DOI:
10.1016/j.robot.2022.104046

Document Version
Peer reviewed version

Environment-adaptive learning from demonstration for proactive assistance in human-robot collaborative tasks*

Kun Qian[a,b,c,*], Xin Xu[a,b], Huan Liu[a,b], Jishen Bai[a,b] and Shan Luo[c,d,*]

[a]School of Automation, Southeast University, Nanjing 210096, China
[b]Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education of China
[c]Department of Computer Science, University of Liverpool, Liverpool, L69 3BX, UK
[d]Centre for Robotics Research, Department of Engineering, King's College London, London,WC2R 2LS, UK

ARTICLE INFO

Keywords:
Learning-from-demonstration
Interactive movement primitives
Human-robot collaboration
Proactivity
Robot grasping

ABSTRACT

Proactive assistance in human-robot collaboration remains a challenging objective, as the spatial-temporal coordination of the human-robot motion must be considered in conjunction with the object and environmental context. In this paper, we propose an environment-adaptive probabilistic interaction primitive method using learning-from-demonstration. In particular, we propose a novel phase estimation algorithm called Single-axis Uniform Interval Interpolation, which alleviates the restriction of Gaussian or uniform distribution of phase variables. In addition, the environmental constraints in human-robot interactive skills are learned via the regression between environmental parameters and the weight vectors. The proposed method is implemented in a proactive robotic system for typical industrial-motivated human-robot collaborative scenarios, such as assistive push-button assembly and human-robot collaborative object covering. The experimental result validates the effectiveness of the proposed approach.

1. Introduction

Demonstration-based robotic skill learning provides an attractive potential for human-robot collaborative industrial applications, as it can significantly reduce the setup time of a cobot. Compared with traditional methods that manually tune the parameters of a robot's control system for different task instances, Learning-from-Demonstration (LfD) methods [52, 16, 50, 42] enable robots to synthesize their operational parameters according to human demonstrations. It provides an intuitive way of transferring human manipulation skills to robots, without the requirement of a dedicated expert. Besides, recent advances in markerless vision-based human hand motion capture [29, 49] as well as kinesthetic teaching [30, 40] have also supported the fast growth of LfD methods in sweep/strike tasks [35], block assembly [11, 45], object manipulation [37, 13], and other similar applications.

Nevertheless, applying LfD methods to human-robot collaborative tasks that involve handover [3] and pick-and-place motions has not been investigated enough. The key challenge for industrial cobots to interact with humans is safe and compliant movements. Although the LfD methods are well-suited for building intuitive correspondence between a teacher's physical movements and a robot's acceptable behavior, simple trajectory cloning is insufficient for the safety and compliance requirements in human-robot collaborative tasks. Robots are required to adapt to human behaviors by dynamically changing their pre-planned tasks. Therefore, in this paper, we have particular concerns in the following two aspects.

To begin with, the safety and compliance standard requires spatial-temporal coordination of human-robot movements. For example, by observing the human movements, a robot partner should recognize the partner's intention of reaching for a screw or holding a screwdriver, as well as predict his/her movements to spatially coordinate its trajectory w.r.t the human movement.

By leveraging Interaction Probabilistic Movement Primitive (IProMPs) [23], spatially coordinated robot trajectories are generated in conjunction with the prediction of human motion. To further account for temporal variations of the IProMP model [26, 24], the phase of a human trajectory during execution is estimated in a probabilistic framework, which enables generating proactive robot trajectories before the human finishes his/her movement. However, current phase estimation methods assume Gaussian or uniform distributions of a probabilistic phase variable. But for the demonstration motions with non-Gaussian or non-uniform phase variables, current methods may fail in estimating the phase precisely.

Secondly, Learning-from-demonstration (LfD) is not a simple replay. The generalization capability requires an industrial cobot to adapt its behaviors to the variance in task parameters as well as environmental changes. Such environmental variance includes perturbation in realistic environments (e.g., obstacles and other environmental changes) and new task parameters (e.g., varied goals). This indicates that IProMPs should incorporate additional information from the immediate environment, rather than solely relying upon the observations of the human.

To address these problems, this paper presents an environmental adaptive IProMP method to program cobots

*This work is sponsored by the Natural Science Foundation of Jiangsu Province (No.BK20201264), Zhejiang Lab (No.2022NB0AB02), and the National Natural Science Foundation of China (No.61573101). This work is also partially supported by the China Scholarship Council.

kqian@seu.edu.cn (K. Qian); shan.luo@liverpool.ac.uk (S. Luo)
ORCID(s): 0000-0001-7429-1742 (K. Qian); 0000-0003-4760-0372 (S. Luo)

adaptively for enhanced generalization capability to suit environmental changes. Our work extends IProMPs [26, 24] by learning the regression between environmental parameters and the weight vectors, hence the latent environmental constraints on the human-robot joint trajectories can be learned. In particular, we focus on handling the temporal variance of demonstrations via more accurate phase estimation rather than temporal normalization. In this regard, we propose a new phase estimation algorithm called Single-axis Uniform Interval Interpolation. Therefore, we alleviate the restriction of Gaussian or uniform distribution on phase variables. In addition, the proposed method is implemented as an LfD system, namely, $PickAssist$, for human-robot collaborative tasks that involve robot pick-and-place manipulations. The system is capable of predicting human movements, generating task-constrained grasp pose toward unseen objects, and coordinating the robot's reactive motion before his/her movement ends.

A similar idea to our work is task-parameterized trajectory learning of robotic manipulation skills [7, 51]. But our work is different from theirs in that we aim at human-robot trajectory correlation while these studies are not concerned about human-robot collaboration. Another similar work in [6] has proposed environment-adaptive interaction primitives that capture spatial human-robot interactive skills. Our work is different from [6] in that we additionally consider the temporal generalization issue for dealing with the temporal variance of demonstrations. Another difference is that [6] is based on Interaction Primitives (IPs) while our approach leverages IProMPs, which are more stable in trajectory prediction and more suitable for the sparse observation of human hand motions.

In this paper, our major contributions are:

- For multiple human-robot joint demonstration trajectories, we alleviate the restriction of their phase variable satisfying Gaussian or uniform distribution by introducing a novel phase estimation algorithm called Single-axis Uniform Interval Interpolation. The algorithm improves the phase estimation accuracy and human motion prediction accuracy in situations under environmental changes, where Gaussian or uniform assumption is violated.

- To accommodate for the capability of adapting robot behaviors to suit the variance in environment parameters in human-robot collaboration tasks, we extend IProMPs to be environmental-adaptive by learning the regression between environmental parameters and the weight vectors. Therefore, the environmental context embedded in the human-robot joint demonstration trajectories can be learned.

- We have developed a proactive robotic system called $PickAssist$ that seamlessly integrated the Environmental adaptive IProMP (EIProMP) with the proposed phase estimation algorithm. We experimentally verified the favorable performance of the proposed system in two application scenarios: assistive push-button assembly and human-robot collaborative object covering.

The remainder of this paper is organized as follows. Section 2 gives a brief review of the related work. Section 3 describes the concept and system architecture. Section 4 explains the proposed EIProMP model, the learning method as well as the new phase estimation algorithm. We explore the effectiveness of the proposed system through extensive experiments and evaluation in Section 5, followed by a conclusion of the paper.

2. Related Work

The advantage of active, human-in-the-loop interaction has been demonstrated in many human-robot collaborative tasks such as table assembly [43], covering objects [6], and other manufacturing scenarios [34, 47, 48, 27, 28, 46, 44]. In such situations, the collaborative manipulation of objects remains a challenging objective because robots must learn to not only adapt to different human partners [26, 24, 20] but also to new task and environment parameters [6].

Reinforcement Learning (RL) has found a huge application in robotic task learning. Model-free RL approaches optimize the target policy directly and do not assume prior knowledge of environmental dynamics. But model-free RL approaches that train a robot to learn any given task would require a prohibitively large amount of data and time due to the sparsity of rewards in the real world. Compared with model-free approaches, model-based reinforcement learning approaches are relatively sample-efficient. But model-based methods require constructing a model using data collected from the interaction with the environment. Therefore, the model accuracy becomes a bottleneck to policy quality [14]. In addition to data efficiency, another concern is that various physical limits make applying RL in real-world robotic systems much more difficult than in simulation [19].

Compared with RL methods that learn from scratch, LfD requires fewer interactions because it can be performed using efficient supervised learning. Moreover, in LfD, instead of hand-engineered reward functions, experts provide a set of demonstrations, which is usually easier than defining the reward. Therefore, in this paper, we focus on learning real-world human-robot collaborative tasks from a limited number of human demonstrations.

For general LfD problems, Behavior Cloning (BC), Inverse Reinforcement Learning (IRL) and Dynamical Movement Primitives (DMPs) variants [41, 18, 22] are three commonly applied approaches. In all cases, the spatial and temporal generalization capability under noisy human demonstration data is a significant concern.

Behavior Cloning approaches learn the mapping between states and actions from a sequence of demonstrated state-action pairs using supervised training. Under the condition of limited samples, the policy obtained directly based on behavior cloning will encounter the "distributional shift" problem [12], wherein a cloned policy might fail in

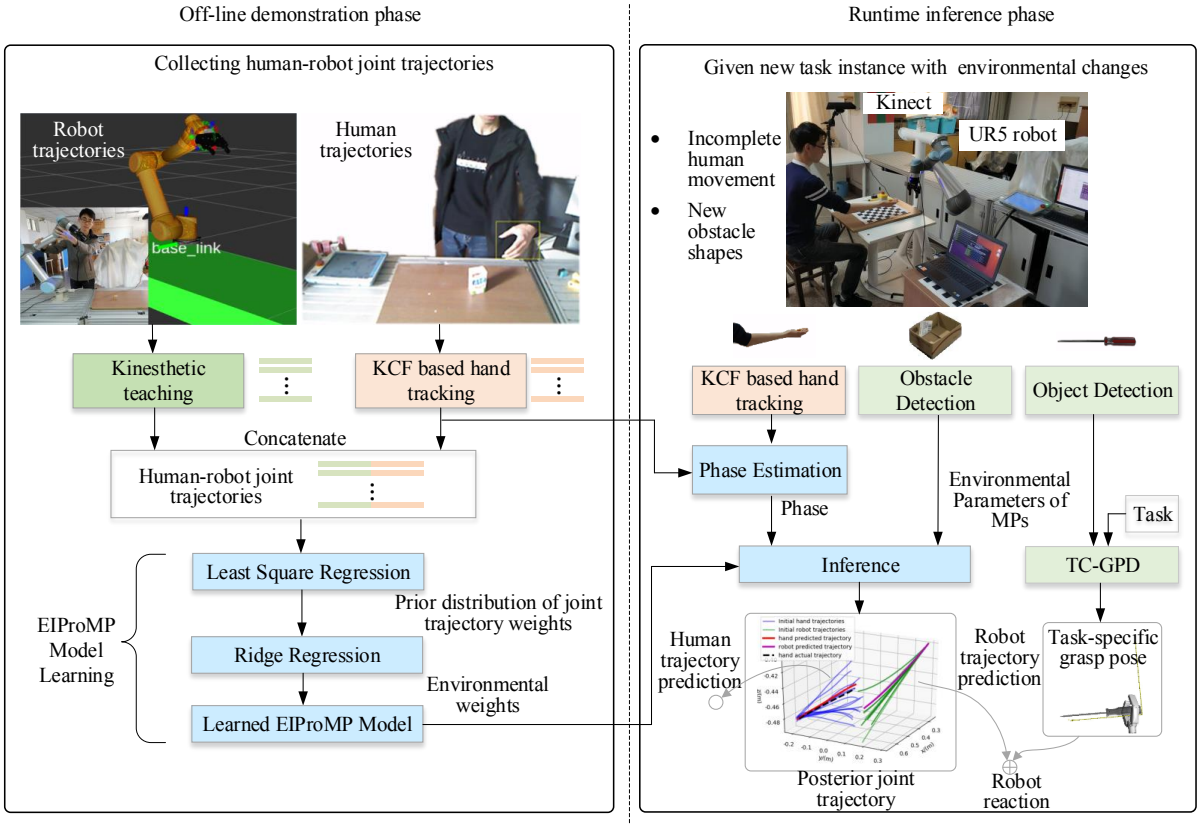
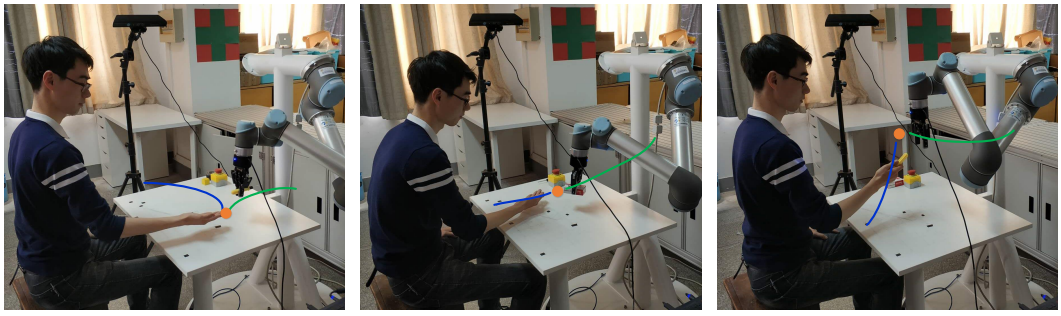


Figure 1: The *PickAssist* system overview. The system is designed to work in an offline training phase and a runtime inference phase. In the offline training phase, an EIProMP model is learned from human-robot joint trajectories. Meanwhile, in the runtime inference phase, the human-robot joint motion is predicted from partial observation of the human movement.

unfamiliar scenes during autonomous execution. Solutions to this problem [39] rely on iteratively querying an expert based on states encountered by some intermediate cloned policy. To deal with the non-Markovian and inconsistent nature of human demonstration movements, BC can be integrated with RNNs such as LSTM to incorporate temporal information [38]. The results in [38] indicate that the memory presented in RNNs helps the robot to remember and stick to a particular strategy similar to humans, under the condition that human demonstrations at each time are inconsistent. Inverse Reinforcement Learning (IRL) can deduce the reward function based on the given limited teaching data, so as to improve the generalization performance of the learning strategy [10]. Benefiting from the merits of IRL, several works [9][32] have applied it for predicting behaviors such as autonomous driving behaviors. Nevertheless, IRL methods have generally been less efficient than direct methods for learning from demonstration [10]. In addition, human hand movement demonstration in the real world collaboration scenarios is noisy, inconsistent, and may not be precisely optimal, which may bring about difficulties in solving a reward function that can best explain the expert behavior, or there may exist many rewards that can explain an optimal policy [10].

Dynamic systems such as DMP variants are able to encode and reproduce trajectories with spatial variance. These approaches are more tractable because they require a limited number of human demonstrations and maintain relatively low computational complexity, which is essential for proactive robotic behavior. A recently introduced method to learn interactions from demonstrations is the framework of Interaction Primitives (IPs) [2]. The DMPs-based IPs maintain a distribution over the DMP parameters by learning from multiple demonstration trajectories. During the execution phase, IPs firstly observe partial trajectory from human partners and identify the current phase of the interaction using Dynamic Time Wrapping (DTWs), and then compute the distribution over DMP parameters to control the robot to cooperate with human partners. The IPs model is based on the situation that multiple demonstration tracks are repeatedly demonstrated and keep the same shape. Cui *et al* [5, 6] take environment variables as additional degrees of freedom of human-robot joint trajectory based on IPs, and participate in Kalman filter iteration. The predicted trajectory depends on part of human observation trajectory and environment variables at the same time, which can modulate the robot trajectory affected by environmental factors in real-time. The problem with the IPs model is that they do not solve the general problem of phase estimation and the trajectory prediction result might be unstable in situations with temporal variances.

The spatial and temporal uncertainties in human-robot



(a) (b) (c)

Figure 2: Examples of human-robot joint demonstrations (a)$handover\_screwdriver$ (b)$handover\_screw$,(c)$fetch\_screwdriver$

interaction incite the use of a probabilistic framework instead of DMP variants to encode movements. To this end, the implementation of Interaction Primitives was later extended to Probabilistic Movement Primitives (ProMPs) [33]. A key advantage of using ProMPs is that uncertainty at all levels of modeling can be represented through probability theory, especially the uncertainties associated with human motion and the variety of possible trajectories. Based on ProMPs, Maeda [26, 24] and Ewerton [8] propose Interactive Probabilistic Movement Primitive (IProMP) for handling the spatial and temporal uncertainties in the interaction. IProMPs enable more stable human motion prediction and they are more suitable for sparse hand motion observations. However, the lack of considering temporal variance in the original IProMP methods leads to the fact that the robot must wait for the human to complete before it can start to react.

A prospect to handle this problem is that during execution the temporal variations of human movements are governed by the phase. Under partial and occluded observations, the use of DTW for temporally aligning trajectories becomes impractical [24, 17]. Maeda and Neumann [25] assume that the phase of multiple demonstration joint trajectories satisfy Gaussian distribution, and then they generate multiple phase candidate variables for human partial observation trajectories. Based on the IProMP model, they solve the phase estimation problem using partial observation of human trajectories. Ewerton [8] uses GMMs to model the trajectories demonstrated in different directions. The multiple demonstration trajectories of each task with different directions are modeled as a Gaussian kernel. The Gaussian kernel is selected according to the initial direction of the partner's action to determine the task category. Campbell [1, 4] designs Ensemble Bayesian Interaction Primitives (eBIP), which uses multi-modal data such as electromyogram signal, joint force, and visual information to predict robot trajectory in real-time.

3. System Overview

As depicted in Figure 1, in the offline training phase, human and robot trajectories of training samples are obtained separately with time stamps and then combined later in the same coordinate frame. Figure 2 and Figure 3 illustrate the examples of "$handover\_screwdriver$", "$handover\_screw$", and "$fetch\_screwdriver$" tasks [21] as well as the human-robot joint trajectories. For the first two tasks, we demonstrated distinctive trajectories for easier classification of the human intention by using a simple Bayesian classifier. Therefore, the required objects to be picked can be automatically determined according to the hand motion observations.

The human demonstration trajectories are obtained by a Kernelized Correlation Filters (KCF) based hand tracker [49, 21] using a Kinect camera. For a human demonstration trajectory, the corresponding robot's kinesthetic teaching was performed with the approximately same duration as that of the human partner. A smooth and dense robot trajectory is obtained by interpolation between the taught keyframes using MoveIt. To synchronize a pair of human-robot training trajectories, we firstly force the duration of the robot trajectory to be equal to that of the human partner and then resample the robot trajectory according to the human trajectory keyframes. Then the proposed Environment-adaptive Interactive Probabilistic Movement Primitive (EIProMP) models are learned from the human-robot joint trajectories.

In the runtime inference phase, the human partner performs the intended action, e.g., reaching for a tool for the assembly task. Then the $Inference$ module predicts the human-robot joint trajectory based on the incomplete observation of human movement. In particular, a phase estimation algorithm called Single-axis Uniform Interval Interpolation is proposed for addressing the temporal variability of human movements between the demonstration and the execution. To draw contextual parameters, the $ObstacleDetection$ module automatically measures the environmental parameters such as the obstacle height and width, using another Kinect camera. The $TC-GPD$ module conducts Task-Constrained Grasp Pose Detection [36] on the objects to be manipulated, e.g., a screwdriver or a hammer. The combination of the predicted robot's reactive trajectory and the task-specific optimal grasp pose ensures the robot compliantly reacts to the human partner before his movement ends. Most importantly, the proposed EIProMP model allows the generation of trajectories that adapt to environmental changes in the robot's workspace. In this paper,

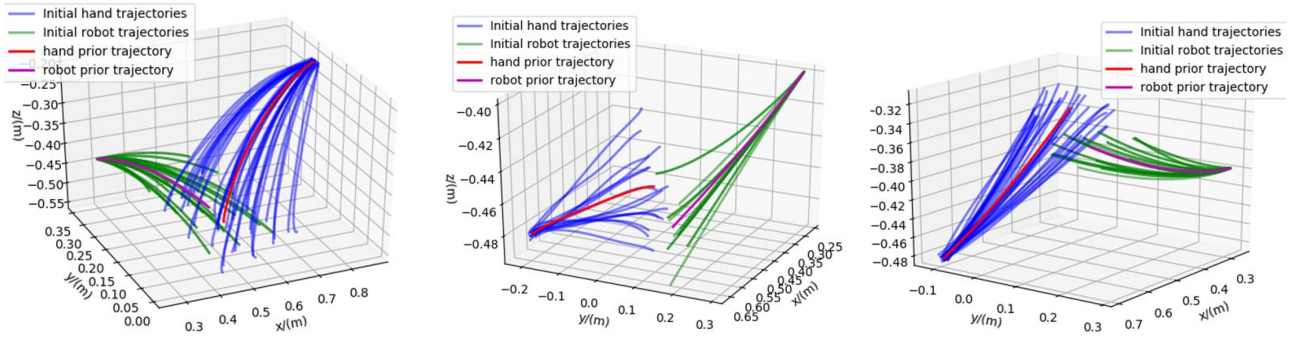


Figure 3: Examples of human-robot joint trajectories (a)*handover_screwdriver* (b)*handover_screw*,(c)*fetch_screwdriver*

we consider the variances in obstacle sizes as environmental changes, which are formulated as environmental parameters.

4. Approach

4.1. EIProMPs

IProMPs denote the DoFs of human and robot's demonstration trajectories are P and Q, respectively. For a time step t a concatenated position is represented by $\mathbf{y}_t$[26]:

$$\mathbf{y}_t = [y^H_{1,t}, ..., , y^H_{P,t} y^R_{1,t}, ..., y^R_{Q,t}]. \tag{1}$$

Thus a trajectory with T time steps is represented as $\mathbf{y}_{1:T}$. The superscript $(.)^H$ and $(.)^R$ represent the human and robot DoFs, respectively. In EIProMPs, we introduce an additional environmental DoF $\mathbf{E}$ that models the task-specific environmental parameters, e.g., the width and height of an obstacle. For M human-robot joint demonstration trajectories $\{\mathbf{y}_{1:T,m}, \mathbf{e}_m\}|^M_{m=1}$, the environment parameters are denoted as $\mathbf{E} = [\mathbf{e}_1, ..., \mathbf{e}_M]^{\mathrm{T}}$, and the mth demonstration trajectory is $\mathbf{y}_m$:

$$\mathbf{y}_m = [\mathbf{y}^H_m, \mathbf{y}^R_m, \mathbf{e}_m] \tag{2}$$

Assuming a smooth trajectory, a parameterization of $\mathbf{y}_t$ can be achieved by linear regression on N Radical Basis Functions (RBFs), denoted as

$$\mathbf{\Psi}_t = [\psi_1(t), \psi_2(t), ..., \psi_N(t)]^{\mathrm{T}}, \tag{3}$$

Here $\psi_i(t)$ is the ith RBF at time step t, and is associated with a weight vector $\mathbf{w}_i$. Thus the parameterization of M demonstration trajectories leads to a set of weight vectors $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_M]$. According to [26], we model $\mathbf{w}$ to follow a Gaussian distribution and define a learning parameter θ to govern the distribution of $\mathbf{W}$, such that $p(\mathbf{w}; \theta) \sim N(\mathbf{w}|\mu_\mathbf{w}, \mathbf{\Sigma}_\mathbf{w})$, and $\theta = \{\mu_\mathbf{w}, \mathbf{\Sigma}_\mathbf{w}\}$. Then the mean $\mu_\mathbf{w}$ and the covariance $\mathbf{\Sigma}_\mathbf{w}$ can be learned from M demonstration trajectories using Least Squares Regression. With the learned parameters θ, trajectories are parameterized as

$$p(\mathbf{y}_t; \theta) = p(\mathbf{y}_t|\mathbf{w}) \sim N(\mathbf{y}_t|\mathbf{H}_t\mathbf{w}, \mathbf{\Sigma}_y), \tag{4}$$

in which $\mathbf{H}_t = diag((\mathbf{\Psi}^{\mathrm{T}}_t)_1, ..., (\mathbf{\Psi}^{\mathrm{T}}_t)_P, (\mathbf{\Psi}^{\mathrm{T}}_t)_1, ..., (\mathbf{\Psi}^{\mathrm{T}}_t)_Q))$. Here $\mathbf{P}$ and $\mathbf{Q}$ represent the dimensions of human trajectories and robot trajectories, respectively.

The key of our EIProMP is that by assuming a regression relation between $\mathbf{E}$ and $\mathbf{W}$, we define an environment related weight vector β such that:

$$\mathbf{E} = \mathbf{W}^{\mathrm{T}}\beta, p(\mathbf{E}|\mathbf{W}) \sim N(\mathbf{E}\left|\mathbf{W}^{\mathrm{T}}\beta, \sigma^2_E\mathbf{I}\right) \tag{5}$$

$$\beta = \left(\mathbf{W}^{\mathrm{T}}\mathbf{W} + \lambda\mathbf{I}\right)^{-1}\mathbf{W}^{\mathrm{T}}\mathbf{E} \tag{6}$$

Here β can be computed using Least Squares Regression. In particular, Ridge Regression is employed to obviate overfitting, and λ is a regularization term, and thus β can be calculated.

In the runtime inference phase, joint trajectries are predicted given a new environmental parameter $\mathbf{e}^*$. The observation at time step t that only contains human trajectory is denoted as $\mathbf{y}^O_t$:

$$\mathbf{y}^O_t = [y^H_{1,t}, ..., y^H_{P,t} y^R_{1,t}, ..., y^R_{Q,t}] = [y^H_{1,t}, ..., y^H_{P,t}, 0, ..., 0], \tag{7}$$

and the observation during the time interval $[t_1, t_2]$ is $\mathbf{y}^O_{t_1:t_2} = [\mathbf{y}^O_{t_1}, ..., \mathbf{y}^O_{t_2}]^{\mathrm{T}}$. The key to inferring the robot trajectory is to estimate the posterior probability of the weights based on the incomplete observation $\mathbf{y}^O_{t_1:t_2} \in R^{S\times(P+Q)}$ that contains S human movement points.

With the given new environmental parameter $\mathbf{e}^*$, the augmented observation is denoted as $[\mathbf{y}^O_{t_1:t_2}, \mathbf{e}^*]$. A Kalman filter is employed to iteratively estimate the posterior parameters $\mu^{\text{new}}_\mathbf{w}$ and $\mathbf{\Sigma}^{\text{new}}_\mathbf{w}$, whose initial values are taken from the prior distribution of $\mathbf{w}$ learned from the training demonstration trajectories. The resulting posterior probability of the weights is denoted as θ_{new}. Then the trajectory distributions that predict the human movement are obtained according to Equation (4).

4.2. Phase Estimation

Estimating the temporal variability of demonstrated movements is modeled as the phase estimation problem. For the training samples, the phase estimation is achieved by

using the time alignments method such as DTW. For the partially observed human movements during execution, it is required to estimate the best fitting phases for the observed trajectory points according to the phases learned from the demonstration samples. Such phase estimation is necessary for predicting human movements to ensure proactive robot movements before the human execution movement ends.

Denote the phases of M human-robot joint demonstration trajectories are $\mathbf{T} = \{T_i|, i = 1, ..., M\}$. The scaling factor α_i of each trajectory can be computed as [24]:

$$\alpha_i = T_i / T_{nom} \tag{8}$$

$$T_{nom} = (\sum_{i=1}^{M} T_i)/M \tag{9}$$

Phase estimation of the IProMP methods is achieved by learning a distribution over the phase variable $z(t)$. The IProMP methods in [26, 24] incorporate the temporal variance as part of the model by learning a distribution over phases from the same demonstrations previously used to create the IProMPs. These methods assume that the human movement is governed by a single phase and the phase ratios of different demonstrations vary according to a Gaussian or uniform distribution. By computing the most probable scaling value α^* according to the sparse partial sequence of observations, these methods estimate the phase variable as $z(t) = \alpha^* t$.

In contrast, in our push-button assembly and object covering cases, we have the following assumptions, which are reasonable and common in practice:

- Assumption 1: The joint trajectories are recorded in multiple task instances with different environmental parameters. A specific task contains some repeated demonstration movements.
- Assumption 2: In a specific dimension, F_{dim}, the multiple demonstrated trajectories are monotonous and they share the same start point and the same endpoint.
- Assumption 3: In our tasks, the robot's point-to-point movement is supposed to adapt to a human's motion while crossing over an obstacle, rather than bypassing one side of the obstacle. Therefore, we only consider the 2D variation of the obstacle size, i.e., the width and height.

According to Assumption 1, such demonstration motions with non-Gaussian or non-uniform phase variables are common in practice. However, previous studies [25, 24] assume that the phase ratios of different demonstrations vary according to a Gaussian distribution or uniform distribution.

To alleviate the Gaussian or uniform restriction of the phase variable of the demonstrated movements, we devise a novel phase estimation algorithm called Single-axis Uniform Interval Interpolation, as shown in Algorithm 1. In Algorithm 1, Line 1 and Line 2 indicate that each demonstrated trajectory is reconstructed by interpolating with N equal intervals in the F_{dim} dimension. Line 3 suggests that as a result of the interpolation, a nominal time index sequence $\{1, ..., N\}$ is obtained. Other dimensions are interpolated according to the same nominal time index sequence, which results in reconstructed trajectories in all dimensions.

Algorithm 1 Phase estimation

Input: Demonstration trajectories $\{\mathbf{y}_{1:T,m}, \mathbf{e}_m\}|_{m=1}^{M}$ and a partially observed trajectory $[\mathbf{y}^O_{t_1:t_2}, \mathbf{e}^*]$

Output: The aligned time index for the ending point of the observed trajectory

1: for mth demonstration trajectory $[\mathbf{y}_{1:T,m}, \mathbf{e}_m]$
2: $\{1, ..., N\} = Interpolate(\mathbf{y}_{1:T,m}, F_{\text{dim}})$
3: $\mathbf{y}_{1:N,m} \leftarrow \mathbf{y}_{1:T,m}$
4: end for
5: Calculate $p(\mathbf{w}; \theta) = N(\mathbf{w}|\mu_{\mathbf{w}}, \mathbf{\Sigma}_{\mathbf{w}})$, $\theta = \{\mu_{\mathbf{w}}, \mathbf{\Sigma}_{\mathbf{w}}\}$
6: $\mathbf{u} = \text{Average}\{\mathbf{y}_{1:N,m}\}, m = 1, ..., M$
7: $\mathbf{v} \leftarrow Interpolate(\mathbf{y}^O_{t_1:t_2}, F_{\text{dim}})$
8: $n^* = \text{DTW}(\mathbf{u}, \mathbf{v})$
9: Return n^*

Table 1
Denotion of different obstacle sizes

	$h = 0.117m$	$h = 0.185m$	$h = 0.259m$
$w = 0.070m$	env1	env4	env7
$w = 0.140m$	env2	env5	env8
$w = 0.210m$	env3	env6	env9

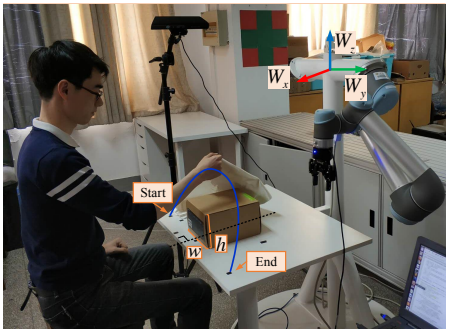


Figure 4: The "Object Covering With Cooperation (OCWC)" task scenario

As an example, the scenario of an "Object Covering With Cooperation"(OCWC) task is shown in Figure 4. The "OCWC" task involves human robot collaboratively covering a piece of cloth over a box, whose varied width and height are given as $\{w, h\}$, according to Assumption 3. The expected robot's behavior is to generate a synchronized trajectory that is spatially aligned to humans with a certain spatial shift. During the preparation of training data, we considered 9 different combinations of w and h as environmental parameters, in which $w \in \{0.070m, 0.140m, 0.210m\}$ and $h \in \{0.117\text{m}, 0.185m, 0.259m\}$. We denote the combinations as "env1",..., "env9", defined in Table 1. For each width-height pair, 10 trajectories of human-robot joint

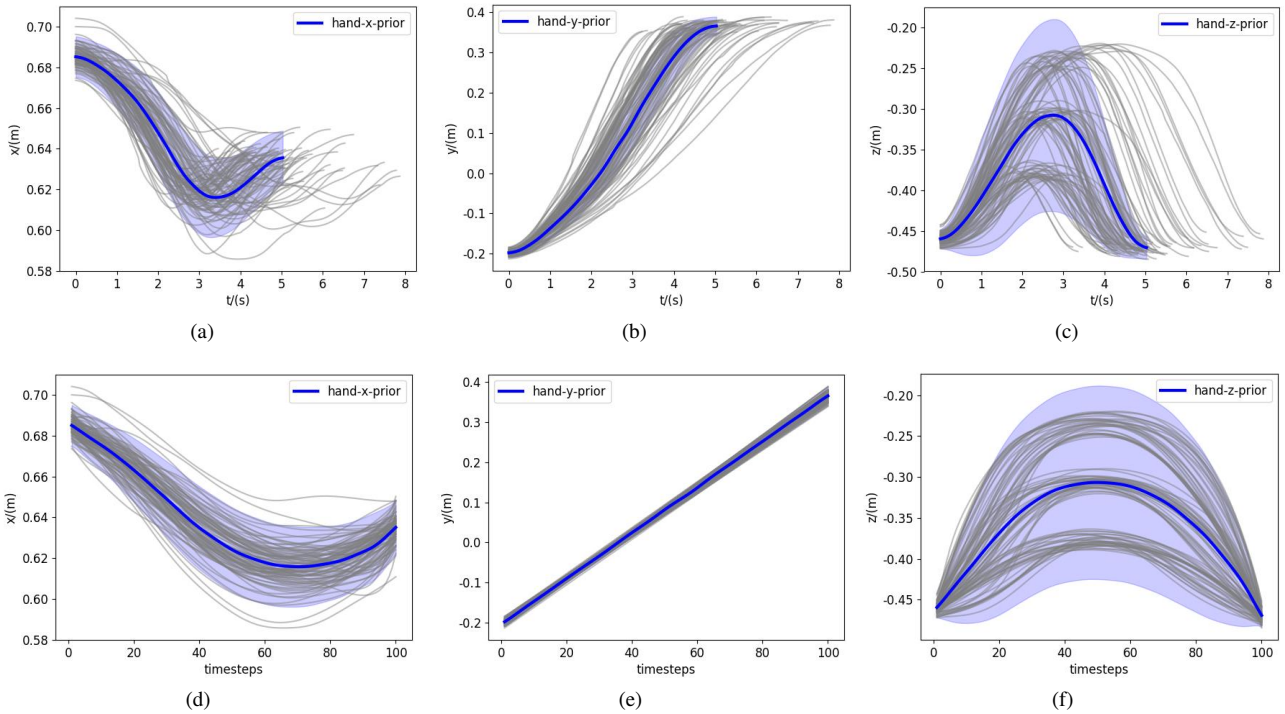


Figure 5: The raw training trajectories and the reconstructed trajectories of the "OCWC" task. (a), (b) and (c) are the X, Y, and Z data of the 90 training trajectories, respectively. (d), (e) and (f) are the reconstructed X, Y, and Z data, respectively.

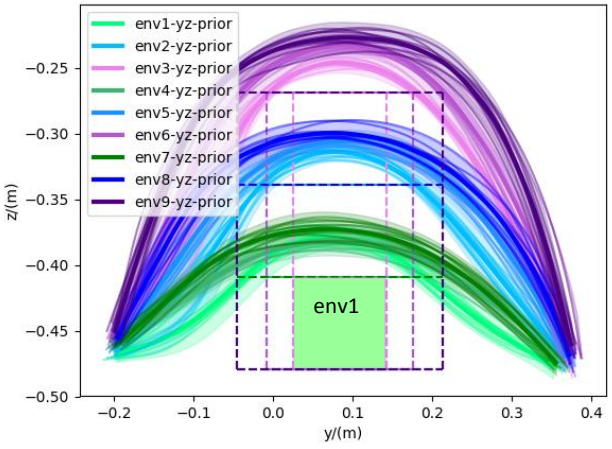


Figure 6: Human hand trajectories of the "OCWC" task in the Y-Z plane

demonstration movements were recorded and a total number of 90 trajectories for training were obtained. The X, Y, and Z data of the 90 training trajectories are shown in Figure 5(a), (b), and (c), respectively. In this example, we set F_{dim}="Y" according to Assumption 2. Then the Y-axis trajectories are processed using Single-axis Uniform Interval Interpolation with time steps N=100. The reconstruction result of the Y-axis trajectories is shown in Figure 5(e). Then we align the X-axis and the Z-axis data to the reconstructed Y-axis data, resulting in the reconstructed X-axis and the Z-axis trajectories as shown in Figure 5(d) and (f), respectively.

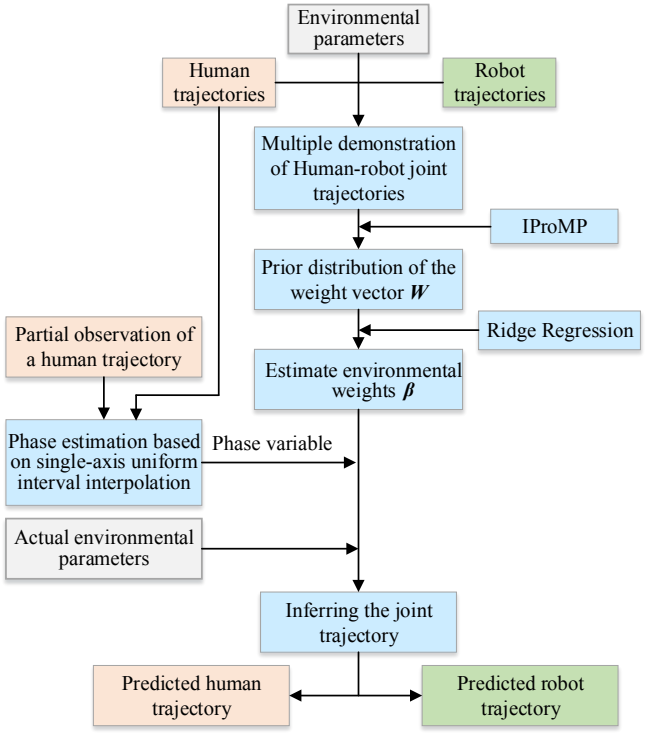


Figure 7: The overall diagram of the EIProMP method with phase estimation

The blue curves in Figure 5(d), (e), and (f) indicate mean

trajectories.

To more intuitively illustrate the benefit of trajectory reconstruction, we replot Figure 5 (e) and (f) in a Y-Z plane as shown in Figure 6. Nine dashed rectangles of different sizes indicate different $\{w, h\}$ pairs. Trajectories corresponding to these nine different $\{w, h\}$ pairs are also plotted in different colors. According to Figure 6, the reconstructed trajectories are capable of better reflecting the environmental features, compared with the original trajectories.

From the reconstructed demonstrated trajectories, the prior parameters of weights $\mathbf{w}$ are regressed, as indicated by Line 5 in Algorithm 1. The means of all demonstration trajectories on the interval points are computed as $\mathbf{u} = (u_1, ..., u_N)$, as indicated by Line 6. Each mean trajectory is taken as a reference trajectory that represents the reconstructed "mean" trajectory of the corresponding demonstrations. Then, for a partially observed human trajectory with S points, the same interpolation operation is performed in its F_{dim} dimension to obtain the testing trajectory $\mathbf{v} = (v_1, ..., v_S)$, as indicated by Line 7. The final step as shown in Line 8 is to utilize the DTW algorithm for computing the best-matched point of the endpoint of the test trajectory relative to the reference trajectory. The best matched time index n^* is the result of the estimated phase of v_S. Then the corresponding phases of other testing trajectory points can be estimated in a similar way.

To summarize, the overall diagram of the EIProMP method is shown in Figure 7. The proposed phase estimation method is suitable for partially and sparsely observed human movements. Another feature is that the estimated n^* is independent of the density of the observed trajectory as well as the time duration of the human movement.

5. Experiments

For comparison purposes, we first provide the result of IProMPs in Section 5.1 and then describe the experimental results of EIProMPs in detail in Section 5.2. We tested the accuracy of the proposed EIProMP model with the phase estimation algorithm, as well as their generalization capability under varied environmental parameters. In Section 5.3, we designed two typical application scenarios, assistive push-button assembly and human-robot collaborative object covering, for validating the viability of the *PickAssist* system.

5.1. Result of IProMPs

We begin with a brief description of testing IProMPs for the purpose of providing a comparison with the EIProMP. The IProMP method was tested on the "$handover\_screwdriver$" "$handover\_screw$" and "$fetch\_screwdriver$" tasks. Take the "$handover\_screw$" task as an example, Figure 8(a), (b), and (c) provide the prediction error of the human hand, robot's movement as well as the estimated phase, respectively, at different observation ratios. Given a proportion of human hand movement observation, the scaling factor α can be estimated using three different models, including the Maximum a posteriori method (MAP), the Minimum distance-based model(DI), and the Maximum likelihood estimation method (MLE)[26]. The first two methods assume that the scaling factor $\boldsymbol{\alpha} = \{\alpha_i|, i = 1, ..., M\}$ follows a Gaussian distribution, and the third method assumes uniform distribution. Please refer to [26] for more information about the three methods. The results in Figure 8 demonstrate that for all of the three models, the prediction error decrease, and the success rate increase with longer observation. Figure 9 shows the corresponding prediction result of the human partner's trajectory, in which the *obs_ratio* was empirically taken as 0.7. In Figure 9, the color dots represent the observation of human hand movement, and the shape of the predicted trajectory is consistent with the observation.

5.2. Result of EIProMPs

5.2.1. The choice of observation ratio

Taking the object covering task (see Figure 4) as an example, we first investigate what percentage of observations is feasible for predicting human motions, in the cases that the actual environmental parameter $\mathbf{e}^*$ during replay may be different from the given one.

We collected training trajectories under 9 different environmental parameters, 10 for each width-height pair, as mentioned in Section 4.2. The test was conducted using Leave-One-Out-Cross-Validation (LOOCV), which means when we test a specific trajectory at an observation ratio, its observed part is taken as the testing trajectory, whilst the rest 89 trajectories are used as the training data. We used the proposed method and tested the phase estimation accuracy, the human movement prediction accuracy as well as the success rate of human movement predictions under different observation ratios.

Figure 10 depicts the results when the 10^{th}, 40^{th}, 70^{th} and 90^{th} trajectories, taken at different observation ratios, are input as observations. In this experiment, the environmental parameter is set as $\{w = 0.21\text{m}, h = 0.259\text{m}\}$ (i.e., "env9" in Table 1). This means that the environmental parameter $\mathbf{e}^*$ involved in the test is previously "seen" during training, but only for the 90^{th} testing trajectories, the environmental parameter aligns with the given one.

Figure 10(a) indicates that the phase estimation errors are less than 3 time steps for all the testing trajectories. Figure 10(b) depicts the human motion prediction accuracy, in which the endpoint positional error between the predicted human hand trajectory and the ground truth was measured. The result in Figure 10(b) indicates that the human hand motion prediction errors grow with the observation ratio for the 10^{th}, 40^{th} and 70^{th} trajectory, because of the mismatched environment parameters. But for the 90^{th} trajectory, the prediction error remains relatively low.

Figure 10(c) illustrates the human motion prediction success rate. To be more specific, we define a successful human motion prediction if the predicted human trajectory crosses over the box to be covered and ends at a goal place with a positional error lower than 0.05m, compared with its ground truth endpoint. In applications of human-robot handover and object covering, such a positional error is

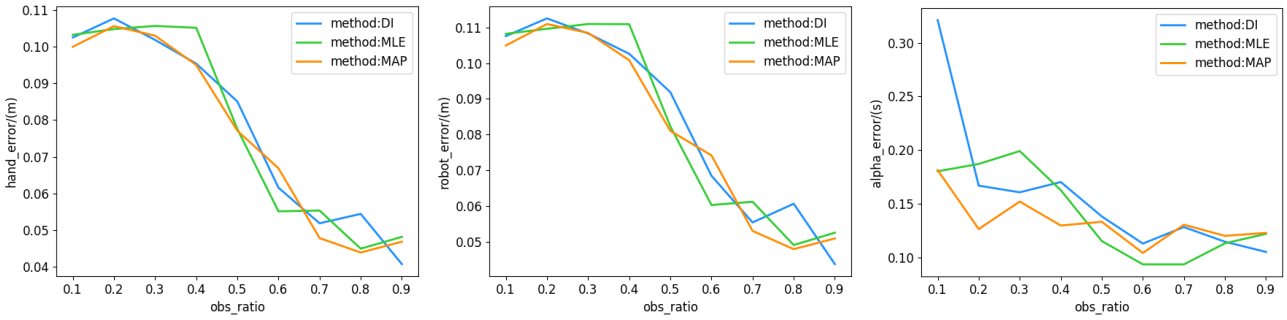


Figure 8: Prediction error of the "*handover_screw*" task, with (a)hand position prediction error, (b)robot position prediction error, and (c)phase estimation error.

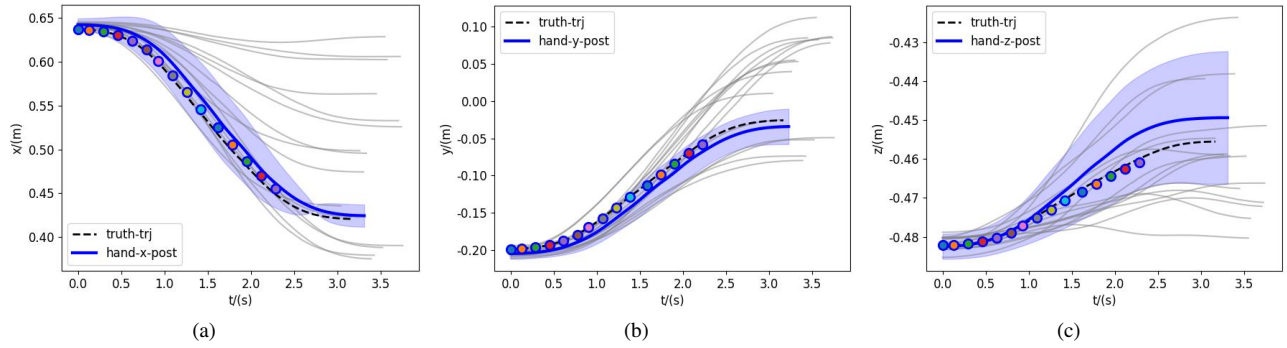


Figure 9: Predicted trajectory of the "*handover_screw*" task in (a)X, (b)Y and (c)Z axis.

acceptable because human hands are adaptable to such an error. The result indicates that for the 10^{th}, 40^{th} and 70^{th} trajectory, the prediction success rate decreases with the observation ratio. But for the 90^{th} trajectory, the prediction success rate always stays above 80%.

The results in Figure 10 indicate that correctly given environment parameters will ensure low prediction errors and high success rates. The results also imply that when $obs\_ratio < 0.2$, EIProMPs achieve a high success rate (above 80%) regardless of whether the actual environmental parameters are correctly given. For this reason, we choose $obs\_ratio < 0.2$ as a proper observation ratio in the following experiments. This also ensures the robot's reaction in advance before it observes the complete human motion.

5.2.2. Comparison of different phase models

Our demonstration trajectory set contains multiple task instances with different environmental parameters. Meanwhile, a single task may contain repeated motions with similar durations. Therefore, the Gaussian or uniform assumption on the phase variable distribution is violated. To explicitly illustrate the problem, we plot the phase dis-

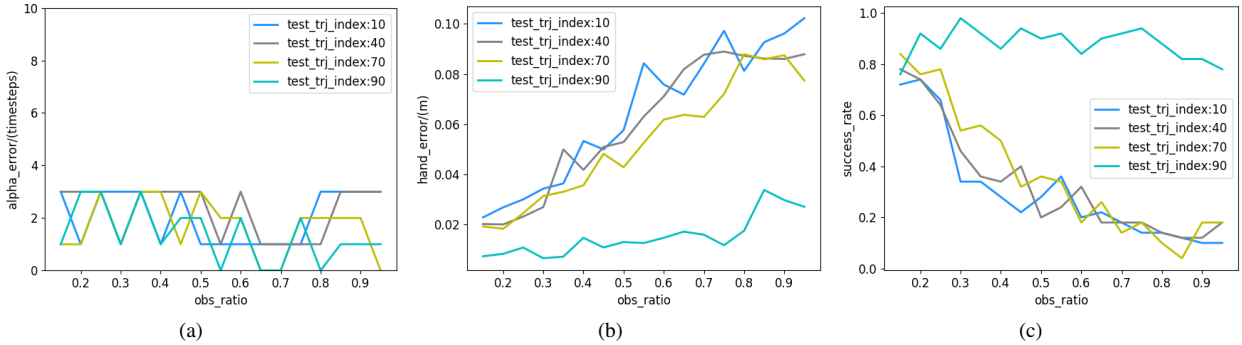


Figure 10: Testing EIProMP using 10^{th}, 40^{th}, 70^{th} and 90^{th} demonstration trajectories at different observation ratio. (a)Phase estimation error, (b)Hand position prediction error, (c)success rate

tribution of our demonstration trajectories in Figure 11. We further compute the KL divergence between the phase distribution and a standard Gaussian or uniform distribution, which is 0.18813 or 0.52992. These cues verify the non-Gaussian and non-uniform nature of the phase distribution of our demonstration trajectories.

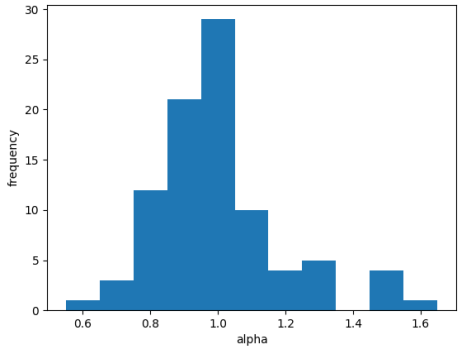


Figure 11: Phase distribution of the trajectory samples involved in our experiments.

In contrast to the temporal models in IProMP[26, 24], one of the EIProMP's benefits is the capability of alleviating the restriction of their phase variable satisfying Gaussian or uniform distribution. To verify that, we compare the three existing phase estimation methods with our Single-axis Uniform Interval Interpolation method in phase estimation accuracy. Note that the MAP, DI and MLE methods mentioned in Section 5.1 essentially predict the temporal length of a human movement according to partial observations, but the Single-axis Uniform Interval Interpolation method estimates the proportion of the predicted endpoint in relation to an entire reference trajectory. Therefore, we compute $1 - \alpha^*/\alpha_{groundtruth}$ for the MAP, DI, and MLE methods, so that the four methods can be evaluated by a unified and normalized metric. According to this metric, the results in Figure 12 and Figure 13 indicate that the Single-axis Uniform Interval Interpolation method achieves lower phase estimation errors than the MAP, DI, and MLE methods. In addition, the left part of Figure 14 demonstrates that a large phase estimation error in the MAP, DI and MLE methods will bring about a large delay, whilst the robot is supposed to catch up with the human partner after it has collected the required observations.

5.2.3. Effect of considering environmental parameters

Figure 15 shows the comparison of EIProMP (blue lines) and IProMP (green lines) under the condition of $w = 0.21$m and $h = 0.259$m being the given environmental parameters. We performed 10 independent trials of this experiment, and therefore the variance of using EIProMP (light blue) and IProMP (light green) are also illustrated in Figure 15.

We took the first 20% of the 70^{th} trajectory as the human motion observation and compared the human hand motion prediction error and the prediction success rate of EIProMP and IProMP in Figure 15(a) and (b), respectively. This corresponds to the situation that the actual environmental parameter during the robot's task replay is different from the given one. That explains why the hand motion prediction error of EIProMPs is higher than that of IProMPs, though both errors are below 0.05m. Nevertheless, the prediction success rate of EIProMPs is significantly higher than that of IProMPs, which indicates that EIProMPs can adapt better to unexpected environmental parameters. This result is acceptable in practice: If a human is acting according to an environmental parameter different from the given one, we would prefer successful human-robot collaboration to highly accurate human motion predictions. Besides, the hand

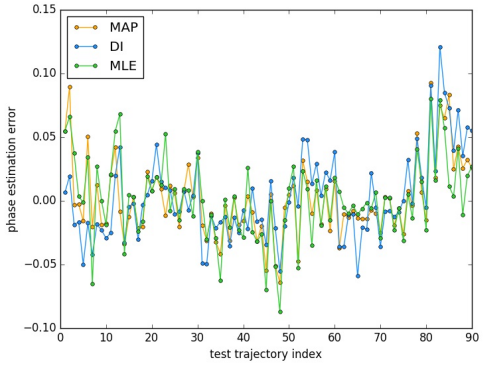


Figure 12: Phase estimation result using MAP, MLE and DI methods

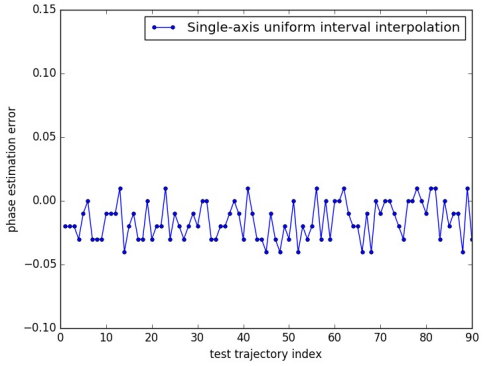


Figure 13: Phase estimation result using Single-axis Uniform Interval Interpolation method

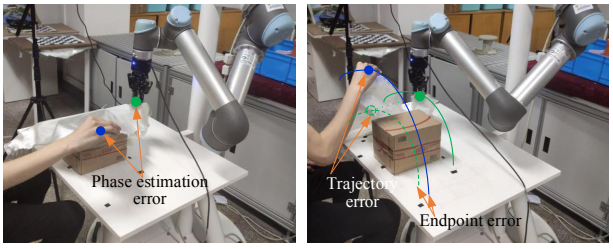


Figure 14: Errors in phase estimation (left) and human hand prediction (right)

prediction error of EIProMPs increases, and the success rate decreases with longer observation. The reason is also that the actual environmental parameter mismatches the given one.

Moreover, we took the first 20% of the 90^{th} trajectory as the human motion observation, and the results are shown in Figure 15(c) and (d). This indicates the situation that the actual environmental parameter aligns well with the given one. In this situation, IProMP and EIProMP achieve similar performance at high observation ratios, but EIProMP produces lower human motion prediction error and higher prediction success rate at lower observation ratios. This implies that the EIProMP method will ensure quicker proactive robot movements when only a few human trajectory points are observed.

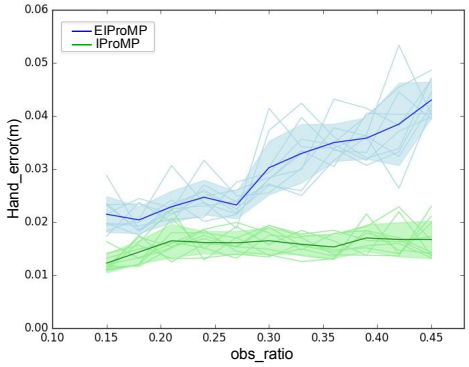
5.2.4. Predicted trajectories

We further observe the generalization performance of EIProMPs in trajectory prediction. Three untrained new environmental parameters, $\mathbf{e}_1 = \{w = 0.1\text{m}, h = 0.1\text{m}\}$, $\mathbf{e}_2 = \{w = 0.2\text{m}, h = 0.2\text{m}\}$ and $\mathbf{e}_3 = \{w = 0.3\text{m}, h = 0.3\text{m}\}$, were respectively input to the algorithm as the given environmental parameter. Meannwhile, the 45^{th} trajectory was selected as the testing data, with its first 20% treated as the observation.

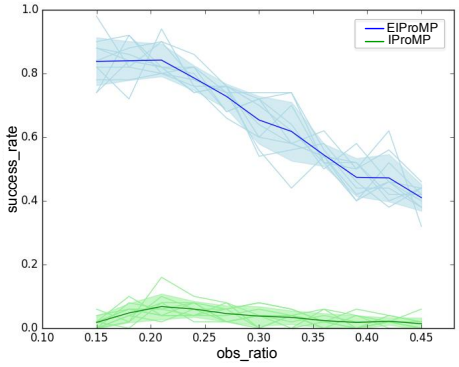
Figure 16 shows the predicted trajectories in the Y-Z plane, where the color dots are the observed human trajectory points, the grey curves are the training samples and the blue curves are the predicted human trajectories with their confidence bounds. Figure 16(a), (b) and (c) correspond to the new given environmental parameters $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$, respectively. In Figure 16, we also plot the given environmental parameters in form of blue rectangles in the Y-Z plane. Moreover, Table 2 shows the details of the prediction success rates.

According to the results, as the environmental parameter $\mathbf{e}_2$ is the closest to that of the 45^{th} trajectory, the corresponding prediction success rate in Table 2 is the highest, and the predicted trajectory in Figure 16 (b) fits the observed trajectory. In contrast, as $\mathbf{e}_1$ and $\mathbf{e}_3$ are quite different from that of the 45^{th} trajectory, the predicted trajectories in Figure 16 (a) and (c) have large variance compared with the observations. The right part of Figure 14 as well as Figure 16 (a) indicates the situations when $\mathbf{e}_1$ is given: Since $\mathbf{e}_1$ is lower than the ground truth, the robot generated lower trajectory than the human hand. It's also interesting to discover that the case of $\mathbf{e}_1$ produced a higher prediction success rate than $\mathbf{e}_3$, because $\mathbf{e}_1$ is more similar to the environmental parameters in the training trajectories while $\mathbf{e}_3$ is quite different from them.

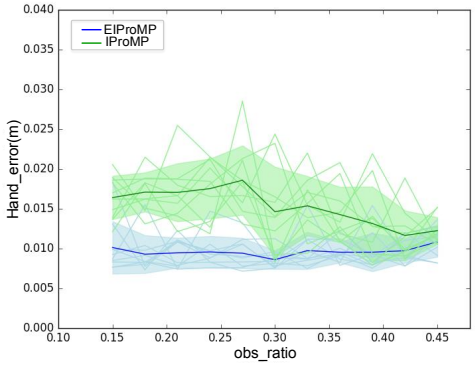
The results indicate threefold meanings. First, although the three environmental parameters were all "unseen", the algorithm can modulate proper and safe trajectories across the given obstacle. This indicates that EIProMPs possess a high generalization capability against the bias between the training and testing environmental parameters. Second, for EIProMPs the predicted trajectories are more "royal" to



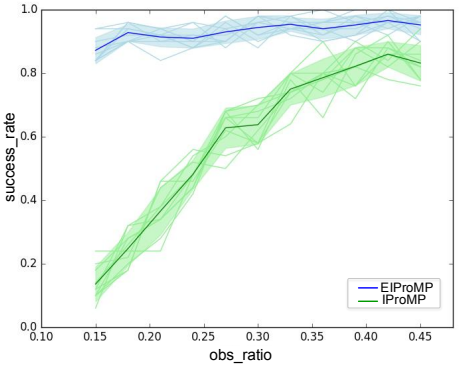
(a)



(b)



(c)



(d)

Figure 15: Comparison between EIProMP and IProMP

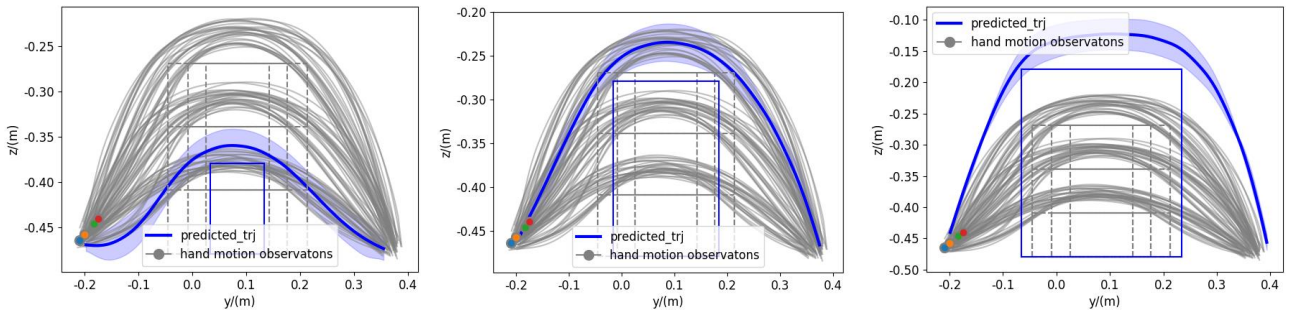


Figure 16: Generalization capability of EIProMP with untrained and different environment parameters: (a)$\mathbf{e}_1$, (b)$\mathbf{e}_2$ and (c)$\mathbf{e}_3$. For (b), the predicted human trajectory well fits the obervsation because the actual environmental parameter is close to the given one.

Table 2
Generalization performance of EIProMP

New environmental parameters	Mean hand prediction error/(m)	Phase estimation error (time steps)	Prediction Success Rate
$\mathbf{e}_1$	0.026831	2	82%
$\mathbf{e}_2$	0.038444	2	86%
$\mathbf{e}_3$	0.033916	2	48%

the given environmental parameters than to the observations of human hands, no matter whether the environmental parameters are correctly given. This is because safety (obstacle avoidance) takes priority over human compliance for EIProMPs. It's also noteworthy that although the environmental parameters have a greater influence on the predicted trajectory, the observed human trajectories also have their impact - The observed trajectory will enforce the predicted trajectory to gradually deform toward it, according to Figure 16.

5.3. Human-robot collaboration experiment

The *PickAssist* system entails the use of a UR5 robot, equipped with a Robotiq 2-fingers gripper and a Kinect camera. To support 6DoF and task-specific grasp using single-view point clouds, the grasp pose detection function of *PickAssist* is built on a Task-Constrained Grasp Pose Detection (TC-GPD) approach[36]. We follow the definition of part affordance labels [31] that includes seven different types of part affordance: "grasp", "cut", "scoop", "contain", "pound", "support" and "wrap-grasp". In particular, for the two-fingered parallel-jawed grippers of UR5 robots, we associate the "grasp" affordance to the objects involved in our application scenarios. Examples of the task-constrained grasp pose detection and the result of a handover task are shown in Figure 17 and Figure 18, respectively. More details of the task-constrained grasp pose detection and experimental settings can be found in our previous work [36, 15, 21]. Then the *PickAssist* system was tested in two scenarios.

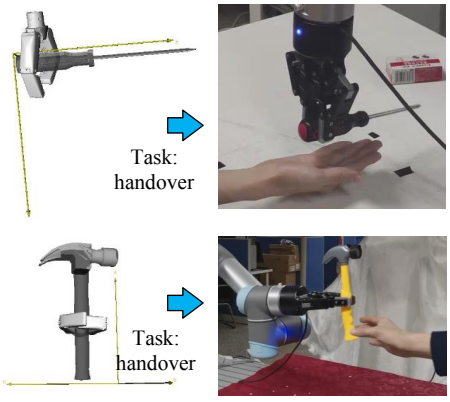


Figure 17: Task-constrained grasping

The first testing scenario is robot-assistive push-button assembly. A human partner expects the robot to proactively pass a screwdriver and a box containing screws to him, respectively, and take back the screwdriver from his hand after the assembly task is completed. This scenario contains two manipulation tasks: "Object Handover To Human" (OHTH) and "Object Handover To Robot"(OHTR). These tasks were further instantiated as "OHTH_screwdriver", "OHTH_screw" and "OHTR_screwdriver", as explained in Table 3. Given the six basic and predefined subtasks listed in Table 4, the OHTH task can be decomposed into a sequence of "PUO->HO->CO-R", and the OHTR task can be decomposed as "FO->PD->CO-R". It's worth mentioning that although the current implementation is based on predefined subtasks decomposition, automatic activity segmentation can be seamlessly integrated into our framework.

Figure 19 shows the scenario with the definition of coordinate systems. The human partner was asked to stretch his arm to reach for an object, and the proposed method enabled the robot to react to the human partner before his movement ends. During the online task replay phase, all the OHTH and the OHTR tasks were instantiated with the actual target object as their input. These task instances were then executed according to the corresponding subtask sequences.

The proposed method inherits the advantages of the

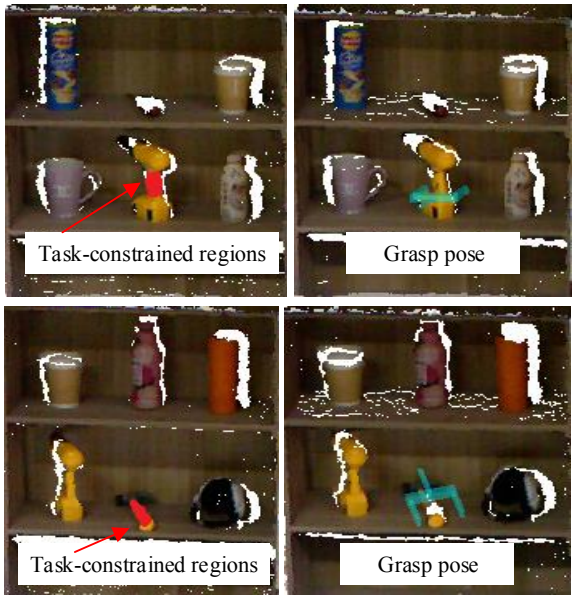


Figure 18: Examples of task-constrained grasp pose detection

Table 3
Task instances in the robot assistive push-button assembly scenario

Task instance	Descriptions
OHTH_screwdriver	robot handovers the screwdriver to the human
OHTH_screw	robot handovers the screw box to the human
OHTR_screwdriver	robot receives the screwdriver from the human partner

Table 4
Definition of subtasks

Subtasks	Abbr.
Pick Up Object	PUO
Handover Object	HO
Fetch Object	FO
Put object Down	PD
Object Covering using a piece of cloth	OC
COoperation-Release	CO-R

Probabilistic Movement Primitive methods [26, 24] in adapting to human motion speed variations. To be more specific, the proposed phase estimation algorithm is independent of the duration of human movement. Nevertheless, in the real-world experiment, we assumed a 2.2*sec* observation of human hand motion is sufficient for the prediction, according to the average duration time of the training samples. Because otherwise, the system would not be able to decide when the observation is sufficient for starting the robot's reaction. We experimentally found that the 2.2*sec* observation was sufficient for predicting the human trajectories in the handover tasks.

Figure 20 shows the generated robotic movements of the "OHTH_screwdriver" task. In the top left sub-figure,

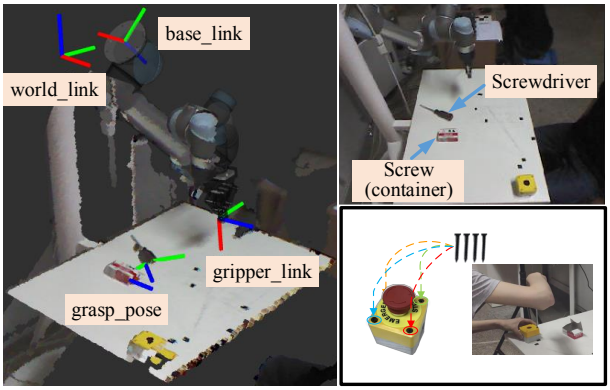


Figure 19: Configuration of the robot assistive push-button assembly scenario

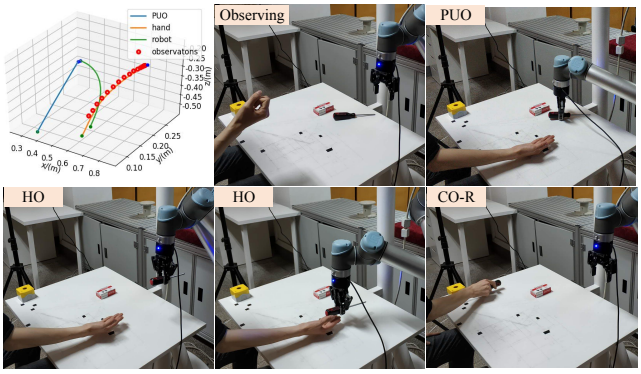


Figure 20: Result of executing the "OHTH_screwdriver" task instance

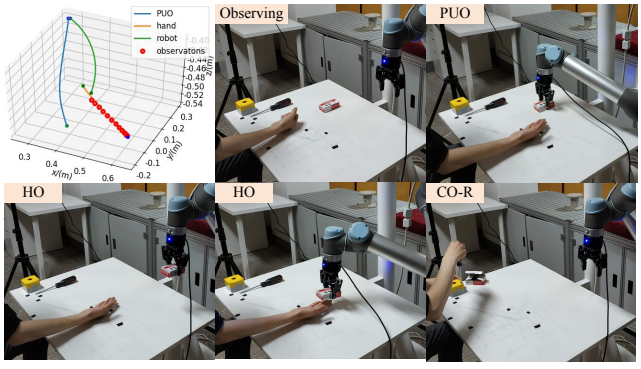


Figure 21: Result of executing the "OHTH_screw" task instance

PUO stands for picking up the screwdriver on the desk. The red circles are the observed human hand motion of reaching for a screwdriver. The predicted human hand movement of reaching for a screwdriver is shown as the orange trajectory and the generated robot's reactive movement of passing the screwdriver is shown as the green trajectory. Note that the robot adapted its movement to reach the goal according to the predicted human hand movement. Other sub-figures

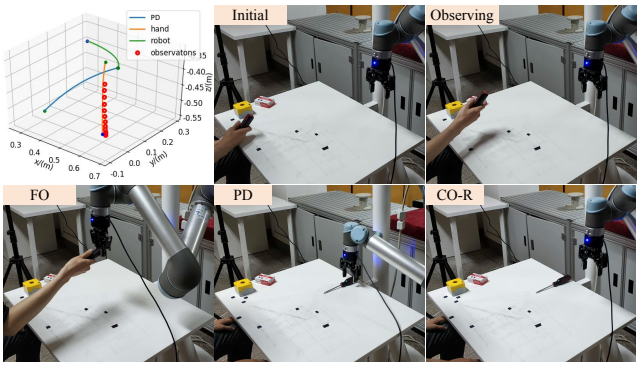


Figure 22: Result of executing the "OHTR_screwdriver" task instance

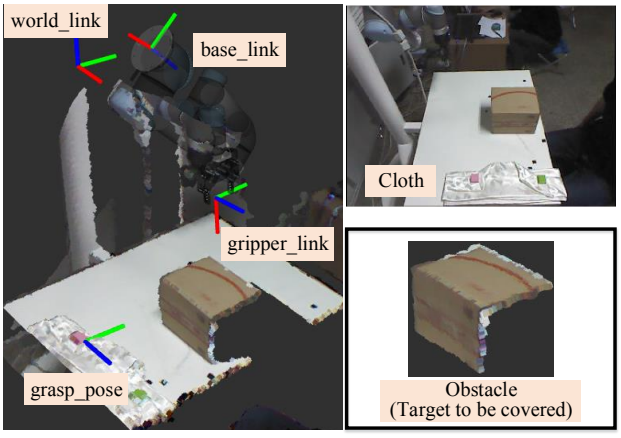


Figure 23: Configuration of the human-robot collaborative object covering scenario

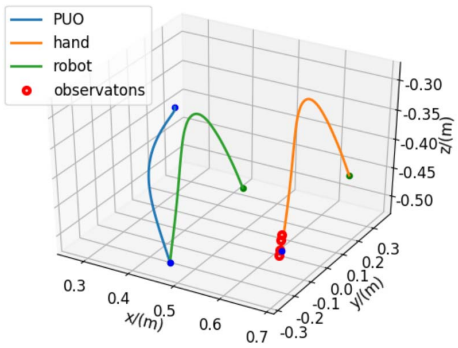


Figure 24: Motion trajectory of the human-robot collaborative object covering scenario

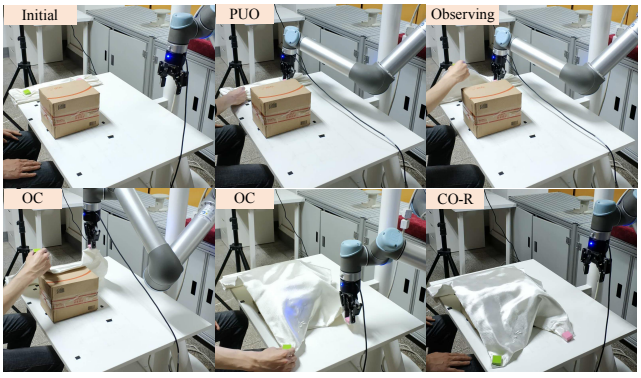


Figure 25: Result of the human-robot collaborative object covering scenario

show the snapshot of a real UR5 robot's movements. Figure 21 and Figure 22 illustrate the corresponding results of the "OHTH_screw" task and the "OHTR_screwdriver" task.

The second scenario is human-robot collaborative object covering ("OCWC"), as shown in Figure 23. The "OCWC" task was also decomposed into a sequence of subtasks, "PUO->OC->CO-R". In the experiment, the robot firstly executed the PUO subtask to pick up a corner of a piece of cloth, and meanwhile, the human partner took hold of another corner. Then the human partner started to act as if he would cover a box and the robot tracked the human hand movement for 1 second, as indicated by the red circles in Figure 24. The reason why we captured 1 second's human motion is similar to what has been mentioned above, except that it ensures $obs\_ratio = 0.2$ in our experiments since we consider obstacles in the second scenario. With regard to the obstacle issue, the box is considered as both the target to be covered and an obstacle whose width and height are recognized as environmental parameters. The Kinect based visual perception system reported the geometric size of the box as $\{l = 0.21\text{m}, w = 0.16\text{m}, h = 0.16\text{m}\}$.

Figure 24 shows the human and robot's trajectory during the execution. Using the partial observation of the human's movement, the robot predicted the posterior human movements as shown by the orange trajectory in Figure 24. To complete the assistance, the system generated the robotic trajectory for execution by shifting the predicted human trajectory along the X, Y and Z direction with a predefined shift $\{-0.21\text{m}, -0.07\text{m}, -0.04\text{m}\}$, as shown by the green trajectory in Figure 24. Figure 25 shows a snapshot of the box covering the scenario. The result indicates that although the robot started the covering motion with a short delay, it sped up to catch up with the human's movement, and finally the human and robot finished the covering movement almost simultaneously. The result implies a successful human-robot collaboration in the object covering task as well as a proactive response by reacting to the human partner before his movement ends.

6. Conclusion

In this paper, we propose an Environment-adaptive Interactive Probabilistic Movement Primitive (EIProMP) that accommodates the capability of adapting robot behaviors to suit the variance in environment parameters in human-robot collaboration tasks. Compared with previous methods, we propose a new phase estimation algorithm called Single-axis Uniform Interval Interpolation, which alleviates the

restriction of Gaussian or uniform distribution of the phase variable for the multiple joint demonstrations. Experimental results in the joint task of push-button assembly and object covering with a UR5 robot show increased robustness to ambiguity in partner activity as well as environmental changes. According to the results, the EIProMP achieves favorable generalization performance under the circumstances of unseen environmental parameters. It also ensures a higher success rate of human-robot collaboration and safe robot motions, compared with conventional methods that ignore environmental information. In the future, we aim to develop this framework to leverage more visual context of human partners and objects being manipulated for more complex interactive activities.

References

[1] Ben Amor, H., Campbell, J., 2017. Bayesian interaction primitives: A slam approach to human-robot interaction, in: Conference on Robot Learning, pp. 1–9.

[2] Ben Amor, H., Neumann, G., Kamthe, S., Kroemer, O., Peters, J., 2014. Interaction primitives for human-robot cooperation tasks, in: Proceedings - IEEE International Conference on Robotics and Automation, pp. 2831 – 2837.

[3] Caccavale, R., Saveriano, M., Finzi, A., Lee, D., 2019. Kinesthetic teaching and attentional supervision of structured tasks in human-robot interaction. Autonomous Robots 43, 1291 – 1307.

[4] Campbell, J., Stepputtis, S., Amor, H.B., 2019. Probabilistic multimodal modeling for human-robot interaction tasks. arXiv:1908.04955.

[5] Cui, Y., Poon, J., Matsubara, T., Miro, J.V., Sugimoto, K., Yamazaki, K., 2016. Environment-adaptive interaction primitives for human-robot motor skill learning, in: IEEE-RAS International Conference on Humanoid Robots, pp. 711 – 717.

[6] Cui, Y., Poon, J., Valls Miro, J., Yamazaki, K., Sugimoto, K., Matsubara, T., 2018. Environment-adaptive interaction primitives through visual context for human–robot motor skill learning. Autonomous Robots 43(8), 1225–1240.

[7] Duque, D.A., Prieto, F.A., Hoyos, J.G., 2019. Trajectory generation for robotic assembly operations using learning by demonstration. Robotics and Computer-Integrated Manufacturing 57, 292 – 302.

[8] Ewerton, M., Neumann, G., Lioutikov, R., Amor, H.B., Peters, J., Maeda, G., 2015. Learning multiple collaborative tasks with a mixture of interaction primitives, in: Proceedings - IEEE International Conference on Robotics and Automation, pp. 1535 – 1542.

[9] Fernando, T., Denman, S., Sridharan, S., Fookes, C., 2021. Deep inverse reinforcement learning for behavior prediction in autonomous driving: Accurate forecasts of vehicle motion. IEEE Signal Processing Magazine 38, 87–96.

[10] Fu, J., Luo, K., Levine, S., 2018. Learning robust rewards with adversarial inverse reinforcement learning, in: Int. Conf. Learning Representation, (ICLR), pp. 1–15.

[11] Gu, Y., Sheng, W.H., Crick, C., Ou, Y.S., 2018. Automated assembly skill acquisition and implementation through human demonstration. Robotics and Autonomous Systems 99, 1–16.

[12] de Haan, P., Jayaraman, D., Levine, S., 2019. Causal confusion in imitation learning, in: Advances in Neural Information Processing Systems.

[13] Hu, J., Xiong, R., 2019. Trajectory generation with multi-stage cost functions learned from demonstrations. Robotics and Autonomous Systems 117, 57 – 67.

[14] Janner, M., Fu, J., Zhang, M., Levine, S., 2019. When to trust your model: Model-based policy optimization, in: Advances in Neural Information Processing Systems.

[15] Jing, X., Qian, K., Xu, X., Bai, J., Zhou, B., 2021. Domain adversarial transfer for cross-domain and task-constrained grasp pose detection. Robotics and Autonomous Systems 145, 103872.

[16] Joshi, R.P., Koganti, N., Shibata, T., 2019. A framework for robotic clothing assistance by imitation learning. Advanced Robotics https://doi.org/10.1080/01691864.2019.1636715.

[17] Kuang, Y., Cheng, H., Zheng, Y., Cui, F., Huang, R., 2019. One-shot gesture recognition with attention-based dtw for human-robot collaboration. Assembly Automation 40, 40 – 47.

[18] Kulvicius, T., Biehl, M., Aein, M.J., Tamosiunaite, M., Worgotter, F., 2013. Interaction learning for dynamic movement primitives used in cooperative robotic tasks. Robotics and Autonomous Systems 61, 1450 – 1459.

[19] Li, Z., Zhao, T., Chen, F., Hu, Y., Su, C.Y., Fukuda, T., 2018. Reinforcement learning of manipulation and grasping using dynamical movement primitives for a humanoidlike mobile manipulator. IEEE/ASME Transactions on Mechatronics 23, 121–131.

[20] Lioutikov, R., Maeda, G., Veiga, F., Kersting, K., Peters, J., 2020. Learning attribute grammars for movement primitive sequencing. International Journal of Robotics Research 39, 21 – 38.

[21] Liu, H., 2020. Movement primitives based robot learning by demonstration and its applications. Southeast University.

[22] Liu, H., Qian, K., Gui, B., Ma, X., 2019. Task generalization of robots based on parameterized learning of multi-demonstration action primitives. Robot 41, 574 – 582.

[23] Maeda, G., Ewerton, M., Lioutikov, R., Ben Amor, H., Peters, J., Neumann, G., 2014. Learning interaction for collaborative tasks with probabilistic movement primitives, in: IEEE-RAS International Conference on Humanoid Robots, pp. 527 – 534.

[24] Maeda, G., Ewerton, M., Neumann, G., Lioutikov, R., Peters, J., 2017a. Phase estimation for fast action recognition and trajectory generation in human-robot collaboration. International Journal of Robotics Research 36, 1579 – 1594.

[25] Maeda, G., Neumann, G., Ewerton, M., Lioutikov, R., Peters, J., 2018. A Probabilistic Framework for Semi-autonomous Robots Based on Interaction Primitives with Phase Estimation. Springer International Publishing, Cham. pp. 253–268.

[26] Maeda, G.J., Neumann, G., Ewerton, M., Lioutikov, R., Kroemer, O., Peters, J., 2017b. Probabilistic movement primitives for coordination of multiple human-robot collaborative tasks. Autonomous Robots 41, 593 – 612.

[27] Meziane, R., Otis, M.J.D., Ezzaidi, H., 2017. Human-robot collaboration while sharing production activities in dynamic environment: Spader system. Robotics and Computer-Integrated Manufacturing 48, 243 – 253.

[28] Michalos, G., Kousi, N., Karagiannis, P., Gkournelos, C., Dimoulas, K., Koukas, S., Mparis, K., Papavasileiou, A., Makris, S., 2018. Seamless human robot collaborative assembly – an automotive case study. Mechatronics 55, 194–211.

[29] Mizera, C., Delrieu, T., Weistroffer, V., Andriot, C., Decatoire, A., Gazeau, J.P., 2020. Evaluation of hand-tracking systems in teleoperation and virtual dexterous manipulation. IEEE Sensors Journal 20, 1642 – 1655.

[30] Muller, D., Veil, C., Seidel, M., Sawodny, O., 2020. One-shot kinesthetic programming by demonstration for soft collaborative robots. Mechatronics 70, 102418.

[31] Myers, A., Teo, C.L., Fermüller, C., Aloimonos, Y., 2015. Affordance detection of tool parts from geometric features, in: 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 1374 – 1381.

[32] Nishi, K., Shimosaka, M., 2020. Fine-grained driving behavior prediction via context-aware multi-task inverse reinforcement learning, in: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 2281–2287.

[33] Paraschos, A., Daniel, C., Peters, J., Neumann, G., 2013. Probabilistic movement primitive, in: Advances in Neural Information Processing Systems (NIPS), p. 2616–2624.

[34] Pellegrinelli, S., Moro, F.L., Pedrocchi, N., Molinari Tosatti, L.,

Tolio, T., 2016. A probabilistic approach to workspace sharing for human-robot cooperation in assembly tasks. CIRP Annals 65, 57 – 60.

[35] Pervez, A., Lee, D., 2018. Learning task-parameterized dynamic movement primitives using mixture of gmms. Intelligent Service Robotics 11(1), 61–78.

[36] Qian, K., Jing, X., Duan, Y., Zhou, B., Fang, F., Xia, J., Ma, X., 2020a. Grasp pose detection with affordance-based task constraint learning in single-view point clouds. Journal of Intelligent and Robotic Systems: Theory and Applications 100, 145 – 163.

[37] Qian, K., Liu, H., Valls Miro, J., Jing, X., Zhou, B., 2020b. Hierarchical and parameterized learning of pick-and-place manipulation from under-specified human demonstrations. Advanced Robotics 34, 858 – 872.

[38] Rahmatizadeh, R., Abolghasemi, P., Behal, A., Bölöni, L., 2017. Learning real manipulation tasks from virtual demonstrations using lstm. arXiv:1603.03833.

[39] Ross, S., Gordon, G., Bagnell, D., 2011. A reduction of imitation learning and structured prediction to no-regret online learning, in: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, pp. 627–635.

[40] Sakr, M., Freeman, M., MacHiel, H., Loos, V.D., Croft, E., 2020. Training human teacher to improve robot learning from demonstration: A pilot study on kinesthetic teaching, in: 29th IEEE International Conference on Robot and Human Interactive Communication, RO-MAN 2020, pp. 800 – 806.

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

[42] Sena, A., Howard, M., 2020. Quantifying teaching behavior in robot learning from demonstration. International Journal of Robotics Research 39, 54–72.

[43] Shukla, D., Erkent, O., Piater, J., 2017. Proactive, incremental learning of gesture-action associations for human-robot collaboration. RO-MAN 2017 - 26th IEEE International Symposium on Robot and Human Interactive Communication 2017-January, 346 – 353.

[44] Tsarouchi, P., Matthaiakis, A.S., Makris, S., Chryssolouris, G., 2017. On a human-robot collaboration in an assembly cell. International Journal of Computer Integrated Manufacturing 30, 580–589.

[45] Vogt, D., Stepputtis, S., Weinhold, R., Jung, B., Ben Amor, H., 2016. Learning human-robot interactions from human-human demonstrations (with applications in lego rocket assembly), in: IEEE-RAS International Conference on Humanoid Robots, pp. 142 – 143.

[46] Wang, L., Gao, R., Váncza, J., Krüger, J., Wang, X., Makris, S., Chryssolouris, G., 2019. Symbiotic human-robot collaborative assembly. CIRP Annals 68, 701–726.

[47] Wang, L., Liu, S., Liu, H., Wang, X.V., 2020. Overview of human-robot collaboration in manufacturing, in: Proceedings of 5th International Conference on the Industry 4.0 Model for Advanced Manufacturing, pp. 15 – 58.

[48] Wang, X.V., Kemeny, Z., Vancza, J., Wang, L., 2017. Human-robot collaborative assembly in cyber-physical production: Classification framework and implementation. CIRP Annals 66, 5 – 8.

[49] Xu, J., Qian, K., Liu, H., Ma, X., 2018. Hand pose estimation for robot programming by demonstration in object manipulation tasks, in: Chinese Control Conference, CCC, pp. 5328 – 5333.

[50] Xu, X., Qian, K., Zhou, B., Chen, S., Li, Y., 2021. Two-stream 2d/3d residual networks for learning robot manipulations from human demonstration videos, in: 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 3353 – 3358.

[51] Zaatari, S.E., Wang, Y., Li, W., Peng, Y., 2021. itp-lfd: Improved task parametrised learning from demonstration for adaptive path generation of cobot. Robotics and Computer-Integrated Manufacturing 69.

[52] Zhu, Z., Hu, H., 2018. Robot learning from demonstration in robotic assembly: A survey. Robotics 7(18), 1–25.