Learning to Assist Bimanual Teleoperation using Interval Type-2 Polynomial Fuzzy Inference

Ziwei Wang, Member, IEEE, Haolin Fei, Yanpei Huang, Member, IEEE, Quentin Rouxel, Bo Xiao, Member, IEEE, Zhibin Li, Member, IEEE, and Etienne Burdet, Member, IEEE

Abstract—Assisting humans in collaborative tasks is a promising application for robots, however effective assistance remains challenging. In this paper, we propose a method for providing intuitive robotic assistance based on learning from human natural limb coordination. To encode coupling between multiple-limb motions, we use a novel interval type-2 (IT2) polynomial fuzzy inference for modeling trajectory adaptation. The associated polynomial coefficients are estimated using a modified recursive least-square with a dynamic forgetting factor. We propose to employ a Gaussian process to produce robust human motion predictions, and thus address the uncertainty and measurement noise of the system caused by interactive environments. Experimental results on two types of interaction tasks demonstrate the effectiveness of this approach, which achieves high accuracy in predicting assistive limb motion and enables humans to perform bimanual tasks using only one limb.

Index Terms—Human-robot collaboration, Bimanual manipulation, IT2 polynomial fuzzy system, Robot learning, Gaussian process.

I. INTRODUCTION

ROBOTS can assist humans and potentially alleviate their workload during collaborative tasks, while the human user typically contributes to the adaptation of trajectories and efforts. Effective human-robot coordination is crucial to ensure that the interaction is transparent to the user. A supervised teleoperation scheme can allow the robot to follow human commands [1]–[3]. Similarly, the robotic agent of [4] assists in trajectory training for novices based on task-dependent reinforcement learning. From a cognitive perspective, robots could be made more flexible and better interact with human users by learning from human movements using logical inference. Recent works have emphasized the importance of the robot to generate commands based on the prediction of the user’s movements [5]–[7]. Different from transfer learning [8]–[10], learning from demonstration [11]–[14] or human posture prediction in physical human-robot interaction [15], our idea is to generate human-robot assistive control based on the coordination observed in bimanual control or in the collaborative control between multiple humans.

Intrinsic nonlinearity and uncertainty of human motion pose great challenges to developing the aforementioned coordination models. Transferring cognitive processes observed in human brain to robotic applications may be used to generate behaviors that are robust to changes in the environment [16]. Different from neural networks [17] and regression models [18] that rely on an arbitrary system structure, fuzzy systems can represent nonlinear plants starting from human knowledge, which can reduce the computational complexity and the number of required parameters [19]. Takagi-Sugeno fuzzy model based (T-S FMB) systems realize nonlinear approximation capability with arbitrary accuracy through a weighted sum of linear subsystems [20]. As an effective extension, the polynomial fuzzy model relaxes the restrictions on the argument domain, allows the system matrices to contain polynomials, and therefore can express nonlinearities well [21].

In terms of T-S and polynomial FMB (PFMB) identification, most reported methods obtain fuzzy sets and fuzzy rules by clustering methods and perform linear identification operations with least-square-like under each rule [22], [23]. The above algorithms leveraged the first order of input data and inter-data relationship, i.e. the input data are decoupled. However, the limb motion in each dimension contributes to the other in a coupled way, thus the above approaches might miss the intra-data relationship.

Moreover, they are not designed to deal with parameter uncertainty in the membership function induced by human motion, where measurement errors may result in vibrations in the membership function and thereby degrade the modeling accuracy. The interval type-2 (IT2) fuzzy set is a promising tool to handle such uncertainties, captured within the footprint of uncertainty (FOU) formed by the upper and lower membership functions (i.e., UMF and LMF). In [24], [25], the UMF and LMF are constructed by adjusting the fuzzy partition matrix exponent in terms of fuzzy c-means (FCM) clustering, which is beneficial to address parameter uncertainty in system identification. The FOU is obtained through the heuristic-based and histogram-based methods by scaling the parameters in predefined type-1 membership functions [26]. However, the above methods for constructing IT2 fuzzy sets can only handle the parameter uncertainty within a predefined range, which...
relies on the prior knowledge of system uncertainty.

To address the intrinsic nonlinearity and uncertainty of human motion for human-robot coordination, this paper introduces a Gaussian process (GP) based IT2 polynomial fuzzy learning (GP-IT2PFL) algorithm, which enables humans to perform the bimanual task only by one limb. The proposed method aims to attain a trajectory-level shared control in teleoperation by utilizing a learned policy that is anticipated by the human operator. Based on the modified FCM clustering with z-score normalization, the entire task is classified into fuzzy sets yielding corresponding membership functions. Mapping nonlinearity, parameter uncertainty, and measurement noise can be handled effectively, where the FOU is constructed by a GP. The innovations of the GP-IT2PFL algorithm are as follows:

(1) Cross-polynomial and solo-polynomial matrices are introduced to reflect the intrinsic coupling of input data with different features and dimensions, enabling to represent the intra- and inter-class features in human limbs coordination.

(2) The UMF and LMF are developed by the posterior distribution predicted through a non-parametric GP, which facilitates the capture of parameter perturbations due to measurement noise and system uncertainty. Therefore, the GP-IT2PFL algorithm is featured data-driven and does not depend on the identification structure.

The paper first describes an IT2 polynomial fuzzy learning method with a GP-based membership function, which aims at generating human-robot assistive control. Then we tested the GP-IT2PFL algorithm in two human-robot collaboration scenarios. First, a foot-robot coordination experiment was carried out for grasping, picking and placing tasks. We recorded hand-foot coordination while performing these tasks. The GP-IT2PFL algorithm then was used to generate hand-like trajectories for foot-foot coordination to assist foot behaviors thus freeing up the hands for direct manipulation. Second, we recorded the movements of two collaborating humans in a human-robot shared telesurgery experiment, then modeled one human as a robot partner, where the robot trajectory was generated by the GP-IT2PFL algorithm that assisted humans in a peg-transfer task. These applications validate the developed algorithms and illustrate some of the opportunities they open.

II. METHODS

A. Mathematical Notation

\[ Z^+ [a, b] \] refers to the set of natural numbers from \( a \) to \( b \). For \( x = [x_1, x_2, \ldots, x_n] \in \mathbb{R}^n \) and \( d \in Z^+ [1, +\infty] \), \( G_d(x_1) = [x_1^{d-1}, x_1^{d-2}, \ldots, 1]^\top \) and \( G_d(x) = \{G_d(x_1), G_d(x_2), \ldots, G_d(x_n)\} \in \mathbb{R}^{n \times d} \). \( \chi \) is the column vector whose \( x_i \) is the \( i \)-th entry. \( V_d(x) \in \mathbb{R}^{n \times d} \) denotes the Vandermonde matrix of degree \( d - 1 \). “\( \circ \)” is the Hadamard product. For \( y = [y_1, y_2, \ldots, y_n] \in \mathbb{R}^n \), \( P_{xy} = \sum_{i=1}^{n} V_d(x) G_d(y_i) \in \mathbb{R}^n \) is defined as the cross-polynomial matrix of a binary tuple \( x-y \). \( P_x = \sum_{i=1}^{n} V_d^T(x) G_d(x_i) \in \mathbb{R}^n \) is the solo-polynomial matrix of \( x \), in which \( V_d^T(x) \) is the Vandermonde matrix with all elements of row \( i \) being 1. “\( \vee \)” is the product operator for cross-polynomial matrix \( P_{xy} \) and block matrix \( M \) such that \( M \vee P_{xy} = \sum_{i=1}^{n} M_i \circ V_d(x_i) G_d(y_i) \in \mathbb{R}^n \), where \( M_i \) is the \( i \)-th block entry of \( M \). \( I \) is the identity matrix of the appropriate dimension.

B. Working Principle

In order to provide trajectory adaptation tailored to a human operator, we construct a unified IT2 polynomial fuzzy framework by utilizing kinesthetic data (i.e., the respective position, orientation and velocity) from two-limb coordination. This framework is developed through four steps: (i) firstly, we utilize an FCM clustering approach to carry out task segmentation, whereby each cluster denotes a distinct subtask; (ii) the resulting membership functions are generated through a Gaussian process, although they are dependent on kinesthetic data and subject to uncertainty arising from human motion; (iii) IT2 polynomial fuzzy model parameters with respect to each subtask are identified online by introducing cross-polynomial and solo-polynomial matrices; and (iv) the adaptation trajectory is produced through the integration of the membership grades and the corresponding fuzzy model. Fig. 1 illustrates how the GP-IT2PFL based inference engine can generate motion adaptation for hand-foot coordination.

Remark 1: The GP-IT2PFL framework can be used to learn human-robot collaboration from various types of multi-limb coordination. In the hand-foot coordination experiment of Section III the coordinated left-hand motion is learned to later automatically accompany the foot motion. In the experiment of Section IV, the two interfaces are controlled by two humans who collaborate on a shared task. During the training phase, the left-hand and foot motion commands are used to generate hand motion commands of the two individuals. In the testing phase, one robot is autonomously controlled by the GP-IT2PFL algorithm to assist the movement of one human’s arm. This is validated in the shared telesurgery experiment.

C. Algorithm Implementation

Denote the two-limb motion states at the training stage as \( [x_i, y_i]^N_{i=1} \), where the subscript \( i \) stands for the \( i \)-th sample and \( N \) the number of samples. Since the bimanual task usually consists of multiple segments, the coordination of the two-limb trajectories in each segment is difficult to describe with the same model. Therefore, we utilize the FCM clustering method to perform task segmentation. Since the potential multiple features are contained in \( [x_i]^N_{i=1} \), FCM clustering is modified to solve the following optimization problem using z-score normalization

\[
\min_{\omega} \sum_{i=1}^{N} \sum_{j=1}^{l} (\omega_j(x_i))^\chi \left\| \frac{x_i - \mu_x}{\sigma_x} - c_j \right\|^2 \tag{1}
\]

subject to \( \sum_{j=1}^{l} \omega_j(x_i) = 1 \) \tag{2}

where \( l \) is the number of fuzzy rules extracted by subtractive clustering [28], \( \mu_x \) and \( \sigma_x \) are the mean and standard derivation corresponding to \( [x_i]^N_{i=1} \), respectively. \( \chi > 1 \) is the fuzzy partition matrix exponent. \( C = \{c_1, \ldots, c_l\} \) stands for the set
of fuzzy set centers. The membership function with respect to the rule $j$ is denoted by $\omega_j(x_i)$. Taking the Lagrange multiplier $\lambda \in \mathbb{R}_+$ into the optimization problem (1), one can obtain

$$\mathcal{L} = \sum_{i=1}^{N} \sum_{j=1}^{l} (\omega_j(x_i))^n \left| \frac{x_i - \mu_x}{\sigma_x} - c_j \right|^2 + \lambda(1 - \sum_{j=1}^{l} \omega_j(x_i)).$$

(3)

First-order partial derivatives with respect to $\lambda$ and $\omega_j(x_i)$ are given by

$$\nabla_{\lambda, \omega_j(x_i)} \mathcal{L} = 0 \iff \left\{ \begin{array}{l} x_i^2 - \mu_x \sigma_x \omega_j(x_i)^n - c_j \omega_j(x_i)^n = 0, \\
\sum_{j=1}^{l} \omega_j(x_i) = 1. \end{array} \right.$$\n
(4)

Solving the above equation group yields the center of each fuzzy set as:

$$c_j = \frac{\sum_{i=1}^{N} (\omega_j(x_i))^n (x_i - \mu_x)}{\sigma_x \sum_{i=1}^{N} (\omega_j(x_i))^n},$$

(5)

and

$$\omega_j(x_i) = \begin{cases} 0 & \text{if } x_i = \mu_x + \sigma_x c_k \\
\frac{1}{\prod_{k=1}^{l} \left( \frac{x_i - \mu_x - \sigma_x c_k}{\sigma_x} \right)^{\omega_k}} & \text{if } x_i \neq \mu_x + \sigma_x c_k \\
\sum_{i=1}^{l} \omega_j(x_i) & \text{if } x_i = \mu_x + \sigma_x c_j \end{cases}$$

(6)

With the obtained fuzzy clusters (5) and the membership functions (6), the inherent relationship between the two-limb coordination trajectories can be represented by the following discrete-time polynomial fuzzy system for $j \in \mathbb{Z}^+[1, l]$

Rule $j$: IF $A_j(z_i)$ is $B_j$

THEN $y_i = M_j \langle z_i, a_i \rangle + C_j \langle z_i, v_i \rangle + K_j \langle z_i, z_i \rangle$

(7)

where $z_i = (x_i - \mu_x)/\sigma_x$ is the z-score normalized form of $x_i$, $a_i = (v_{i+1} - v_i)/\Delta T$, $v_i$ and $v_{i+1}$ represent the time-derivative of $z_i$ at $i$th and $(i + 1)$th sample, which can be obtained through Unscented Kalman filter (UKF) to overcome the high computational effort associated with the linearisation of complex motion models. $\Delta T$ is the sample time, $A_j(z_i)$ the premise variable, and $B_j$ the fuzzy set whose centre is $c_j$, $M_j$, $C_j$, and $K_j$ are the coefficient matrices of appropriate dimensions, respectively. $P_{z_i a_i}$ and $P_{z_i v_i}$ are the cross-polynomial system matrices of $\{z_i, a_i\}$ and $\{z_i, v_i\}$, in which the degree of polynomial matrices is $d_j - 1$. $\mathcal{P}_{z_i} \in \mathbb{R}^n$ is the solo-polynomial matrix of $z_i$.

**Remark 2:** Introducing the Vandermond matrix in cross-polynomial and solo-polynomial system matrices enables (7) to incorporate the contributions of each element within a single input $z_i$ and of the individual elements of binary input $\{(z_i, a_i)\}$ and $\{(z_i, v_i)\}$. Thus, solo-polynomial and cross-polynomial system matrices facilitate the representation of intra-class and inter-class relationships of $\{x_i\}_{i=1}^{N}$, respectively, exhibiting the intrinsic coupling of limb motion in terms of features and dimensions.

Given $a_i = \text{col}_{p=1}^{n} \{a_{ip}\} \in \mathbb{R}^n$ and $z_i = \text{col}_{p=1}^{n} \{z_{ip}\} \in \mathbb{R}^n$ for $p \in \mathbb{Z}^+[1, n]$, we have a more general form of (7) using the equivalent property of polynomial matrix

$$y_i = \mathcal{M}_j \langle z_i, a_i \rangle + \mathcal{C}_j \langle z_i, v_i \rangle + \mathcal{K}_j \langle z_i, z_i \rangle,$$

(8)

where the polynomial operator $\langle z_i, a_i \rangle$ for binary tuple with different features is defined as: $\langle z_i, a_i \rangle = \text{col}_{p=1}^{n} \{z_{ip}, a_{ip}\}$.
with
\[
\langle z_i, \alpha_p \rangle = \left[ \sum_{q=1}^{d_j} z_{i_q} \alpha_p - q \alpha_p^{-1} \right] \left[ \sum_{q=1}^{d_j} z_{i_q} \alpha_p - q \alpha_p^{-1} \right] \left[ \sum_{q=1}^{d_j} z_{i_q} \alpha_p - q \alpha_p^{-1} \right] \left[ \sum_{q=1}^{d_j} z_{i_q} \alpha_p - q \alpha_p^{-1} \right], \forall \alpha_i = \text{col}^n_{p=1} \{ \alpha_{ip} \} \in \mathbb{R}^n.
\]

Regarding the input with same features, then the polynomial operator becomes \( \langle z_i, z_{ip} \rangle = \text{col}^n_{p=1} \{ \langle z_i, z_{ip} \rangle \} \) with
\[
\langle z_i, z_{ip} \rangle = \left[ \sum_{q=1}^{d_j} z_{i_q} - q \alpha_p^{-1} \right] \left[ \sum_{q=1}^{d_j} z_{i_q} - q \alpha_p^{-1} \right] \left[ \sum_{q=1}^{d_j} z_{i_q} - q \alpha_p^{-1} \right] \left[ \sum_{q=1}^{d_j} z_{i_q} - q \alpha_p^{-1} \right], q \in \mathbb{Z}^+[1, d_j].
\]

It is worth pointing out that the \( p \)th element of \( \langle z_i, z_{ip} \rangle \) is a polynomial of \( z_{ip} \) and the remaining elements are polynomials of \( z_{iq} \) and \( z_{ip} \) for \( p, q \in \mathbb{Z}^+[1, n] \) and \( q \neq p \). Let the augmented variables be \( Z_i = (\langle z_i, \alpha_i \rangle, \langle z_i, v_i \rangle, \langle z_i, z_{i} \rangle)^T \) and \( H_i^2 = [M_j, C_j, K_j] \), a more compact form of \( \mathcal{R} \) can be derived by
\[
\tilde{y}_i^j = H_i^2 Z_i.
\]

The polynomial coefficients in \( \mathcal{R} \) for the \( j \)th rule can be estimated using the modified recursive least-square with dynamic forgetting factor (RLS-DFD):
\[
\tilde{H}_{i+1}^2 = \tilde{H}_i^2 + \tilde{y}_{i+1}^j + (P_i Z_{i+1})^T P_i Z_{i+1} \left( \frac{P_i Z_{i+1}}{\beta_{i+1}} + (Z_{i+1})^T P_i Z_{i+1} \right),
\]
\[
P_{i+1} = \frac{1}{\beta_{i+1}} \left( P_i - \frac{P_i Z_{i+1}(Z_{i+1})^T P_i}{\beta_{i+1} + (Z_{i+1})^T P_i Z_{i+1}} \right),
\]
\[
\beta_{i+1} = \beta_0 + (1 - \beta_0) \left( 1 - \tanh(\beta^*(\tilde{y}_{i+1}^j)^T \tilde{y}_{i+1}^j) \right),
\]
where \( \tilde{y}_{i+1}^j = y_{i+1}^j - \tilde{H}_i^2 Z_i \) is the innovation vector. \( \beta_0 \in (0, 1) \) is the time-varying forgetting factor, which dynamically moderates the effect of new data on parameter estimates according to the innovation. \( \beta_0 \in (0, 1) \) is a constant scalar initially defined for the forgetting factor. \( \beta^* \in \mathbb{R}^+ \) is a tuning parameter.

Considering the measurement noise in \( z_i \) and the system uncertainty, it is difficult to obtain the exact membership grades, which is prone to enter the FOU formed by the upper and lower membership functions. In order to capture the uncertainty in the membership function, we model it as a Gaussian process with a squared exponential kernel function
\[
\omega_j(z) \sim \mathcal{GP} (\mu_j(z), \kappa(z, z'))
\]
where \( \mu_j(z) \) and \( \kappa(z, z') \) are the prior mean function and the kernel function. Without loss of generality, each entry of the squared exponential kernel function is calculated as
\[
[\kappa(z, z')]_{ij} = \sigma^2 \exp \left( -\frac{\|z_p - z'_q\|^2}{2\sigma^2} \right)
\]
where \( z \in \mathbb{R}^{n \times N_p}, z' \in \mathbb{R}^{n \times N_p}, p \in \mathbb{Z}^+[1, N_p] \) and \( \nu \in \mathbb{Z}^+[1, N_p] \). \( \sigma_\kappa \) and \( \nu_\kappa \) are the optimal hyperparameters obtained by maximizing marginal log-likelihood. Let \( z^* \sim \omega_j(z^*) \) be the random sample (observations) from the data set \( \{z_i, \omega_j(z_i)\}_{i=1}^n \), the observations, therefore, follow a joint Gaussian distribution with the rest data (predictions)
\[
\Omega_j^* \sim \mathcal{GP} \left( \begin{bmatrix} \mu_{\omega_j}^* \kappa(\mu^*, z^*) \end{bmatrix}, \begin{bmatrix} \sigma^2 \kappa(\mu^*, z^*) \end{bmatrix} \right),
\]
in which \( \Omega_j^* = \text{col}^n_{\nu=1} \{\omega_j(z^*)\} \). \( \mathcal{N} = \{z^*_1, z^*_2, \ldots, z^*_n\} \). \( i \neq \nu \). The observation noise considered in \( (17) \) is subject to the Gaussian distribution \( \mathcal{N}(0, \sigma^2) \). Thus, the posterior membership function can be deduced according to marginalization and conditional distribution \( [29] \)
\[
\omega_j(z) \sim \mathcal{GP} \left( \mu_{\omega_j}(z), \sigma_{\omega_j} \right)
\]
where \( \mu_{\omega_j}(z) = \mu_{\omega_j}(z) + \kappa(Z^*, z^*) \mu_{\omega_j}(Z^*) + \sigma^2 \kappa^{-1}(\Omega_j^* - \mu_{\omega_j}^*) \) and \( \sigma_{\omega_j} = \sqrt{\kappa(Z^*, z^*) + \sigma^2 \kappa^{-1} - \kappa(Z^*, z)^2} \). Let \( \alpha \) be the confidence coefficient, the FOU induced by system uncertainty and measurement noise is limited by
\[
\omega_j(z) = \mu_{\omega_j}(z) - \alpha \sigma_{\omega_j},
\]
and \( \omega_j(z) = \mu_{\omega_j}(z) + \alpha \sigma_{\omega_j} \),
\[
\omega_j(z) \sim \mathcal{GP} \left( \mu_{\omega_j}(z), \sigma_{\omega_j} \right)
\]
in which \( \omega_j(z) \) and \( \omega_j(z) \) denote the UMF and LMF for the corresponding rule such that \( 0 < \omega_j(z) < \omega_j(z) \leq 1 \). In this regard, the firing strength of the rule \( j \) is within the following interval set:
\[
W_j(z_i) = [\omega_j(z_i), \omega_j(z_i)], j \in \mathbb{Z}^+[1, l],
\]
which leads to an interval set \( \{y^L_j, y^R_j\} \) by substituting \( [29] \) into \( (7) \). The upper and lower bounds of \( y_i \) are processed in ascending order as \( \{y^L_j\} \) and \( \{y^R_j\} \), then the ordinal number of the associated membership function is adjusted to match the sorted output. The center-of-sets type-reduction method \( [30] \) is employed to convert the interval set into crisp output:
\[
y_{y_c} = \min_{z \in \mathbb{Z}^+[1, l]} \frac{\sum_{q=1}^{l-1} \omega_{y_c}(z) y^q + \sum_{q=1}^{l} \omega_{y_c}(z) y^q}{\sum_{q=1}^{l-1} \omega_{y_c}(z) + \sum_{q=1}^{l} \omega_{y_c}(z)}
\]
\[
y_{y_R} = \max_{z \in \mathbb{Z}^+[1, l]} \frac{\sum_{q=1}^{l-1} \omega_{y_R}(z) y^q + \sum_{q=1}^{l} \omega_{y_R}(z) y^q}{\sum_{q=1}^{l-1} \omega_{y_R}(z) + \sum_{q=1}^{l} \omega_{y_R}(z)}
\]
where \( L \) and \( R \) are the switching points obtained by Karnik–Mendel (KM) or EKM algorithm \( [31], [32] \) such that \( y^L_i \leq y_c \leq y^R_i \) and \( y^R_i \leq y_c \leq y^L_i \). Hence, the crisp output of the \( \Pi^2 \) PFMB system can be calculated as
\[
y_i = \frac{y_{y_c} + y_{y_R}}{2}.
\]
Algorithm 1: Data-Driven GP-IT2PFL Algorithm

Data: Input-Output pair data \([x_i, y_i]_{i=1}^{N}\)

Result: IT2 polynomial fuzzy system and identified clusters

1. Z-Score normalization for the collected input data \(x_i\)
2. Calculate the cluster centres and prior membership functions as (5) and (6)
3. while \(i \leq N\) do
   4. Conduct state augmentation as \(Z_i\) using polynomial operator
   5. Update the parameter matrices as (12)-(14)
   6. Construct the Gaussian process for membership function as (15)
   7. Calculate the UMF and LMF as (19)
   8. Calculate the IT2 polynomial fuzzy output as (23)
   9. end

Remark 3: The first and second-order derivatives of the input data are also included in the observation matrix (8), where the adopted UKF technique facilitates noise attenuation. Additionally, the implementation of z-score normalization as described in (1) assists in mitigating the detrimental impact of input noise on the clustering outcomes. Therefore, it facilitates the expression of the two-limb coordination trajectories. It ensures that the polynomial-operator-based modeling approach is more effective to extract the nonlinear features of MIMO mapping, compared to traditional methods that treat the input data as independent (25), (33)-(35).

Remark 4: In contrast to (35), (36) which consider parameter corrections to build IT2 fuzzy sets, a non-parametric Gaussian process is utilized to model uncertainty in the membership function directly. The resulting FOU benefits from capturing the parameter perturbations and the black-box characteristics due to measurement noise and system uncertainty. It is worth noting that the membership function in (6) is utilized as the prior mean function \(\mu_\omega(z_i)\) when reconstructing the FOU in terms of uncertainty. The inverse of the covariance matrix \((\kappa(Z^*, Z^*) + \sigma^2 I)\) can be effectively numerically calculated through Cholesky decomposition.

III. COORDINATED MANIPULATION

We have carried out two experiments to check whether the proposed algorithm can capture coordination patterns between human limbs. In this section, we first test the hand-foot coordination of a solo operator in three bimanual picking and placing tasks.

A. Experiments

This experiment was conducted at the Institute of Perception, Action, and Behaviour at Edinburgh University. The bimanual hand-foot coordination system is composed of two torque-controlled, 7-DoF Franka Emika robotic manipulators, which are both equipped with a flat 3D-printed contact plate mounted on the end-effector. The bimanual impedance controller (37) is implemented in velocity and offset modes to make full use of the robot workspace while ensuring safe interaction with the environment. Real-time module calculation is performed on the basis of the rigid-body dynamic library Pinocchio, where EiQuadProg++ is used to solve quadratic programs for inverse kinematics. The operator can provide 6-DoF (position and orientation) commands via the foot interface and Sigma-7 haptic device (Force Dimension Inc., Switzerland), where the foot interface is a parallel-serial-structure (38) using multiple force sensors to collect the foot position and force information in four DoFs. The gravity compensations for both hand and foot interfaces are set to assist human manipulation and eliminate human fatigue.

We carry out three typical tasks based on the hand-foot coordination system, namely (1) picking and placing a deformable ball (Fig. 2a-c); (2) picking and placing a rigid block (Fig. 2d-f); and (3) grasping a mobile block on the conveyor (Fig. 2g-i). The object block in tasks 2 and 3 has a rectangular shape (12 × 16 × 14 cm), weight 0.8 kg, and is fitted with interconnecting industrial fittings (Anderson Power Mid-Power Spec-Pak). Task 1 involves capturing the softball located above the block and then moving it to the basin in the red area. Notice that the block is removed after the grasping action is completed to avoid collisions. In task 2, the block located in the red area on the right side is first transferred to the red area on the left side and then placed back in its original position on the right side. With respect to task 3, the dual-arm robot’s objective is to grab the block moving on the conveyor and place it in the red rectangle. The three tasks are progressively more difficult and require higher levels of hand-foot coordination, ranging from static to dynamic grasping.

At the training stage, the aforementioned three tasks are performed by the left hand and foot corresponding to the left and right robotic manipulators, respectively, whose spatial motion positions on the \(x\), \(y\), and \(z\) axis are fed into the proposed approach as inputs. Subsequently, the left manipulator moves according to the commands generated by the proposed algorithm at the test stage, while the right robot arm is controlled by the foot interface. The parameters are set as \(l = 4\), \(\chi = 2\), \(\alpha = 1.96\), \(N = 1000\), \(\sigma_\epsilon = 0.1\), \(\sigma_\kappa = 0.6009\), \(\eta_\kappa = 0.0053\), and \(\Delta T = 0.01\) s.

B. Results

To examine the performance of the GP-IT2PFL approach, we evaluate the estimation performance by root-mean-square error:

\[
\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \|y_i - \hat{y}_i\|^2},
\]

where \(y_i\) and \(\hat{y}_i\) represent the actual and the estimated end-effector position, respectively. We construct the neural network (39) and Gaussian process (40) for three tasks in bimanual hand-foot coordination for comparison, where the dataset is identical with the GP-IT2PFL approach, and the number of samples is 5000 (training: 4000, test: 1000) based on an empirical ratio in (41). In particular, the neural network (NN)
Fig. 2. Hand-foot coordinated manipulation tasks. Side views of the (a) initial, (b) grasping, and (c) placing stages of task 1; Demonstration in the operator views of the initial grasping (d), intermediate positioning (e), and final placing (f) stages in task 2; Demonstration in the operator views of the initial (g), grasping (h), and placing (i) stages of task 3.

is composed of 2 hidden layers with 40 and 20 neurons with hyperbolic tangent and rectified linear unit (ReLU) activation functions, the final output layer is with log-sigmoid activation function, where the parameters of the NN are initialized by [42]. Table I presents the comparative results in tasks 1-3, where each value represents the average of the x, y, and z-axis trajectory estimation errors corresponding to different methods. The estimated membership functions subject to UMF and LMF are depicted in Fig. 3, where the practical membership functions subject to measurement errors and uncertainties are effectively captured by the UMF and LMF. We see that the proposed algorithm enables the human to perform bimanual tasks only by foot interface independent of the intervention of two hands.

<table>
<thead>
<tr>
<th>Task</th>
<th>Neural Network</th>
<th>GP</th>
<th>GP-IT2PFL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task 1</td>
<td>0.5808</td>
<td>0.5824</td>
<td>0.074</td>
</tr>
<tr>
<td>Task 2</td>
<td>0.2337</td>
<td>0.2254</td>
<td>0.053</td>
</tr>
<tr>
<td>Task 3</td>
<td>0.5511</td>
<td>0.5556</td>
<td>0.023</td>
</tr>
</tbody>
</table>

IV. SHARED TELESURGERY

To further assess the performance of the proposed method, we carried out a second experiment on human-human bimanual surgical manipulation using the Da Vinci surgical robotic system at Imperial’s Hamlyn Center for Robotic Surgery.
The main operator navigates the left gripper onto the initial peg, while the assistive operator (right gripper) motion data are the outputs. As the surgical robot system lacks force/torque sensors, we have used the open angle of the gripper to represent the grasping force.

B. Results

Fig. 2 shows the real right-hand trajectories and the estimation ones in human-human cooperation, where the assistive operator movement is effectively represented with minimal error. The RMSEs of the estimation errors in x, y, and z directions are $4.2036 \times 10^{-4}m$, $5.3761 \times 10^{-4}m$ and $5.3254 \times 10^{-4}m$, respectively. Similarly, Fig. 3 depicts the estimation profiles of right-hand gripper trajectories in human-human cooperation. The main operator moves the left-hand gripper to the initial peg at the beginning. Upon arrival, the left-hand gripper opens and grasps the block in 8.3s, then transfers the block to the right-hand gripper, as shown in Fig. 3b. It can be found that the right-hand gripper increases its opening angle to take over the object within 7.47s. The estimated angle of the right-hand gripper is close to the real angle in terms of time duration (0.93s difference) and degree (RMSE = 0.7571°). The opening angle of the right-hand gripper and left-hand gripper in Fig. 6 are respectively denoted by positive and negative values of $\theta$.

Four trained mappings (three for translations and one for gripper angle) are implemented for the right-hand robotic manipulator of the surgical robotic system. It acts as an assistive robot cooperating with the human operator according to the operator’s real-time manipulation. Figs. 4c-d demonstrate the trained assistive robot motion at the test stage (human-robot collaboration), which recognizes human motion and cooperates accordingly. For example, the assistive robot gripper waits for the operator’s action within an initial period of 2.93s. When the assistive robot recognizes that the main operator has grasped the object and is moving toward it, it adjusts the opening angle of the gripper to hand over the object, which takes approximately 19.11s. After the object is successfully transferred and the human operator releases the grasping, the assistive robot holds the object and quickly moves to the target peg in 21.37s (see video 2). With the involvement of the assistive robot, the main operator completes the entire task in 27s.

V. Discussion

This paper proposed a new GP-IT2PFL framework to facilitate robot motion learning with tailored assistance in human-robot cooperation, which enables humans to perform bimanual tasks with one limb. This algorithm solved the coupling issue between intra- and inter-input data in two-limb coordination using the polynomial operator and minimized the system uncertainty and measurement noise by Gaussian-process-based upper and lower membership functions. The proposed approach was tested in two applications: solo two-limb coordination and dyad collaboration. The successful implementation requires the same subject to perform a consistent and stable motion strategy under the same initial conditions during training and test phases, where the parameters in the associated fuzzy model are tailored to this subject.
Fig. 4. Experimental setup for robotic surgery: Da Vinci system for laparoscopic surgery with Omega-7 hand controllers. (a) The local console includes dual Omega-7 hand controllers and one monitor to provide visual feedback from the remote side. (b) Da Vinci surgical robotic system with dual manipulators and one camera at the remote side.

Fig. 5. Cooperative peg-transfer task. (a) The human operator controls the left side tool to grasp the block object on the initial peg and passes it to the robot-controlled tool. (b) The robot-controlled tool takes over the transfer block and moves it to the target peg.

Fig. 6 shows that similar behaviors are generated with the GP-IT2PFL algorithm (Fig. 6.c,d), compared with the trajectories and gripper movement of the assistive operator (Fig. 6.a,b). Furthermore, the experiment results on the Da Vinci surgical system have shown that the assistive robot can reduce the intensity of the surgeon’s manipulation effectively by implementing the proposed method, which facilitates assisting and enhancing the movements of both hands, towards enabling an operator to perform tri-or quad-manipulation by the hands, with some robot arms controlled directly with the hands and the other through coordinated motion learned from previous interactions.
Fig. 6. Experimental results from (a-b) human-human training to (c-d) human-robot test. Trajectory learning from the main operator and assistive operator: (a) Cartesian trajectories in human-human telesurgery ($x$, $y$, and $z$ denote the right-hand movement along the associated direction. The dashed lines are the estimate with respect to the associated direction using the proposed method). (b) Gripper grasping $\theta$: the dashed line stands for the estimate of gripper movement. $\theta_{RH}$ and $\theta_{RH}$ represent the angle at which the left and right-hand grippers open. Trajectory learning from the human operator and assistive robot: (c) Cartesian trajectories in human-robot telesurgery ($x$, $y$, and $z$ denote the assistive robot movement along the associated direction). (d) Assistive gripper grasping: $\theta_{RP}$ is the angle of the assistive robot gripper.

adapting to surgeon behavior autonomously. Trained by the trajectories and gripper movements of human-human collaboration, intuitive human-robot cooperation was demonstrated in a peg-transfer task with self-determining movements from the assistive robot.

With the assistive robot, the surgeon can perform the peg-transfer task alone, thus avoiding potential communication errors with an assistant. Moreover, Fig. 6 demonstrates that the proposed GP-IT2PFL approach allows for modeling the coordinated gripper movements. Hence, the assistive robot driven by the proposed method embodies the capability to assist humans with telesurgery tasks in the force-sensor-free scenario. It sheds light on the fact that the proposed algorithm is effective in modeling human motion adaptation.

REFERENCES


no. 4-5, pp. 698–721, 2021.


