with HMM and GMM-GMR, DMPs are easier to be explained and have linearity in the parameters of expressions with robustness, and continuity.

Due to the advantages, DMPs were widely used for dual-arm robot skill learning. Kulvicius et al. proposed sensory feedback together with a predictive learning mechanism that allows tightly coupled dual-agent systems to learn an adaptive, sensor-driven interaction based on DMPs [16]. Gams et al. reckoned that the original DMPs function should be modified by adding not only acceleration term but also velocity term, to get a smoother

interaction. So they proposed the coupling of originally independent robotic trajectories by expanding framework of DMPs, which enables the bimanual execution tightly coupled for cooperative tasks [17]. Zhao et al. presented a reinforcement learning (RL) algorithm called the policy improvement with path integrals for sequences of DMPs (SDMPs) to learn and adjust recorded trajectories of dual-arm robot cooperative manipulation [18]. Colome et al. studied simultaneously learning a DMP-characterized robot motion and the joint couplings through linear dimensionality reduction (DR), which provides valuable qualitative information leading to a reduced and intuitive algebraic description of such motion [19]. Lee et al. integrated a similar method for the cooperative aerial transportation with the random tree star (RRT*) to enable cooperative aerial manipulators to carry a common object and keep reducing the interaction force between multiple robots while avoiding an obstacle in the unstructured environment [20].

Seen from the above-mentioned methods, it is not hard to notice that the original DMPs function is modified by adding coupling terms calculated by the relative distance or force errors to change the path and ensure the relative distance tracking errors converge to 0. But, the dynamic performance of the trajectories such as how to enlarge and reduce relative distance flexibly and avoidance obstacles are not considered for the moving process. If we only use a fixed relative distance limitation for the dual arms, the object approaching skills such as changes of the speed and contact forces will be ignored.

In this paper, we will integrate the control strategy of Barrier Lyapunov Functions (BLFs) and DMPs to enable the generate trajectories to satisfy predesigned transient performance for the relative distance of robot end effects. As the trajectory generalized by DMPs is determined by three variables: the start and end points and sampling time interval, and the measured data with errors and noises will be processed (e.g. aligning, filtering etc.) for skill learning, even for the processed data, the data-driven learned results may be against physical limitations. The proposed method based on the integration of BLFs and DMPs will address this problem. Similar ideas combining control and motion planning methods for manipulation have been explored in previous work [29]-[31].

Additionally, we will combine the generated actions in the joint space with solution of a redundant dual-arm robot

to perform human-like operations. Though a similar study about human-like coordinative learning in the Cartesian and joint space for a redundant dual-arm robot has been studied by Qu et al. [21], we will propose another idea by defining a "swivel angle" and combining the null-space method and the results of DMPs with distance constraints. With the dual-arm demonstration data acquired through a Kinect, an experiment is taken based on Matlab and Gazebo to verify the effectiveness of the proposed method.

The rest of the paper is organized as follows: Section 2 makes a brief introduction about DMPs and the problems of skill learning for dual-arm redundant robots. Section 3 presents the DMPs & BLFs framework and related three calculating modules. Section 4 masks three experiments to certify the effectiveness and application of the proposed method. Finally, in Section 5, the conclusions of this paper are summarized.

Problem description

General DMPs model

DMPs model is firstly proposed in [22] and updated by Ijspeert et al.[23], whose function is expressed as

$$\begin{cases} \tau \dot{v} = \alpha_z \left(\beta_z \left(g - x \right) - v \right) + f \left(s \right) \\ \tau \dot{x} = v \end{cases}, \qquad (1)$$

where $\alpha_z, \beta_z > 0$ are coefficients of a two-order function as the linear part in (1), ensuring the convergence of the generated trajectory to the unique attractor point at $x = g$, $v = 0$. $f(s) = \theta^T \Psi(s)$ is a forcing function and a linear combination of nonlinear radial basis functions, and $\theta = \left[w_1, w_2, ..., w_n \right]^T$, $\Psi(s) = \left[\psi_1, \psi_2, ..., \psi_n \right]^T$ and

$$\psi_k = \frac{\varphi_k(s)s}{\sum_{k=1}^{n} \varphi_i(s)}, \varphi_k(s) = \exp(-h_k(s - c_k)^2), k = 1, 2, ...n,$$

where c_k and $h_k > 0$ are centers and widths of radial basis functions respectively. $\tau > 0$ is a timing parameter adjusting speed before execution of movements and s is a phase variable to achieve dependency of function $f(s)$ out of time. The dynamics of s is expressed by a canonical system

$$\tau \dot{s} = -\gamma s, \qquad \gamma > 0. \qquad (2)$$

Term s has implicit relation with time that can modify the convergence time by changing γ, and θ can be learned by supervised learning algorithms e.g. locally weighted regression (LWR). The purpose of calculating process is to minimize the error function:

$$\min \sum \left(f^{Tar}(s) - f(s) \right), x = g, v = 0, \tau > 0, \quad (3)$$

where $f(s)$ is the forcing function in (1), and $f^{Tar}(s)$ represents the target value of $f(s)$:

$$f^{Tar}(s) = \tau \dot{v} - \alpha_z \left(\beta_z (g - x) - v \right). \quad (4)$$

DMP-based dual-arm robot manipulation

From human demonstrations to the skills learned and generalized by robots, we will solve the following three problems: joint distance restriction, redundant joint resolution and relative distance limitations (Figure 1).
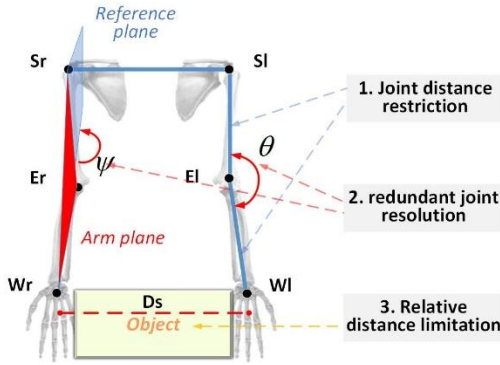

Fig.1 Problems for dual-arm robot manipulation

Joint distance restriction is caused by the bone lengths of the adjacent joints such as the elbow and the wrist or the elbow and the shoulder. It is a constant. Redundant joint resolution is to plan the robot arm joints to achieve human-like motions and the relative distance limitations provide constraints to the robot end effectors.

Meanwhile, measuring noise is acquired together with raw data. If we use the data for skill learning, then learned results will be influenced by noise and measuring errors. Therefore, the signals such as EMG measurements will be processed before learning procedures. Images and videos affected by occlusions are reconstructed, which may cause new uncertainties and errors. An example shown in figure 2 reveals the measuring errors of hand positions are larger than the ones of the shoulders and elbows. Based on the pre-knowledge such as the size of the object, bone length between the elbow and the shoulder, or exact position of the object, we can re-plan joints and positions of the end

effector's to suit new cases. Here, we argue that the known conditions have fixed constraints and propose a new DMP -based framework for dual-arm robot cooperative skill learning with consideration of the above three problems.
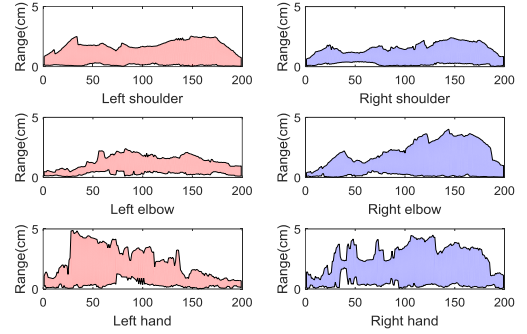
Fig.2 Position measuring errors of shoulders, elbows and hands measured by Kinect

Robotic human-like manipulation has been studied for several decades. By using nullspace method for redundant robotic arm, robots can avoid conflicts with own arms and outer obstacles with multiple joint motion planning results. As the skills for robot ends and joints are generated in both the Cartesian and joint space, the previous researches like [9] and [13] only for the joint or Cartesian space cases are not applicable. We will combine the constrained skills and the nullspace method for skill generalization for redundant dual-arm robots.

BLFs-based improved DMPs for human-like skill learning and redundancy resolution

In this section, we propose three solutions for the above relative distance limitation, joint distance restriction, and redundant joint resolution in three subsections. We firstly specify mathematical symbols in the following paragraphs in Table 1.

Table 1 Mathematical symbols and meanings

Symbols	Meanings
$\bullet_j$, $j = l, r$,	Variables of left and right arms (or robot arms)
$\bullet_{\bar{j}}$, $j = l, r$,	Cooperative role of the j*th* arm
$\bullet^i$, $i = 1, 2, 3$	Variables of the i*th* dimension
$^1\bullet, {}^2\bullet$	Calculations in the first and second round
$\mathbf{x}_{wj}$	Position vector of the hands (wrists)
$\mathbf{x}_{ej}$	Position vector of the elbow
$\mathbf{x}_{sj}$	Position vector of the shoulder
$\bar{\sigma}_j$, $\underline{\sigma}_j$	Upper and lower boundaries of position errors
d_e, d_s	Desired contant distances from the wrist to the elbow, and from the elbow to the shoulder
$d_j(t)$	Desired varying distances of the hands (wrists)

Integrated BLFs and DMPs skills learning for relative distance limitation

Interactive actions of robot end effectors can be seen as a common effect of relative distance and posture changes. For the cooperative actions e.g. folding clothes, grasping and placing objects, relative distance is always changing during the interaction process with environmental objects. Too large relative tracking errors may cause operational failure such as losing control of the object or conflict with the obstacle.

Following robotic desired relative distance $d_j(t)$, we set predesigned error boundaries as $\sigma_{ij}, i=1,2$, which means boundary violation is not allowed throughout the cooperative manipulation process, then the relationship of the jth hand $\mathbf{x}_{wj}$ and its cooperative role $\mathbf{x}_{w\bar{j}}$ is

$$\underline{\sigma}_j \le \left\| \mathbf{x}_{w\bar{j}} - \mathbf{x}_{wj} \right\|_2 - d_j(t) \le \bar{\sigma}_j . \qquad (5)$$

The expression of DMPs model can be rewritten as a strict feedback nonlinear system as

$$\begin{cases} \dot{x} = v/\tau \\ \dot{v} = \alpha_z \left(\beta_z (g-x) - v \right)/\tau + u \\ u = f(s)/\tau \\ y = x \end{cases}, \qquad (6)$$

where u has the same usage with the forcing function $f(s)$ in (6), while in (1) of [28], it represents the input signal to modify output y. Set $\bar{\kappa}_j^i = k_z \left(\bar{\sigma}_j^i + d_j^i \right)$, $\underline{\kappa}_j^i = k_z \left(\underline{\sigma}_j^i + d_j^i \right)$, and $z_1^i = k_z \left(x_j^i - x_{\bar{j}}^i \right)$, and k_z is a positive factor, $\bar{\sigma}_j^i, \underline{\sigma}_j^i$ and d_j^i are decomposed values of the ith dimension, and x_j^i and $x_{\bar{j}}^i$ are general variables that can be instantiated as x_{wj}^i and $x_{w\bar{j}}^i$ in (5). Then (5) can be expressed as $\underline{\kappa}_j^i \le z_1^i \le \bar{\kappa}_j^i$. Inspired by the asymmetric BLFs candidate in [28], we define a new variable as $z_2^i = d_z \left(\alpha_2 - v_j^i \right), d_z > 0$, where α_2 is a function to be designed and build the Lyaponov candidate as

$$V^c = \frac{1}{2} \log \frac{\left(\bar{\kappa}_j^i - \underline{\kappa}_j^i \right)^2}{4\left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} + \frac{1}{2}\left(z_2^i \right)^2 . \qquad (7)$$

The difference of V^c can be calculated as

$$\begin{aligned} \dot{V}^c &= \frac{z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i \right)/2}{\left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} \dot{z}_1^i + z_2^i \dot{z}_2^i \\ &= \frac{z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i \right)/2}{\left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} k_z \tau \left(\dot{x}_{\bar{j}}^i - \dot{x}_j^i \right) + z_2^i d_z \tau \left(\dot{\alpha}_2 - \dot{v}_j^i \right) \end{aligned} \qquad .(8)$$

Theorem 1: Considering the DMPs function described as (6) under the condition of (5), if the initial conditions are such that $\underline{\kappa}_j^i(0) \le z_1^i(0) \le \bar{\kappa}_j^i(0)$ and (9) is satisfied, then the output constraint is never violated and all the closed loop signals are bounded.

$$\begin{cases} \alpha_2 = \dfrac{\tau d_z \dot{x}_{\bar{j}}^i + z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i \right)/2}{d_z} \\ u_j^i = \dot{\alpha}_2 - \left(\alpha_z \left(\beta_z \left(g_j^i - x_j^i \right) - v_j^i \right) \right)/\tau + k_n z_2^i . \\ k_n = \dfrac{k_z}{4\tau d_z^2 \left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} \end{cases} \qquad (9)$$

Proof: Taking the expressions of z_2^i and (6) into (8), we have

$$\begin{aligned} \dot{V}^c &= \frac{z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i \right)/2}{\left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} k_z \tau \left(\dot{x}_{\bar{j}}^i - \alpha_2/\tau + z_2^i/(\tau d_z) \right) + \\ & \quad z_2 d_z \tau \left(\dot{\alpha}_2 - \alpha_z \left(\beta_z \left(g_j^i - x_j^i \right) - v_j^i \right)/\tau - u_j^i \right) \\ &= k_z z_2^i \frac{z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i \right)/2}{d_z \left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} + \\ & \quad \frac{z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i \right)/2}{\left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} k_z \tau \left(\dot{x}_{\bar{j}}^i - \alpha_2/\tau \right) + \\ & \quad z_2^i d_z \tau \left(\dot{\alpha}_2 - \alpha_z \left(\beta_z \left(g_j^i - x_j^i \right) - v_j^i \right)/\tau - u_j^i \right) \\ &= -\frac{k_z \left(z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i - z_2^i \right)/2 \right)^2}{d_z \left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} + \\ & \quad k_z \frac{z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i \right)/2}{\left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} \left[\frac{z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i \right)/2}{d_z} + \tau \dot{x}_{\bar{j}}^i - \alpha_2 \right] + \\ & \quad z_2^i d_z \tau \left(\dot{\alpha}_2 - \alpha_z \left(\beta_z \left(g_j^i - x_j^i \right) - v_j^i \right)/\tau - u_j^i + \right. \\ & \quad \left. \frac{k_z z_2^i}{4\tau d_z^2 \left(\bar{\kappa}_j^i - z_1^i \right)\left(z_1^i - \underline{\kappa}_j^i \right)} \right) \end{aligned} \qquad ,$$

where $k_z > 0$ is a positive number. As $k_z > 0$, and terms $z_1^i - \left(\kappa_{2j}^i + \kappa_{1j}^i \right)/2$ and $z_2^i d_z$ are not always 0, then the sufficient condition for $\dot{V}^c < 0$ is

$$\begin{cases} \tau \dot{x}_j^i - \alpha_2 + \dfrac{z_1^i - \left(\bar{\kappa}_j^i + \underline{\kappa}_j^i\right)/2}{d_z} = 0 \\ \dot{\alpha}_2 - \left(\alpha_z\left(\beta_z\left(g_j^i - x_j^i\right) - v_j^i\right)\right)/\tau - u_j^i + . \\ \dfrac{k_z z_2^i}{4\tau d_z^2 \left(\bar{\kappa}_j^i - z_1^i\right)\left(z_1^i - \underline{\kappa}_j^i\right)} = 0 \end{cases} \quad (10)$$

Then we can get the expressions of α_2, u_j^i and k_n in (9). According to lemma 1 in [28], z_1^i will be kept within the range of $\left(\underline{\kappa}_j^i(t), \bar{\kappa}_j^i(t)\right)$. Following the expression of f^{Tar} in (4), the target value of u_j^i in (6) can be expressed as

$$\begin{aligned} \left(u_j^i\right)^{Tar} &= \dot{\alpha}_2 - \dot{v}_j^i + \left(f_j^i\right)^{Tar}/\tau + k_n z_2^i \\ &= \left(f_j^i\right)^{Tar}/\tau + k_n z_2^i + \dot{z}_2^i/d_z \end{aligned}. \quad (11)$$

Furthermore, we use $f(s)$ to replace $\left(f_j^i\right)^{Tar}$ based on the error function of (3) and get

$$u_j^i = f(s)/\tau + k_n z_2^i + \dot{z}_2^i/d_z = f(s)/\tau + \Delta u_j^i, \quad (12)$$

where $\Delta u_j^i = k_n z_2^i + \dot{z}_2^i/d_z$, which means that Δu_j^i will be calculated by two steps: first, using (3) to get the forcing function $f(s)$; second, calculating z_1^i and z_2^i timely and adding them to (12). Then the output y is determined by the common function of $f(s)$ and the z_2^i generated by BLFs function.

Remark 1: Similar to the improved DMPs for multi-agent formation [16]-[20], by adding a term Δu_j^i, the original path point x (or output y) is modified to fit the constraints in (5). However, the proposed method in (9) is more general compared with the previous special designs only providing limitations for the end effectors or point shape of multi-agent formation. The following subsection will extend this method to the case of joint distance restriction.

Integrated BLFs and DMPs skills learning with joint distance restriction

The challenges for the joint distance restrictions are modifying the distance between the adjacent joints such as the elbow and the shoulder or the elbow and twist. Following the definitions in Table 2, the distance errors ranges (like σ_{ij} for relative distance limitation) are set as σ_e and σ_s. Similar to (5), we can use the following inequalities to reshape the elbow and shoulder positions

as

$$\begin{cases} \left| \left\| \mathbf{x}_{ej} - \mathbf{x}_{wj} \right\|_2 - d_e \right| < \sigma_e \\ \left| \left\| \mathbf{x}_{ej} - \mathbf{x}_{sj} \right\|_2 - d_s \right| < \sigma_s \end{cases}. \quad (13)$$

However, as mentioned in Section 2, if we want to replan positions of the elbow and the shoulder, we should take the real measurements of the elbow and the shoulder as reference, but they are limited by the hands' relative distance conditions. Therefore, we will reshape the elbow and the shoulder distance satisfying both (5) and (13) by calculating a common result for the two conditions.

Additionally, following Fig.2, the measuring errors of the hands are larger than those of the elbows. There are measuring mistakes due to the occupations, then the errors are processed in the following two steps:

1) Satisfy limitations for hands' motions $^1\mathbf{x}_{wj}$ as:

$$^1\mathbf{x}_{wj} := \left| \left\| \mathbf{x}_{ej} - {}^1\mathbf{x}_{wj} \right\|_2 - d_e(t) \right| < \sigma_e, \quad (14)$$

where $^1\mathbf{x}_{wj}$ represents the results for the first round. Eq. (14) is used to filter the data of hands first by the elbow measurements.

2) Synchronized constraints for the hands, elbows and shoulders

$$^2\mathbf{x}_{wj} := \begin{cases} \sigma_{1j} < \left\| {}^2\mathbf{x}_{w\bar{j}} - {}^2\mathbf{x}_{wj} \right\|_2 - d_j(t) < \sigma_{2j} \\ -\sigma_e < \left\| \mathbf{x}_{ej} - {}^2\mathbf{x}_{wj} \right\|_2 - d_e(t) < \sigma_e \end{cases}, \quad (15)$$

$$\mathbf{x}_{ej} := \begin{cases} -\sigma_s < \left\| \mathbf{x}_{ej} - \mathbf{x}_{sj} \right\|_2 - d_s(t) < \sigma_s \\ -\sigma_e < \left\| \mathbf{x}_{ej} - \mathbf{x}_{wj} \right\|_2 - d_e(t) < \sigma_e \end{cases}, \quad (16)$$

$$\mathbf{x}_{sj} := \left| \left\| \mathbf{x}_{ej} - \mathbf{x}_{sj} \right\|_2 - d_s(t) \right| < \sigma_s, \quad (17)$$

where $^2\mathbf{x}_{wj}$ represents the results for the second round and Eqs. (15) to (17) consider both limitations of the relative distance and joint distance to rebuild trajectory.

Remark 2: The calculating basis for (15) and (16) is to find a common conditional result satisfying both two inequalities. But, sometimes there is no overlap area for the two conditions. Here, we reckon that the priority of joint distance restriction is higher than that of relative

distance, thus the inequality of joint distance restriction will be first considered and then σ_j will be modified to adapt to the condition $\kappa_{1j}^i(k) \le z_1^i(k) \le \kappa_{2j}^i(k)$ for the kth calculation.

Remark 3: Similar to the methods in [16]- [20], the skill learning process will be handled first, and then by adding new term Δu_j^i, the trajectory will be generalized. Here, we will learn and generalize angle skills for the elbow and the shoulder first without using any limitations. Then the integrated BLFs and DMPs skills learning are used to modify the generalized trajectory to suit inequalities (15) to (17). The improved DMPs hold the properties of normal DMPs and dynamic performance determined by the factors: start and end points, sampling interval.

Based on the BLFs and DMPs integrated skill learning method, we present the general calculation procedure as follows:
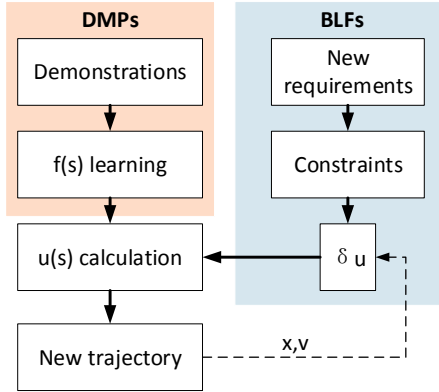
Fig.3 Procedure of the integrated skills learning method

In Fig.3, $f(s)$ in (12) of DMPs and constraints (like (5), (13)-(17)) of BLFs are firstly designed separately. After initialing Δu_j^i in (12), u_j^i will be calculated to generate new trajectories, and the new position and velocity information will be used to update Δu_j^i for the next circulation till the destination.

Robotic human-like redundancy resolution

For replanning robotic actions based on the learned skill from demonstrations, some previous researches proposed human-like swivel motion by using its redundant degrees of the manipulator [24] [25]. After rebuilding positions of the hand, elbow and shoulder, we can generalize the joints and end effectors of the redundant robots. Following the

depictions in [24], arm plane is the plane built with three joint points of the wrist, elbow and shoulder and reference plane is set as the vertical plane to the human body. We set the swivel angle $\psi_j, j = l, r$ as the angle of arm plane and reference plane, shown in Fig.4.
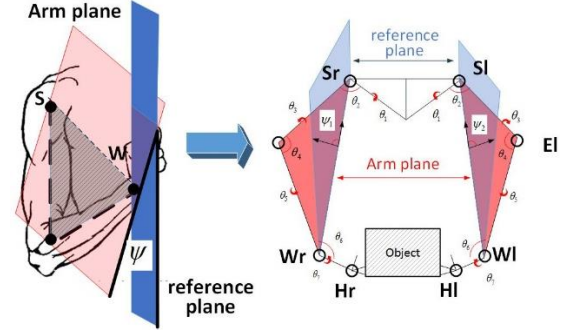
Fig.4 Definition of swivel angle of dual arms

Set the joint velocities of 7-Dof redundant robot arm as $\dot{\mathbf{q}}_j \in R^{7 \times 1}, j = l, f$, and set the generalized end effectors' positions as $^2\overline{\mathbf{x}}_{wj} \in R^{3 \times 1}, j = l, f$, and the distance between the hand and the wrist are ignored for the two-arm holding actions as $\overline{\mathbf{x}}_{hj} \approx^2 \overline{\mathbf{x}}_{wj}$ (in fact due to the object occlusion, all hand positions cannot be well got). Then by extending $\overline{\mathbf{x}}_{hj}$ with gestures of the end effectors to $\mathbf{X}_{hj}$, we have $\dot{\mathbf{X}}_{hj} = \mathbf{J}_j \dot{\mathbf{q}}_j$, and $\mathbf{J}_j \in R^{6 \times 7}$ represents a Jacobian matrix. Following null-space projection, we calculate redundancy solution of a redundant robot arm as

$$\dot{\mathbf{q}}_j = \mathbf{J}_j^+ \dot{\mathbf{X}}_{hj} + \left(\mathbf{I} - \mathbf{J}_j^+ \mathbf{J}_j \right) \mathbf{J}_{Ej}^+ \psi_j \mathbf{U}_{\psi_j} , \qquad (18)$$

where $\mathbf{J}_{Ej} \in R^{3 \times 7}$ is the Jacobian matrix from the elbow of the robot to the robot base as well as the mapping between the swivel angle and joint velocities [24]. $\mathbf{U}_{\psi_j}$ is defined as the velocity director of swivel motion:

$$\mathbf{U}_{\psi_j} = \frac{\overrightarrow{SE} \times \overrightarrow{EW}}{\left\| \overrightarrow{SE} \times \overrightarrow{EW} \right\|} , \qquad (19)$$

where $\overrightarrow{SE} = \mathbf{x}_{sj} - \mathbf{x}_{ej}$ represents a vector from the shoulder to the elbow and $\overrightarrow{EW} = \mathbf{x}_{ej} -^2 \overline{\mathbf{x}}_{wj}$ represents a vector from the elbow to the wrist for the generalized human demonstrations. Then the vector will be used for the robot joint planning and calculation of θ shown in Fig. 1.

Experiment

In the experiment, we will achieve demonstration data by Kinect and verify the manipulation effect through the virtual model in Gazebo. The skeleton data of a task of holding and placing a box is recorded for demonstrations. The positions of joints (the elbow, shoulder and wrist) are used for skill training in Cartesian space as Fig.5 shown.

The upper figures (a) to (g) are the illustrations of human demonstrations and the lower figures (a) to (g) show the human skeleton data acquired by Kinect. But, we can find the acquired sampling data has errors such as the left leg motions in (b) and (d) and the hand motions in (e), which are caused by obstacle occupation.
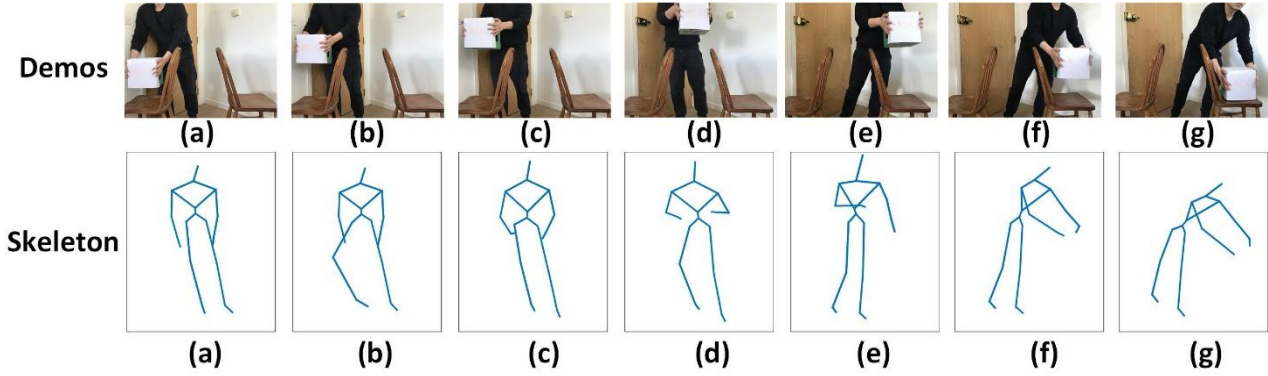
Fig.5 Human demonstrations and skeleton data
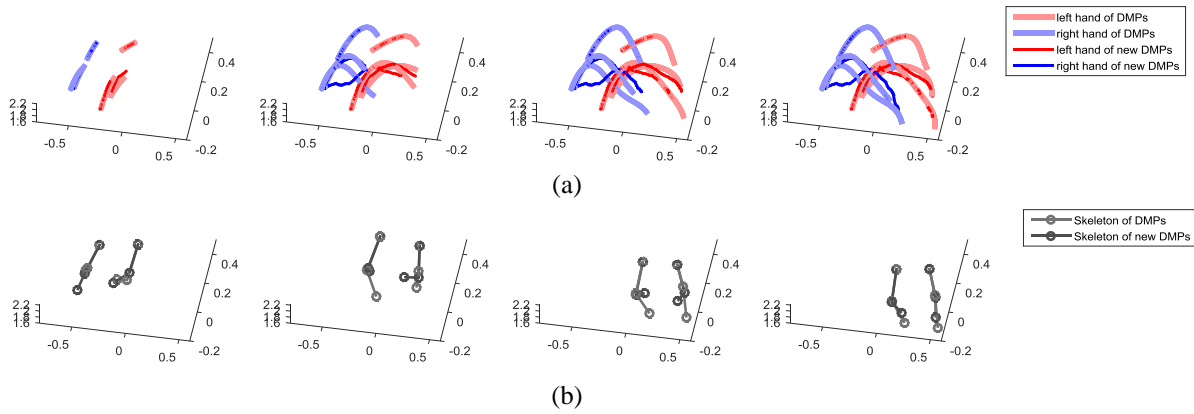
Fig.6 Desired and real shoulder, elbow and wrist movements (a) changes of trajectories (b) changes of skeletons
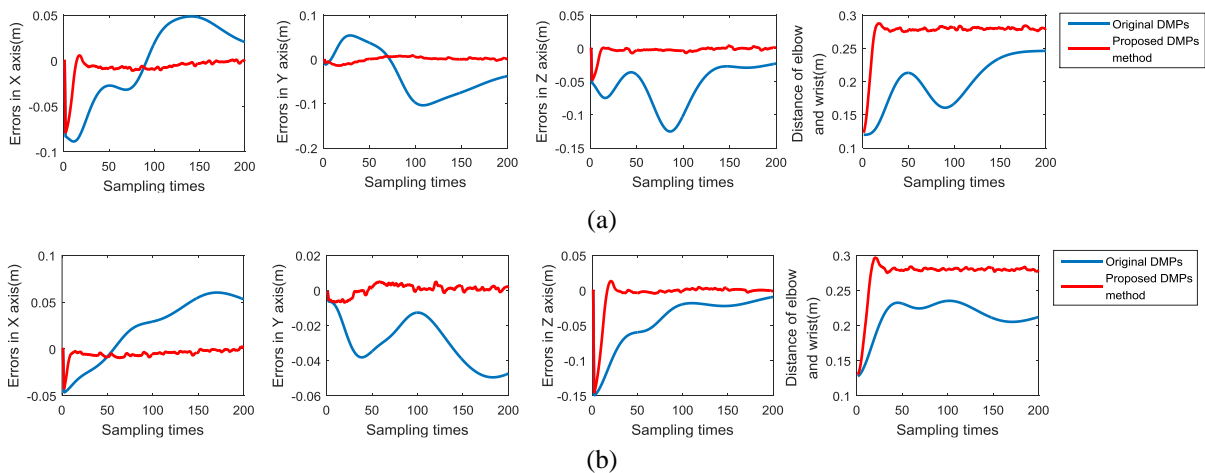

Fig.7 Errors of the desired and real the elbow and wrist positions (a) results of the left arm (b) results of the right arm

Figs. 6 and 7 are the results based on the initialization of hand's positions using (14). Fig. 6 presents changes of the trajectories and skeletons of the original and modified DMPs. Fig. 7 presents tracking errors to the predesigned

referring trajectory fitting the constrains in (14) to (17). Combining with Fig. 7, we can see that the trajectories of both the left and right arms are changed to avoid violation (14) to (17). Seen from the blue lines (results of original

DMPs) in Fig. 7, the amplitudes of the errors in each axis vary from -0.1 to about 0.05, causing the final distance of the elbow and the wrist changed within a large range of $(0.12, 0.25)$, which seriously against to the fact.

As the distance is measured between the elbow and the hand contact point (palm) to about 0.28m, namely $d_e(\infty) = 0.28$. Setting $\sigma_e = 0.01$, then the generalized results show that the errors to desired position decrease to the range of $(-0.01, 0.01)$ after the first few steps and the distance converges to the value around the desired value of 0.28 for both hands. The initial distance errors are large. It need about 15 iterations to guide the large distance errors to decrease to the desired conditions in (14), which is further processed in step 2 by using (15) to (17). Fig.8 (a/b) presents the original distance of the joints (the elbow and wrist, and the elbow and shoulder). It shows that the joint distance varies within a range of $(0.13, 0.32)$ for both the elbow and the shoulder, and the elbow and the wrist of both arms. Using the proposed two-step method, we can get the fixed distance for the joint links and the varying relative distance for the dual hands' manipulation, both of which converge to a stable interval (Fig. 9(a/b) and Fig. 10).

(a)   (b)
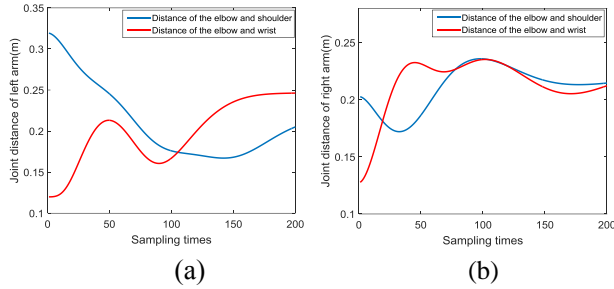
Fig.8 Joint distances of the right and left arms (a) Results of the left arm (b) Results of the right arm

(a)   (b)
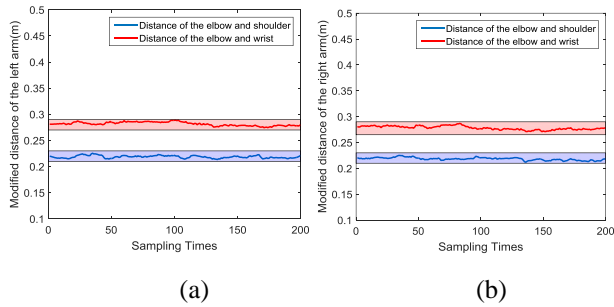
Fig.9 Modified joint distances of the right and left arms (a) Modified distance of the joints of the left arm (b) Modified joint of the joints of the right arm
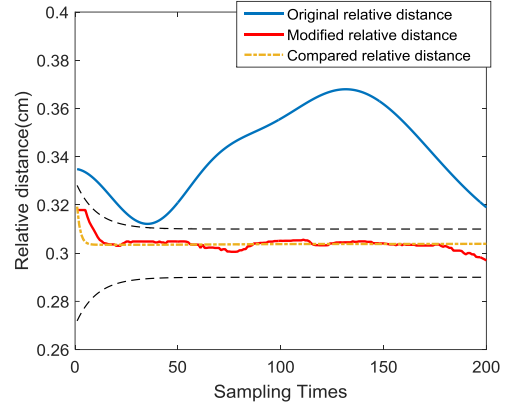
Fig.10 Original, modified and compared relative distance of the wrists

As we set the desired joint distance as $d_e \equiv 0.28$ and $d_s \equiv 0.22$ by actual measurements, and the error range as 0.01, the results presented in Fig.9 verify the effectiveness of the proposed method even for the large initial errors. The performance functions for the upper and lower boundaries are set as

$$\begin{cases} \sigma_{2j} = 0.02\,\mathrm{e}^{-k/10} + 0.01 \\ \sigma_{1j} = -0.02\,\mathrm{e}^{-k/10} - 0.01 \end{cases}, \qquad (20)$$

where k represents the sampling times. Additionally, we compare the results with the method in [17] and set the desired distance as $d_j(t) \equiv 0.3$. Fig.10 shows the measured relative distances of human two hands (blue lines). The black dash lines present the boundaries of relative distance that decrease from 0.32 to the interval $(0.29, 0.31)$. The method in [17] enables the relative distance to quickly decrease and keep the value around 0.305, but the relative distance cannot change along with dynamic performance like (20).

The last step is extending the learned skills to new task. We select two Franka robots as manipulators to hold and place an object with width of 0.3m, the base of which located at $[0.2, 0, 0.4]$ and $[-0.2, 0, 0.4]$. The simulations are taken in Matlab and then the results are transferred and certified in the Gazebo environment. Here, we select the 4th joint as the elbow joint to solve the redundant solution. Fig. 11 presents the simulation process that Two Franka robots are controlled to move the object to follow the trajectories generated in Matlab

under the PD control. During the simulation, we set the object and robot end effectors have a certain degree of deformation to counteract the influence of the relative distance tracking errors.

Conclusion

In this paper, we proposed a new DMP-based skill learning and generalization framework for the dual-arm redundant cooperative manipulation. The framework has three functions: skill learning and generalization for the relative distance limitation, trajectory replanning for the joint distance restriction, and redundant solution for multi-Dof robot based on the generalized dual-arm skills. The two former skill/trajectory learning and generalization methods are studied based on the integration of BLFs and DMPs methods. Using the demonstration data acquired by Kinect, the effectiveness of the proposed framework is verified by a task of holding and placing an object based on the simulations in Matlab and Gazebo. Each technical method is proved and explained by the simulation results. The future work is hopeful to be taken on the real robotic system to complete skill learning autonomously.
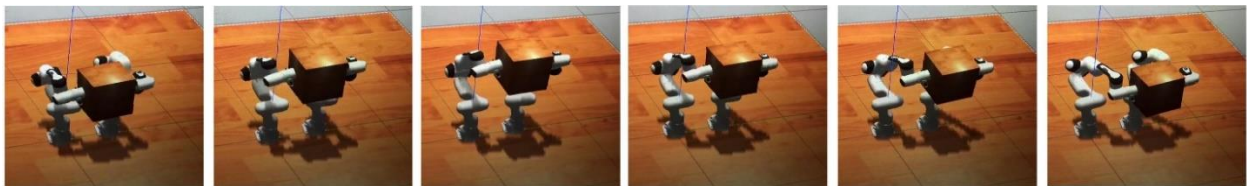
Fig.11 Simulated certification taken based on Matlab and Gazebot

Reference

[1]. Yu Y, Xu Y, Wang F, Li W, Mai X, Wu H (2020). Adsorption control of a pipeline robot based on improved PSO algorithm. Complex & Intelligent Systems, 31:1-7.

[2]. Arokiasami WA, Vadakkepat P, Tan KC, Srinivasan D. (2016) Interoperable multi-agent framework for unmanned aerial/ground vehicles: towards robot autonomy. Complex & Intelligent Systems. 2(1):45-59.

[3]. Li Q, Haschke R, Bolder B, Ritter H (2012) Grasp point optimization by online exploration of unknown object surface. In2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012) pp. 417-422.

[4]. Lu Z, Huang P, Liu Z. (2017) Predictive approach for sensorless bimanual teleoperation under random time delays with adaptive fuzzy control. IEEE Transactions on Industrial Electronics. 65(3): 2439-48.

[5]. Yang C, Jiang Y, Li Z, He W, Su CY. (2016) Neural control of bimanual robots with guaranteed global stability and motion precision. IEEE Transactions on Industrial Informatics. 13(3):1162-71.

[6]. Garcia N, Suárez R, Rosell J. (2017) Task-dependent synergies for motion planning of an anthropomorph-ic dual-arm system. IEEE Transactions on Robotics. 33(3):756-64.

[7]. Yang C, Jiang Y, Na J, Li Z, Cheng L, Su CY. (2018) Finite-time convergence adaptive fuzzy control for dual-arm robot with unknown kinematics and dynamics. IEEE Transactions on Fuzzy Systems, 27(3):574-88.

[8]. Ravichandar H, Polydoros AS, Chernova S, Billard A (2020) Recent advances in robot learning from demonstration. Annual Review of Control, Robotics, and Autonomous Systems, 3.

[9]. Yang C, Zeng C, Fang C, He W, Li Z (2018) A dmps-based framework for robot learning and generalization of humanlike variable impedance skills. IEEE/ASME Transactions on Mechatronics, 23(3):1193-203.

[10]. Yang C, Chen C, He W, Cui R, Li Z (2018) Robot learning system based on adaptive neural control and dynamic movement primitives. IEEE transactions on neural networks and learning systems. 30(3):777-87.

[11]. Qiao, H, Li, C, Yin, P, Wu, W and Liu, Z Y (2016), "Human-inspired motion model of upper-limb with fast response and learning ability–a promising direction for robot system and control, Assembly Automation, 36(1), pp. 97–107.

[12]. Yang C, Chen C, Wang N, Ju Z, Fu J, Wang M (2018) Biologically inspired motion modeling and neural control for robot learning from demonstrations. IEEE Transactions on Cognitive and Developmental

Systems. 11(2):281-91.

[13]. Yang C, Zeng C, Cong Y, Wang N, Wang M (2018) A learning framework of adaptive manipulative skills from human to robot. IEEE Transactions on Industrial Informatics.15(2):1153-61.

[14]. Yang C, Zeng C, Liang P, Li Z, Li R, Su CY (2017) Interface design of a physical human–robot interaction system for human impedance adaptive skill transfer. IEEE Transactions on Automation Science and Engineering,15(1):329-40.

[15]. Yang C, Zeng C, Fang C, He W, Li Z. (2018) A dmps -based framework for robot learning and generalization of humanlike variable impedance skills. IEEE/ASME Transactions on Mechatronics. 23(3):1193-203.

[16]. Kulvicius T, Biehl M, Aein MJ, Tamosiunaite M, Wörgötter F (2013) Interaction learning for dynamic movement primitives used in cooperative robotic tasks. Robotics and Autonomous Systems. 61(12):1450-9.

[17]. Gams A, Nemec B, Ijspeert AJ, Ude A (2014) Coupling movement primitives: Interaction with the environment and bimanual tasks. IEEE Transactions on Robotics. 30(4):816-30.

[18]. Zhao T, Deng M, Li Z, Hu Y (2018) Cooperative manipulation for a mobile dual-arm robot using sequences of dynamic movement primitives. IEEE Transactions on Cognitive and Developmental Systems. 12(1), 18 – 29.

[19]. Colomé A, Torras C (2018) Dimensionality reduction for dynamic movement primitives and application to bimanual manipulation of clothes. IEEE Transactions on Robotics. 34(3):602-15.

[20]. Lee H, Kim H, Kim HJ. (2016) Planning and control for collision-free cooperative aerial transportation. IEEE Transactions on Automation Science and Engineering. 15(1):189-201.

[21]. Qu J, Zhang F, Wang Y, Fu Y. (2019) Human-like coordination motion learning for a redundant dual-arm robot. Robotics and Computer-Integrated Manufacturing, 57:379-90.

[22]. Ijspeert AJ, Nakanishi J, Schaal S (2001) Trajectory formation for imitation with nonlinear dynamical systems. Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the

Next Millennium 2: 752-757.

[23]. Ijspeert AJ, Nakanishi J, Hoffmann H, Pastor P, Schaal S (2013) Dynamical movement primitives: learning attractor models for motor behaviors. Neural computation. 5(2):328-73.

[24]. Su H, Qi W, Yang C, Aliverti A, Ferrigno G, De Momi E (2019) Deep neural network approach in human-like redundancy optimization for anthropomorphic manipulators. IEEE Access. 7: 124207-124216.

[25]. Kim S, Kim C, Park JH (2006) Human-like arm motion generation for humanoid robots using motion capture database. In 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (pp. 3486-3491

[26]. Lamperti C, Zanchettin AM, Rocco P (2015) A redundancy resolution method for an anthropomorphic dual-arm manipulator based on a musculoskeletal criterion. In2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) pp. 1846-1851.

[27]. Luo J, Yang C, Li Q, Wang M (2019) A task learning mechanism for the telerobots. International Journal of Humanoid Robotics. 9;16(02): 1950009.

[28]. Tee KP, Ge SS, Tay EH (2009) Barrier Lyapunov functions for the control of output-constrained nonlinear systems. Automatica, 45(4):918-27.

[29]. Peng G, Yang C, He W and Chen C. L. P., (2020) Force Sensorless Admittance Control with Neural Learning for Robots with Actuator Saturation, IEEE Transactions on Industrial Electronics, 67(4): 3138-3148,

[30]. Huang H, Zhang T, Yang C, Chen C. L.P., (2020) Motor Learning and Generalization Using Broad Learning Adaptive Neural Control, IEEE Transactions on Industrial Electronics, 67(10): 8608 -8617.

[31]. Huang H, Yang C, Chen C. L. P., (2021), Optimal Robot-Environment Interaction Under Broad Fuzzy Neural Adaptive Control, IEEE Transactions on Cybernetics. DOI:10.1109/TCYB.2020.2998984