Novel Learning From Demonstration Approach for Repetitive Teleoperation Tasks

Affan Pervez[1], Arslan Ali[2], Jee-Hwan Ryu[2] and Dongheui Lee[1]

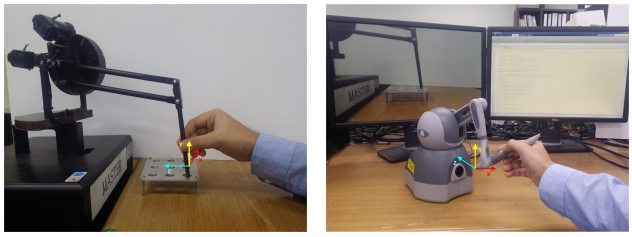
***Abstract*— While teleoperation provides a possibility for a robot to operate at extreme conditions instead of a human, teleoperating a robot still demands a heavy mental workload from a human operator. Learning from demonstrations can reduce the human operator's burden by learning repetitive teleoperation tasks. However, one of challenging issues is that demonstrations via teleoperation are less consistent compared to other modalities of human demonstrations. In order to solve this problem, we propose a learning scheme based on Dynamic Movement Primitives (DMPs) which can handle less consistent, asynchronized and incomplete demonstrations. In particular we proposed a new Expectation Maximization (EM) algorithm which can synchronize and encode demonstrations with temporal and spatial variances, different initial and final conditions and partial executions. The proposed algorithm is tested and validated with three different experiments of a peg-in-hole task conducted on 3-Degree of freedom (DOF) master-slave teleoperation system.**

I. INTRODUCTION

Imitating a task through observations is inherently easy for humans, but can surprisingly be challenging for robots. Generally a robot has to be pre-programmed for performing different tasks. A slight change in a task or the environment can require re-programming of the robot, which can be a tedious and a time consuming process [1]. Learning from Demonstrations (LfD), also known as imitation learning, provides an intuitive way to readily teach new skills to the robots [1], [2]. In LfD, instead of programming, a task is learned from human demonstrations.

The most common teaching modalities for providing demonstrations in LfD include (i) vision based human motion tracking and (ii) kinesthetic teaching. In vision based motion tracking, kinematics information is extracted by a motion tracking system, which is then mapped to a physical model that resembles the human, for instance a humanoid robot [3] [4]. On the other hand, kinesthetic teaching involves providing physical action as a learning paradigm where human moves robots body physically to demonstrate the skill, ruling out the correspondence problem [5] [6]. It is accepted as a more intuitive way of teaching simple tasks.

However, learning paradigm based on kinesthetic teaching is not feasible for certain applications where human demonstrator cannot be co-located with a robot, for instance nuclear waste handling, underwater manipulation and space applications. In those cases, teleoperation often can be considered as the only way for operating a robot located at such inaccessible places [7], [8]. There were several studies which use LfD with teleoperation. One of the earliest work was presented in 1993 which illustrates the skills learning using Hidden Markov Models (HMMs) [9]. In [10], NASA space humanoid robot learned how to perform low level task "reaching and grasping", whereas high-level commands were given by the operator. Rozo et al. [11], [12] used Gaussian Mixture Model (GMM) and Gaussian Mixture Regression (GMR) for teaching a rigid-container emptying skills to a robot via a haptic interface. Other work considered encoding of force/torque signal with large time discrepancies by using a GMM with temporal information encapsulated by a HMM [13].



(a) (b)

Fig. 1: (a) Kinesthetic teaching by directly holding the robot, (b) Teleoperation by using 3-DOF haptic device

Even though there have been several prior researches on learning via teleoperation, but most of them were focused on applying Lfd approaches developed for kinesthetic teaching, undermining the need to address the problems encountered during teleoperation. To the best of our knowledge, there has been no research on developing learning from demonstration approach which can handle inconsistent demonstration with large spatial and temporal variations, demonstration with different starting/ending phases and partial demonstrations of the task. In this paper, we propose a novel approach for handling the unique features encountered during teleoperation. Our approach is based on Dynamic Movement Primitives (DMP) for learning the demonstrations provided by an operator during teleoperation. As opposed to other conventional approaches which use Dynamic Time Wrapping (DTW) as a preprocessing step, our approach aligns and encode the motion data simultaneously and incorporates the temporal and spatial variance presented by using an EM algorithm [14]. In our approach separate DMPs are learned

*This work is partially supported by the German Research Foundation (DFG) and the Industrial Strategic Technology Development Program (10069072) funded by the MOTIE, Korea

[1] Affan Pervez and Dongheui Lee are with Department of Electrical and Computer Engineering, Technical University of Munich, Germany. Email: {affan.pervez, dhlee}@tum.de

[2]Arslan Ali and Jee-Hwan Ryu are with Department of Mechanical Engineering, Korea University of Technology and Education, Korea. Email: {arslanali, jhryu}@koreatech.ac.kr

for each degree of freedom, while the forcing terms of the DMPs are encoded with a GMM.

II. MOTIVATION

Comparison studies between interaction by kinesthetic teaching and teleoperation have shown that users found kinesthetic teaching easier to use, faster for providing the demonstrations, and as a preferred way of demonstrating a task [15], [16]. These studies also concluded that kinesthetic teaching leads to more successful demonstrations in a shorter time duration. By using the setup shown in the Figure 1, we asked an operator to demonstrate four cycles of the peg-in-hole task. Figures 2a and 2b illustrate the demonstrations of the peg-in-hole task recorded via kinesthetic teaching and teleoperation. It can be seen in Figures 2c and 2d that the demonstrations obtained using kinesthetic teaching are much more consistent as compared to the demonstrations recorded via teleoperation. Also trajectories obtained via teleoperation have high spatial-temporal variations.

Apart from the higher level of spatial-temporal variations, demonstrations via teleoperation can pose additional challenges. For instance due to the symmetric nature of the peg-in-hole task, a change in camera position can cause a user to lose track of a single starting region while providing the demonstrations. This can result in different initial and final configurations for different demonstrations. Also an operator might prematurely terminate his motion due to communication issues or a sensor failure, yielding an incomplete demonstration. All these issues make it challenging to apply existing LfD techniques directly for teleoperation.

Most of the existing LfD literature for teleoperation focuses on DTW [17] based motions alignment before encoding the demonstrations. DTW is an algorithm which measure the similarities between two temporal signals executed at different speeds. DTW does not provide a good solution to the aforementioned problems because it assumes same initial and final state of a signal, executed at different speeds. However, DTW cannot be used if this constraint is not fulfilled. Another limitation of GMM/HMM based LfD approaches for teleoperation is that unlike nonlinear dynamical system based approaches, such as DMP [18], the HMM/GMM based encoding with GMR based trajectory generation does not consider the current position of the end-effector when generating the trajectories [19], [20]. The above mentioned reasons raise a scalability issue, when applying LfD approaches developed for kinesthetic teaching to teleoperation. Hence there is a need to develop LfD approaches for handling inconsistencies encountered during demonstrations via teleoperation.

III. TELEOPERATION SKILLS LEARNING

A. Dynamic Movement Primitive

DMP is a way to learn motor actions. It can encode discrete as well as rhythmic movements. A separate DMP is learned for each considered degree of freedom. In DMP framework a canonical system acts as a clock. For synchronized motion of multiple DOFs, each DMP is driven

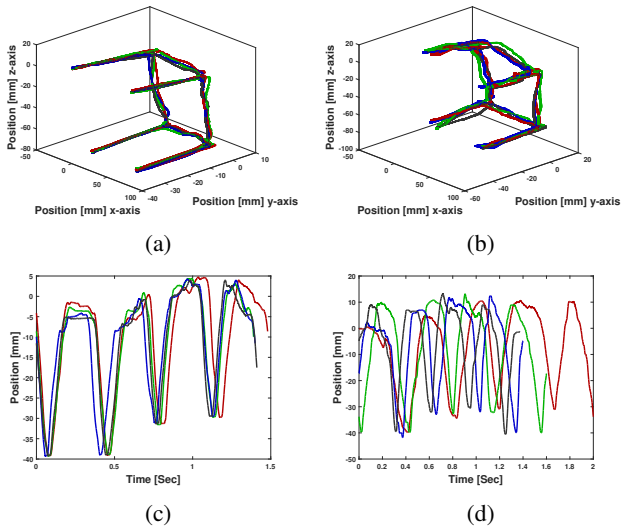


Fig. 2: Four demonstrations recorded via kinesthetic teaching and teleoperation for the peg-in-hole task (a) Recorded trajectories via kinesthetic teaching (b) Recorded trajectories via teleoperation (c) Motion in y-axis using kinesthetic teaching (d) Motion in y-axis using teleoperation.

by a common clock signal. The canonical system for a discrete DMP is $\dot{s} = -\tau\alpha_s s$, where the parameter s is initialized to one and it monotonically decays to zero, τ is the temporal scaling factor while α_s determines the duration of the movement. For a rhythmic DMP, $\dot{s} = \tau\omega$ with the parameter s initialized to zero and increases to 2π at the end of a cycle, and ω determines the phase rate of change. The canonical system drives the second order transformed system:

$$\begin{aligned} \dot{v} &= \tau\alpha_x(\beta_x(g-x)-v) + \tau a\mathcal{F}(s) \\ \dot{x} &= \tau v \end{aligned}$$

where g is either a goal position for discrete DMPs or a mean position for rhythmic DMPs, a is an amplitude modifier term which is usually set as $g - x_0$ with x_0 being the starting position, while the parameters α_x and β_x are set such that the second order system is critically damped. The learning of forcing term $\mathcal{F}(s)$ allows arbitrarily complex movements.

B. DMP learning with GMM/GMR

In the original DMP, locally weighted regression technique is applied in order to learn the nonlinear forcing term $\mathcal{F}(s)$. In this work, we encode the forcing term with GMM and after learning the forcing term will be synthesized using GMR. When human demonstrations are collected, each demonstration trajectory is linearly re-sampled to have n number of samples. For each trajectory, the required forcing terms are calculated by rearranging the terms in the DMP equation:
$\mathcal{F}(s) = \dot{v}/(\tau a) - (\alpha_x(\beta_x(g-x)-v))/a$. Then, we have a sequence of a pair, consisting of position x and forcing terms $\mathcal{F}(s)$. Assuming that we know the corresponding phase variable s, now we encode the joint distribution of these

variables by using a GMM. During the reproduction phase, GMR is used for predicting the forcing term for a given phase signal s, which is plugged into the DMP equation to get the acceleration command. A separate controller is used for executing the motion.

C. EM algorithm for learning from asynchronous data

As discussed earlier, the demonstrations can have temporal variations and thus we cannot attach a phase signal with each trajectory. In order to handle such inconsistent and asynchronized teleoperated demonstration trajectories, we first separate the demonstrations into two parts: one reference trajectory and the rest. The reference trajectory should be a full execution of the motion and it is recommend to be the minimum jerk trajectory.

Assume that there are k demonstration trajectories and the first demonstration is the reference trajectory. From the demonstrations, we have two data sets. The first data set contains only the reference trajectory, its forcing terms and the concatenated phase signals. We call it a complete data ($\mathbf{X}^{Com}$). The second data set contains the remaining trajectories and their forcing terms. However their phase variables s are unknown. Thus, it is termed as incomplete data ($\mathbf{X}^{InCom}$). The missing phase signals in the second data set will be estimated by synchronizing this data set with the first data set iteratively during EM. We will use the notation $\mathbf{x}_i^{Com}$ and $\mathbf{x}_i^{InCom}$ to denote i^{th} column in complete ($\mathbf{X}^{Com}$) and incomplete data sets ($\mathbf{X}^{InCom}$) respectively.

$$\mathbf{X}^{Com} = \mathbf{X}_1 = \begin{bmatrix} \mathcal{F}_1(s_0) & x_{1,0} & s_0 \\ \vdots & \vdots & \vdots \\ \mathcal{F}_1(s_n) & x_{1,n} & s_n \end{bmatrix}^\top \quad \mathbf{X}^{InCom} = \begin{bmatrix} \mathbf{X}_2 \\ \vdots \\ \mathbf{X}_k \end{bmatrix}^\top$$

Now we fit a GMM to the data sets by using an EM algorithm. The GMM parameters and the missing phase signals in each trajectory should be estimated. These parameters are initialized and are then iteratively updated during EM. The values of phase signals are initialized as a linearly increasing value from 0 to 2π in each trajectory for a rhythmic DMP and an exponentially decreasing value from 1 to 0 in each trajectory for a discrete DMP. Now we have n data-points in complete data set and $n_2 = n * (k-1)$ data-points in incomplete data set.

1) E-step: First we separate the variables into two types, missing ($miss$) denoting the phase variable and observable (Obs) denoting all variables other than the phase variable.

$$\boldsymbol{\mu}_m = \begin{bmatrix} \boldsymbol{\mu}_m^{Obs} \\ \boldsymbol{\mu}_m^{miss} \end{bmatrix}, \boldsymbol{\Sigma}_m = \begin{bmatrix} \boldsymbol{\Sigma}_m^{Obs} & \boldsymbol{\Sigma}_m^{Obs,miss} \\ \boldsymbol{\Sigma}_m^{miss,Obs} & \boldsymbol{\Sigma}_m^{miss} \end{bmatrix}$$

In case of a rhythmic DMP, first map all the phase variable within the interval $[\mu_m^{miss} - \pi, \mu_m^{miss} + \pi]$, for calculating the valid probabilities for the m^{th} GMM component.

$$\begin{aligned} p_{i,m} &= \pi_m \mathcal{N}(\mathbf{x}_i^{Com}; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \\ q_{j,m} &= \pi_m \mathcal{N}(\mathbf{x}_j^{InCom}; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \end{aligned}$$

where the initialized (or updated during M-step) phase variables are used for probabilities calculation in incomplete data set. The responsibility terms for the i^{th} and j^{th} data points in complete and incomplete data sets are calculated as:

$$E[z_{i,m}|\mathbf{x}_i^{Com}, \boldsymbol{\theta}_t] = h_{i,m} = \frac{p_{i,m}}{\sum_{l=1}^{k} p_{i,l}} \quad , \quad h_m = \sum_{l=1}^{n} h_{l,m}$$

$$E[z_{j,m}|\mathbf{x}_j^{InCom}, \boldsymbol{\theta}_t] = r_{j,m} = \frac{q_{j,m}}{\sum_{l=1}^{k} q_{j,l}} \quad , \quad r_m = \sum_{l=1}^{n_2} r_{l,m}$$

In incomplete data set, the prediction of the j^{th} missing value with respect to the m^{th} GMM component is done as: $\hat{x}_{j,m}^{miss} = \mu_m^{miss} + \boldsymbol{\Sigma}_m^{miss,Obs}(\boldsymbol{\Sigma}_m^{Obs})^{-1}(\mathbf{x}_j^{InCom,Obs} - \boldsymbol{\mu}_m^{Obs})$ With this predicted value, two additional expectations are calculated for the incomplete data set:

$$E[z_{j,m}, x_j^{InCom,miss}|\mathbf{x}_j^{InCom,obs}, \theta_t] = q_{j,m}(\hat{x}_{j,m}^{miss})$$
$$E[z_{j,m}, x_j^{InCom,miss} x_j^{InCom,miss\top}|\mathbf{x}_j^{InCom,obs}, \boldsymbol{\theta}_t] =$$
$$q_{j,m}(\boldsymbol{\Sigma}_m^{miss} - \boldsymbol{\Sigma}_m^{miss.obs}\boldsymbol{\Sigma}_m^{obs^{-1}}\boldsymbol{\Sigma}_m^{miss.obs^\top} + \hat{x}_{j,m}^{miss}\hat{x}_{j,m}^{miss\top})$$

2) M-step: The mixing weights are updated as:

$$\pi_m = \frac{h_m + r_m}{n + n_2}$$

while the GMM means are updated as:

$$\mu_m = \frac{\sum_{l=1}^{n} \mathbf{x}_l^{Com} h_{l,m} + \sum_{v=1}^{n_2} \mathbf{x}_v^{InCom} r_{v,m}}{h_m + r_m}$$

In case of a rhythmic DMP, the phase signal lies on a circular plane for whcih mean of cos and sin terms is needed:

$$\begin{aligned} \bar{cx} &= \frac{\sum_{l=1}^{n} \cos(x_i^{Com,miss}) h_{l,m} + \sum_{v=1}^{n_2} \cos(x_i^{InCom,miss}) r_{v,m}}{h_m + r_m} \\ \bar{sx} &= \frac{\sum_{l=1}^{n} \sin(x_i^{Com,miss}) h_{l,m} + \sum_{v=1}^{n_2} \sin(x_i^{InCom,miss}) r_{v,m}}{h_m + r_m} \end{aligned}$$

Afterwards the phase variable for a rhythmic DMP is updated in the GMM means with these conditions and wrapped in the interval $[0, 2\pi]$:

$$\begin{aligned} if(\bar{cx}) < 0) \qquad & \mu_m^{miss} = \tan^{-1}\frac{\bar{sx}}{\bar{cx}} + \pi \\ elseif(\bar{sx}) > 0) \qquad & \mu_m^{miss} = \tan^{-1}\frac{\bar{sx}}{\bar{cx}} \\ else \qquad & \mu_m^{miss} = \tan^{-1}\frac{\bar{sx}}{\bar{cx}} + 2\pi \end{aligned}$$

As before, for a rhythmic DMP map phase variable within the interval $[\mu_m^{miss} - \pi, \mu_m^{miss} + \pi]$. Afterwards the covariances are updated as:

$$\boldsymbol{\Sigma}_m = \frac{\sum_{i=1}^{n} h_{i,m}(\mathbf{x}_i^{Com} - \boldsymbol{\mu}_m)(\mathbf{x}_i^{Com} - \boldsymbol{\mu}_m)^\top + \sum_{l=1}^{n_2} \mathbf{A}_l}{h_m + r_m}$$

$$\begin{aligned} \mathbf{A}_l &= r_{l,m} \begin{bmatrix} \mathbf{x}_l^{InCom,obs} - \boldsymbol{\mu}_m^{obs} \\ x_l^{InCom,miss} - \mu_m^{miss} \end{bmatrix} \begin{bmatrix} \mathbf{x}_l^{InCom,obs} - \boldsymbol{\mu}_m^{obs} \\ x_l^{InCom,miss} - \mu_m^{miss} \end{bmatrix}^\top \\ &= \begin{bmatrix} \mathbf{A}_{11} & \mathbf{a}_{12} \\ \mathbf{a}_{21} & a_{22} \end{bmatrix} \end{aligned}$$

where

$$\begin{aligned}
\mathbf{A}_{11} &= r_{l,m}(\mathbf{x}_l^{InCom,obs} - \boldsymbol{\mu}_m^{obs})(\mathbf{x}_l^{InCom,obs} - \boldsymbol{\mu}_m^{obs})^\top \\
\mathbf{a}_{21} &= (E[z_{l,m}, x_l^{InCom,miss}] - r_{l,m}\mu_m^{miss})(\mathbf{x}_l^{InCom,obs} \\
&\quad -\boldsymbol{\mu}_m^{obs})^\top \\
\mathbf{a}_{12} &= \mathbf{a}_{21}^\top \\
a_{22} &= E[z_{l,m}, x_l^{InCom,miss} x_l^{InCom,miss\top}] + \\
&\quad r_{l,m}\mu_m^{miss}(\mu_m^{miss})^\top - 2E[z_{l,m}, x_l^{InCom,miss}](\mu_m^{miss})^\top
\end{aligned}$$

After updating the GMM parameters the next step is to maximize the phase signal values in the incomplete data set. This is done by using Gaussian Mixture Regression but with responsibilities calculated using all of the dimensions (phase values from initialization or from last M-step). For a given input variable $\mathbf{x}_i^{InCom,obs}$ and a given Gaussian distribution m, the expected value of $x_i^{InCom,miss}$ is defined by:

$$\mathcal{P}(x_i^{InCom,miss}|\mathbf{x}_i^{InCom,obs}, m) = \hat{x}_{i,m}$$

where

$$\hat{x}_{i,m} = \mu_m^{miss} + \boldsymbol{\Sigma}_m^{miss,obs}(\boldsymbol{\Sigma}_m^{obs})^{-1}(\mathbf{x}_i^{InCom,obs} - \boldsymbol{\mu}_m^{obs})$$

By considering the complete GMM

$$E(x_i^{InCom,miss}|\mathbf{x}_i^{InCom,obs}) = \sum_{l=1}^{k} r_{i,l}\hat{x}_{i,l}$$

$$\text{with } r_{i,m} = \frac{\pi_m \mathcal{N}(\mathbf{x}_i^{InCom}; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m)}{\sum_{l=1}^{k} \pi_l \mathcal{N}(\mathbf{x}_i^{InCom}; \boldsymbol{\mu}_l, \boldsymbol{\Sigma}_l)}$$

where $E(x_i^{InCom,miss}|\mathbf{x}_i^{InCom,obs})$ is the updated phase value of the i^{th} data point in incomplete data set.

IV. Experimental Results

The proposed approach is evaluated using a peg-in-hole task with a master-slave teleoperation system. The setup consists of SensAble PHANToM Omni as a master device and PHANToM Premium 1.5A as a slave. An aluminum plate with 4 holes as shown in Figure 3 is used for the peg-in-hole task. The operator observed the visual feedback through video displayed on the monitor and controlled the slave with the same orientation as the view from the camera. One execution cycle constitutes of inserting the robot end-effector into the four holes in counter-clockwise direction, while starting and ending above the same hole. During the data collection phase, slave records the Cartesian positions, which are being used for calculating the velocities and accelerations through numerical differentiation. Separate DMPs are learned for each considered degree of freedom i.e. the x, y and z-axis. All the models used phase signal as input and 20 Gaussian in the GMMs. The number of Gaussian in the GMMs can also be optimized [21]. Since the amplitude modification is not required, we set the amplitude modifier term a in the DMP equation to be 1.

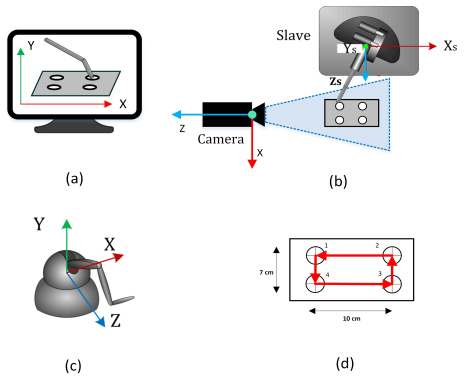


Fig. 3: (a) Visual feedback, (b) Slave with peg-in-hole task, (c) Haptic device, (d) Peg-in-hole task.

A. Experiment One: Temporal and spatial variations

An operator demonstrated four counter-clockwise cycles of the peg-in-hole task. The operator started and ended above the same hole. An operator needs some time to reach an adequate level of performance in teleopeartion and thus first few demonstrations can have large temporal and spatial variations [22], as visible in Figure 2d.

Figure 5b contains the result of LfD method presented in [8], [23] where the trajectories are first aligned by using DTW. Afterwards the phase signal and the spatial data are encoded by using a GMM. Finally GMR is used for motion retrieval. When applying GMR for a rhythmic task, the data for the circular dimension (phase signal) is always mapped in the interval $\mu - \mathbb{X}_i$ and $\mu + \mathbb{X}_i$ for calculating the valid responsibilities. The downsides of this approach are that firstly the value of phase signal cannot be inferred for arbitrary starting point of the end-effector using GMR. This is because the mapping from an output to input of a function is not guaranteed to be one to one. Secondly with phase signal as the only input, the GMM-GMR based encoding does not consider the current position of the end-effector when generating the motion. The learned trajectory partially completes the task by inserting peg in three out of four holes, as shown in Figure 5b.

The next model that we have considered is the DMP based encoding of the demonstrations, with phase signal and forcing terms encoded by using a GMM [24]. Again the forcing terms of different trajectories are first aligned with DTW. This approach fails to reproduce the task, as shown in Figure 5c. Because of dissimilarity in the forcing terms of different trajectories, the DTW fails to align them properly.

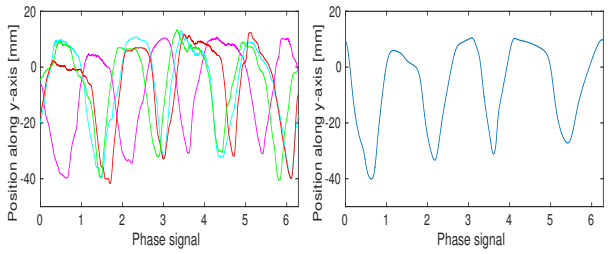
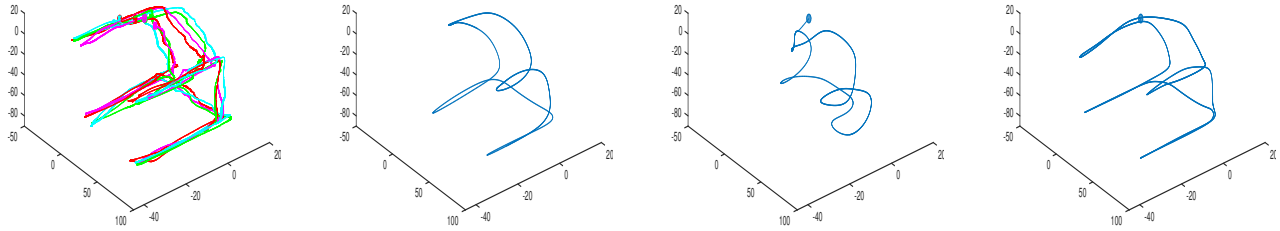
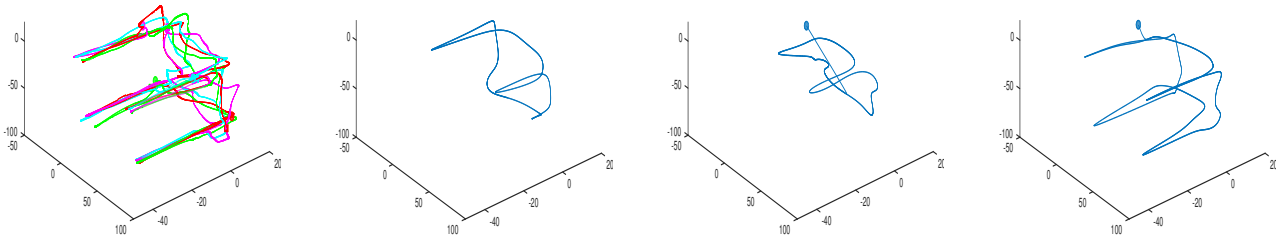


Fig. 4: Experiment 1: (a) Demonstrated motions and (b) learned trajectory for y-axis plotted against phase signal.



(a) (b) (c) (d)

Fig. 5: All axis are in millimeters. (a) Recorded motions for experiment 1, with starting and ending points in each trajectory indicated by a circle and a square respectively. Same starting point and ending point representation is also used in the other two experiments. (b-d) Motion reproductions with different models (b) DTW+GMM based encoding of spatial data with GMR based motion reproduction, (c) DMP model with DTW+GMM based encoding of forcing terms, (d) Our approach.



(a) (b) (c) (d)

Fig. 6: (a) All axis are in millimeters. Recorded motions for experiment 2. The motions start and end above different holes in each demonstration. (b-d) Motion reproductions with different models (b) DTW+GMM based encoding of spatial data with GMR based motion reproduction, (c) DMP model with DTW+GMM based encoding of forcing terms, (d) Our approach.

Figure 5d shows the result of our approach which successfully reproduces the task due to the temporal alignment performed during EM. The asynchronous nature of the motions and the learned motion along y-axis can be visualized in Figure 4. Another benefit of using our DMP based encoding is that it also considers the current position along with the phase signal when doing motion reproduction. The starting value of phase signal can easily be inferred by linearly generating samples of phase signal in between 0 and 2π and then calculating the acceleration value $\dot{v}$ for each of them by using the DMP equation. The sample which yields the lowest value of sum of absolute accelerations of all DMPs is used as the starting point for integrating the canonical system.

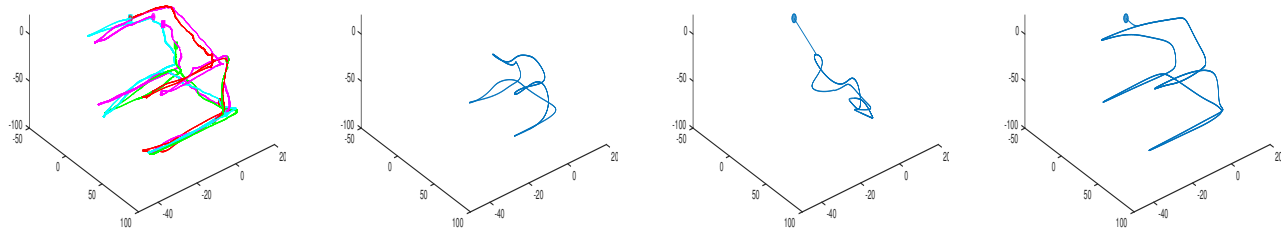
B. Experiment Two: Different initial and final conditions

In the second experiment, an operator was asked to complete four counter-clockwise cycles of the peg-in-hole task, each time starting and ending above a different hole, as illustrated in Figure 6a. A major constraint when applying DTW is that it assumes similar starting and ending positions for a signal, with temporal variations in between. If this condition is violated, its performance degrades. The major advantage with our approach is that it does not need to fulfill any such constraints. Since the holes are symmetric, a user can easily lose track of a single hole while demonstrating the peg-in-hole task in case of the change in camera view.

Figure 6b shows the result of DTW and GMM-GMR approach. The learned trajectory was only able to reach two holes due to the extreme misalignment of phase signal. Figure 6c shows the result of DMP encoding with DTW based alignment of forcing terms. This approach only reaches one hole and fails to learn the complete task due to the aforementioned reasons. Our approach successfully executes the task as shown in Figure 6d, because in our approach the trajectories alignment is performed during the EM.

C. Experiment Three: Incomplete demonstrations

This experiment utilizes data from the first experiment. One full trajectory is used while the others are clipped to simulate the partial executions. This experiment depicts the situation where a human operator may fail to provide a full execution of the motion and abort the execution before completing the full cycle. As shown in Figure 7b DTW aligned GMM-GMR based reproduction generates reaching motion for two out of four holes but did not go deep enough to insert the peg. DMP-GMM based encoding with DTW based alignment of forcing terms completely fails to execute the task as is depicted in Figure 7c. Again our approach successfully reproduces the learned task as shown in Figure 7d. The success rates of the three experiments are summarized in Table I.



(a) (b) (c) (d)

Fig. 7: All axis are in millimeters. (a) Demonstrations for experiment 3. (b-d) Motion reproductions with different models (b) DTW+GMM based encoding of spatial data with GMR based motion reproduction, (c) DMP model with DTW+GMM based encoding of forcing terms, (d) Our approach.

TABLE I: Success rate of different approaches shown as the number of holes reached.

	DTW GMM+GMR	DTW DMP+GMM	Our approach
Experiment 1	3	0	4
Experiment 2	2	1	4
Experiment 3	0	0	4

V. CONCLUSION

In this paper we have shown the typical problems encountered while applying existing LfD approaches for teleoperation applications. We proposed an EM based approach for estimating GMM parameters and phase variables. The proposed approach showed how to align and encode the trajectories simultaneously. Although we have shown results for a rhythmic peg-in-hole task, the proposed approach possess all the desirable properties associated with a DMP model and can encode discrete as well as rhythmic motions. Through multiple peg-in-hole experiments we have demonstrated that our proposed approach can handle large variabilities in the teleoperation demonstrations (e.g., large spatial and temporal variabilities, demonstrations with different starting/ending phases, partial task executions). Our proposed method outperformed existing approaches based on combining DTW based time alignments and trajectories encoding algorithms.

REFERENCES

[1] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, "Robot programming by demonstration," in *Springer handbook of robotics*, 2008, pp. 1371–1394.

[2] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[3] K. Hu, C. Ott, and D. Lee, "Online human walking imitation in task and joint space based on quadratic programming," in *IEEE Int. Conf. on Robotics and Automation*, 2014, pp. 3458–3464.

[4] C. Ott, D. Lee, and Y. Nakamura, "Motion capture based human motion recognition and imitation by direct marker control," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 2008, pp. 399–405.

[5] D. Lee and C. Ott, "Incremental motion primitive learning by physical coaching using impedance control," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2010, pp. 4133–4140.

[6] M. Saveriano, S. An, and D. Lee, "Incremental kinesthetic teaching of end-effector and null-space motion primitives," in *IEEE Int. Conf. on Robotics and Automation*, 2015, pp. 3570–3575.

[7] P. F. Hokayem and M. W. Spong, "Bilateral teleoperation: An historical survey," *Automatica*, vol. 42, no. 12, pp. 2035–2057, 2006.

[8] B. Akgun, K. Subramanian, and A. Thomaz, "Novel interaction strategies for learning from teleoperation," *AAAI Fall Symposium Series*, pp. 2–9, 2012.

[9] J. Y. X. C. ChenZ, "Hidden markov model approach to skill learning and its application to telerobotics," 1993.

[10] R. A. Peters, C. L. Campbell, W. J. Bluethmann, and E. Huber, "Robonaut task learning through teleoperation," in *IEEE Int. Conf. on Robotics and Automation*, 2003, pp. 2806–2811.

[11] L. Rozo, P. Jimenez Schlegl, and C. Torras, "Sharpening haptic inputs for teaching a manipulation skill to a robot," in *IEEE Int. Conf. on Applied Bionics and Biomechanics*, 2010, pp. 331–340.

[12] L. D. Rozo, P. Jiménez, and C. Torras, "Learning force-based robot skills from haptic demonstration." in *CCIA*, 2010, pp. 331–340.

[13] L. Rozo, P. Jiménez, and C. Torras, "A robot learning from demonstration framework to perform force-based manipulation tasks," *Intelligent service robotics*, vol. 6, no. 1, pp. 33–51, 2013.

[14] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[15] K. Fischer, F. Kirstein, L. C. Jensen, N. Kr, K. Kukli, M. aus der Wieschen, and T. Savarimuthu, "A comparison of types of robot control for programming by demonstration," in *ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2016, pp. 213–220.

[16] B. Akgun and K. Subramaman, "Robot Learning from Demonstration: Kinesthetic Teaching vs. Teleoperation," p. 7, 2011.

[17] P. Sanguansat, "Multiple multidimensional sequence alignment using generalized dynamic time warping," *WSEAS Transactions on Mathematics*, vol. 11, no. 8, pp. 668–678, 2012.

[18] S. Schaal, "Dynamic movement primitives-a framework for motor control in humans and humanoid robotics," in *Adaptive Motion of Animals and Machines*. Springer, 2006, pp. 261–280.

[19] M. Power, H. Rafii-Tari, C. Bergeles, V. Vitiello, and G.-Z. Yang, "A cooperative control framework for haptic guidance of bimanual surgical tasks based on learning from demonstration," in *IEEE Int. Conf. Robotics and Automation (ICRA)*, 2015, pp. 5330–5337.

[20] S. Calinon, P. Evrard, E. Gribovskaya, A. Billard, and A. Kheddar, "Learning collaborative manipulation tasks by demonstration using a haptic interface," in *IEEE Int. Conf. on Advanced Robotics*, 2009, pp. 1–6.

[21] A. Pervez and D. Lee, "A componentwise simulated annealing em algorithm for mixtures," in *Joint German/Austrian Conference on Artificial Intelligence (KI)*, 2015, pp. 287–294.

[22] J. Bukchin, R. Luquer, and A. Shtub, "Learning in tele-operations," *IIE Transactions*, vol. 34, no. 3, pp. 245–252, 2002.

[23] S. Calinon, F. Guenter, and A. Billard, "On learning, representing, and generalizing a task in a humanoid robot," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 37, no. 2, pp. 286–298, 2007.

[24] T. Alizadeh, "Statistical learning of task modulated human movements through demonstration," Ph.D. dissertation, Istituto Italiano di Tecnologia, 2014.