Task Parameter Inference in Human-Robot Interaction

Nikol Guljelmović

Master of Science Thesis

TU Delft Delft University of Technology

BioMechanical Engineering - BioRobotics

Challenge the future

TASK PARAMETER INFERENCE IN HUMAN-ROBOT INTERACTION

A Thesis
by

Nikol Guljelmović

in partial fulfillment of the requirements for the degree of

Master of Science
in Biomechanical Design - Biorobotics

at the Delft University of Technology,
to be defended publicly on Friday August 25th, 2017 at 14:00 PM.

Supervisors:	Ir. M. Zeestraten	IIT
	Dr. S. Calinon	IDIAP
	Dr. Ir. J. Kober	TU Delft
	Prof. Dr. Ir. P. P. Jonker	TU Delft

ACKNOWLEDGEMENT

Ever since I saw Data in 'Star Trek: The Next Generation', I was fascinated by intelligent autonomous robots and the question how to learn a machine to interact with humans. After I saw the project ICub, developed at Istituto Italiano di Tecnologia (IIT), I fell in love with the learning robot toddler. This motivated me to pursue an internship or master thesis at IIT (Italy) to learn more about programming by demonstration. Through Professor Pieter Jonker I got in contact with Sylvain Calinon and Martijn Zeestraten. They offered me the opportunity to do an internship at IDIAP research institute in Switzerland and my master thesis at IIT, on programming by demonstration with the use of task-parameterized movement representation. The research on if and how task parameter inference could be achieved was in the beginning phase. I struggled with the question where to apply task parameter inference. For me the most accessible implementation seemed in human robot interaction. I could imagine that task parameter inference can be used in gesture recognition, where the robot observes a human and predict the humans intention act on it. And so my research in this fascinating area of science started.

First of all I would like to thank Pieter Jonker for being my supervisor and for introducing me to Sylvain Calinon and Martijn Zeestraten. I would like to thank Sylvain Calinon for my internship at IDIAP and for my master thesis at IIT and for his guidance throughout my whole research. Especially I would like to thank Martijn Zeestraten for being my supervisor, for all his guidance throughout the research and for his patience, support and time. I would also like to thank Jens Kober for being my co-supervisor.

Furthermore I would like to thank the Stichting Fundatie van de Vrijvrouwe van Renswoude te 's-Gravenhage, Dr. Hendrik Muller's Vaderlandsch Fonds, Genootschap Noorthey and Erasmus +, to have made it possible to pursue the internship and thesis project at IDIAP and IIT.

Special thanks goes to Christina Budimir for always being there for me, believing in me, for her love and encouragement and to Dorus Brouwer for his support to take on this beautiful opportunity, his love, encouragement and patience.

ABSTRACT

The interaction between robots and humans is an important topic within robotic research, with its goal to create robots that can cooperate autonomously with others in human daily life situations. One of the problems is to program a robot to perform skills in a not predefined environment (e.g., the area where to perform the task, different partners, etc.). Programming by demonstration provides a promising solution to this problem.
Task-parameterized movement representation, as an approach for the generalization of demonstrations, is used to represent data from multiple local perspectives within the global reference frame, through which more accurate information about multiple aspects of the movement is given. The estimated transformation between the different perspectives and the global reference frame in task parameter inference can be used for gesture recognition.
In this thesis, task parameter inference in the application of human-robot interaction, a method called TP-inference approach, is investigated. It consists of a combination of task parameter inference and task parameter movement retrieval. A task-driven model is used to generalize the demonstration data and the task parameter inference is achieved by using the orthogonal Procrustes analysis.
The TP-inference approach is tested for various static tasks and is compared to the Probabilistic Movement Primitive (ProMP) approach [1]. The test results indicate that for simple and or similar movement of the human and robot, the TP-inference approach performs less accurate than the ProMP. For complex movements the TP-inference preforms more accurate than the ProMP.

CONTENTS

List of Figures and Tables

Figures

1

INTRODUCTION

1.1. MOTIVATION

Even though robots nowadays are advanced machines, they are still mostly pre-programmed for specific actions. The robots are designed for specific tasks in a specific context. The main question remains, how can the robot perform tasks in new and unpredictable situations. For interactive tasks the robot should be able to adapt his actions to the actions of his partner. The variety of tasks in which the human requires assistance and the variety in which to perform those tasks are practically unlimited. In order to overcome the need for teleoperation and manual "hard-coding" of every behavior, a learning approach is required.

Programming by demonstration (PbD) is a promising approach to deal with the variations within tasks. One of the challenges in PbD is the generalization of a learned task to new situations. Task-parameterized movement representation provide a potential route to achieve this generalization. Task parameters are variables that describe a situation, such as the position of an object in the environment. The robot can use and infer these parameters to understand how its movement relates to the positions of objects in the environment.

Humans are able to understand and estimate gestures, for example how big an object is that another person is referring to or understanding where an object is placed when another person is pointing in that direction. For a robot these aspects of understanding and estimating gestures, can be used in human-robot interaction, where the robot first observes the movement of the human and by inferring the task parameters (task parameter inference) responds accordingly.

Because task-parameterized movement representation is based on a movement, it is applicable to all interaction involving a robot regarding a movement. For example, not only for a humanoid robot or an industrial robot arm picking up a box, but also for quadcopters used in tracking humans during outdoor sports, hexapods playing football, wheeled robots in human-robot teaming for disaster response and even gesture based computer interfaces. Concluding, task-parameterized movement representation is applicable to all interaction were there is an observation of a movement and a response to that observation.

1.2. RELATED WORK

When dealing with human-robot interaction two aspects pose a problem for the robot, action recognition and anticipating on the movement of the human. Which action is the human performing, what should be the action and which variation of that action should the robot perform, are the questions that arise.

Lee et al. [4], presented a method in which the robot imitates the human movement by a motion capturing control method in Cartesian space. By means of a simple motion imitation, the robot

learns the motion and interaction primitives.
Ikemoto et al. [5] presented a machine learning algorithm by using tasks that mimic the human parenting behavior of assisting a child with standing-up and walking.
Although in the aforementioned work the tasks are learned through PbD and the work uses trajectory observations, the combination of force signals and trajectory input makes that the work is not comparable with the work in this thesis.
Amor et al. [6] introduced a representation called the Interaction Primitives (IP), which is a variation on the Dynamic Movement Primitives (DMPs). They focused on the learning of interactive motor skills, which allows the robot to engage in physical cooperative tasks with the human. To achieve this the movement of two humans in interaction was recorded and subsequently used to learn a compact IP model.
The IP specifies both the executed movements and the correlations therein. Thereafter this model is used by the robot to engage in a similar interaction. Ewerton et al. [1], continued the work of Amor et al. [6], where the framework was limited to represent and generalize a single interaction pattern. Ewerton et al. [1] wanted to extent these interactions with additional, and different patterns by using a mixture of interaction Probabilistic Movement Primitives (ProMP). They proposed a method in which a Gaussian Mixture Model (GMM) of the interactive primitives is used to create nonlinear correlations between the movements of the human and the robot. It is a method to learn multiple interaction patterns from uncategorized demonstrations.
The approach proposed in this thesis, has similarities with the the work of Amor et al.[6] and Ewerton et al.[1], as they also used a trajectory based model. Instead of using DMPs to model the trajectories and a correlation between the movements of the human and robot, the proposed approach in this thesis uses a task-parameterized movement representation, where the task parameters inferred from the movement of the human are used as an input to estimate the movement of the robot.

1.3. PROBLEM STATEMENT

This thesis investigates the use of task parameter inference (TP-inference) in human-robot interaction, thereby proposing a new approach to tackle the problems encountered by the robot in interactive tasks. The following questions that are addressed:

RQ1. Can task parameter inference be applied in human-robot interactive tasks?
RQ2. How does the task parameter inference approach compare with a common approach in this field?

1.4. STRUCTURE OF THE THESIS

In Chapter 2 this thesis starts by introducing some background theory. In this chapter, three main topics are discussed in detail: task-parameterized movement representation, the Procrustes analysis, that is used in the TP-inference approach and the background theory, needed for the benchmark approach (ProMP). Chapter 3 discusses the design of the TP-inference approach, continued by the benchmark ProMP approach in detail in Chapter 4. To test the TP-inference approach and to compare the results to the ProMP approach, four different static tasks were recorded by using OptiTrack. The choice of the tasks and the explanation of the experimental setup used to record the tasks, is presented in Chapter 5. The results of the tests and the comparison are analyzed in Chapter 6, followed by a conclusion with a discussion in Chapter 7.

2

PRELIMINARY

To fully understand the designed TP-inference approach and the used benchmark approach, preliminary theory is needed. Therefore, in the first section, task-parameterized movement representation is introduced, giving the 'building blocks' for the TP-inference approach. In the section that follows the Procrustes analysis is explained, which is used to infer the task parameters. Finishing with the explanation of the theory on which the benchmark approach is based on, in the last section.

2.1. TASK-PARAMETERIZED MOVEMENT REPRESENTATION

One of the approaches within imitation learning, to generalize the demonstrations to new situations, is task-parameterized movement representation. This concept will be explained by using the task of picking up a box placed on the table, illustrated in Figure 2.1. During the learning phase four demonstrations are given (left image of Figure 2.1), with in each demonstration a different location and orientation of the (orange) box. The goal is to generalize these demonstrations such that the task can be reproduced in the same and in new situations.

Without introducing any structure about the learning problem, it will not be possible to reproduce the old (orange boxes) and new (red box) situation as shown in Figure 2.1. As illustrated in the figure, there is a small variance between the demonstrations at the beginning of the movement and a high variance at the end of the movement (location of the box). The average of the demonstrations per time step are illustrated in the right image (red line). This line indicates the "best" estimation of the reproduction when assuming the data to be observed from the actor.

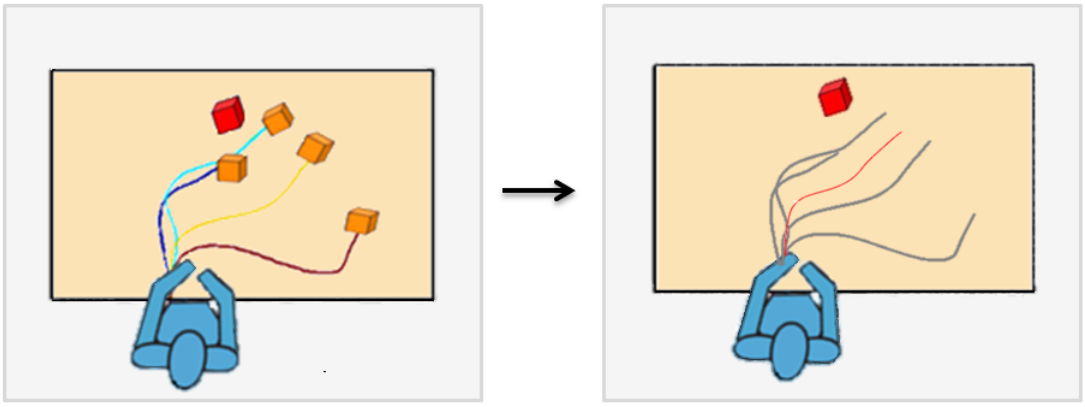


Figure 2.1: An illustration of a human demonstrating picking up a box. In the left image four demonstration are given with each a different end position and orientation of the orange box. The red box shows a potential new situation. The robot should be able to move towards, after observing the demonstrations. The right image shows that it is not possible to retrieve the correct movement to the red box without introducing any structure about the learning problem.

If the data is not only observed from the actor but also from multiple perspectives, more accurate information would be available about the last part of the movement. Going back to the example given in Figure 2.1, the different perspectives would be the global reference frame (indicated in grey, with its axis in the left lower corner) and a local reference frame placed at the center of the humans torso. The transformation between the local reference frame (local frame) and the global reference frame (global frame) is called a task parameter, see Figure 2.2.

The perspective of the global frame and the local frame of the human in this situation, is quit similar and the data observed from these perspectives give similar information: low variance between the demonstration at the beginning of the movement and high variance at the end of the movement. If the data is not only represented from the local perspective of the human, but also from the local perspective of the box, more accurate information would be available concerning the last part of the movement. From the perspective of the box, the data will be less accurate at the beginning of the movement (position of the human) and more accurate at the end (position of the the box). In Figure 2.3 the two local frames are visualized and is shown how these two can be put in one global frame. The right image in Figure 2.3 demonstrates how a rotation (**A**) and a translation (**b**) can be used to task-parameterize a movement. This way of representing the data is called a task-parameterized movement representation.

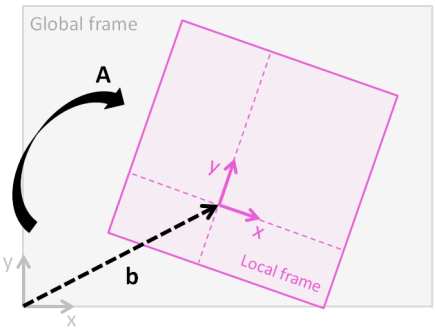


Figure 2.2: Schematic representation of task parameters. A task parameter describes a linear transformation between a global and local frame. The A stands for the rotation matrix and b for the translation vector.

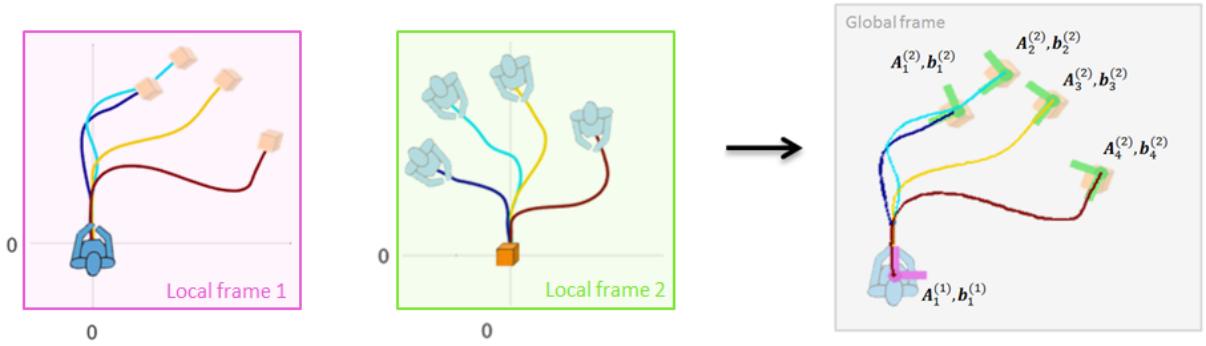


Figure 2.3: A schematic representation of the two local frames and the global frame. The first two images show the two local frames used in this example. One frame is positioned at the center of the human torso (Local frame 1) and one is positioned in the center of the box (Local frame 2). The last image shows the location and orientation of the different local frames (indicated by the location and orientation of the axis with the related color of the local frames) within the global frame. The rotation and translation of the local frame with respect to the global frame is indicated by $\mathbf{A}_p^{(q)}$ and $\mathbf{b}_p^{(q)}$ respectively. The subscript p indicates the number of the demonstration and q the number of the frame.

By using task-parameterized movement representation to estimate the movement for the new situation (e.g. the red box in Figure 2.1), the two local frames will be placed in the global frame, on the position and with the orientation of the new task-parameters (location of the human and the red box). The mean and variance of the data per time step are calculated for both frames. When looking at the beginning of the movement, the local frame of the human provides information with low variance and the local frame of the box provides information with high variance. At the end of the movement the opposite occurs. When retrieving the new movement, the information obtained from the local frame with the lowest variance at that specific time step, will get more value, resulting in an estimation of the new movement, see Figure 2.4.

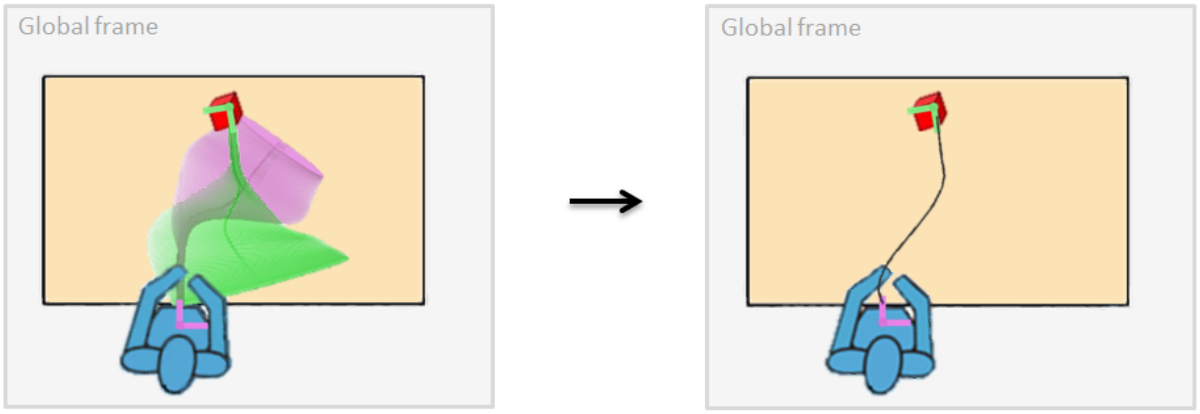


Figure 2.4: An illustration of the movement retrieval. The first image shows how the model parameters, learned from the demonstrations (in this case the mean and covariance from the all demonstrations observed from the local frames, at each time step), are placed in the new situation to retrieve the new movement. The second figure shows the estimated new movement in red, with the use of linear regression.

2.1.1. THREE COMPONENTS WITHIN TASK-PARAMETERIZED MOVEMENT REPRESENTATION

In task-parameterized movement representation there are three main components:

Motion data, the Cartesian coordinates per time step that describe the movement of the task.

Task parameters, the location and orientation of the local frame with respect to the global frame. For the case described in the previous paragraph, the task parameters are the location and orientation of the human and the box with respect to the global frame.

Model parameters, describe the movement or skill behavior. For the case described in the previous paragraph, the model parameters are the set of demonstrations observed by the two local frames (humans perspective and the perspective of the box).

If two components are known, it is possible to estimate the third one, see Figure 2.5.
When the motion data and the task parameters are given (learning phase), the model parameters can be estimated. The model parameters can be represented with various models such as a Gaussian Mixture Model [7–11] and a task-driven model. A task-driven model, is a model that either maintains all the data of the demonstrations (as shown in the first two image in Figure 2.3) or stores the mean and covariance, per timestep, of the demonstrations (shown in the first image of Figure 2.4).
If the task parameters and the model parameters are known, the retrieval of the movement is possible by using approaches as Gaussian Mixture Regression [7, 9, 11, 12], Gaussian Process Regression [7, 13, 14] and Locally Weighted Regression [7, 13]. The type or regression that can be used depends on the representation used to model the skill.
And finally, if the motion data and the model parameters are known, the task parameters can be inferred.
The interest of the literature research was to find methods to infer the task parameters, given the model parameters and a new movement. Scientific research is consulted to find an approach that can infer the transformation between two data sets. Transformation inference methods were found in research fields, such as pattern recognition, computer vision and data analysis. Three promising methods for the task parameter inference problem were found during this research: Epipolar geometry [15–17], the Procrustes analysis [18–25] and the Delaunay Triangulation [26–28].

Even though all three methods are able to find the affine transformation between two shapes, the computational speed and the different possibilities within the Procrustes analysis (affine, full Procrustes with scaling and weighted function) made the Procrustes analysis the most suitable method for the implementation in the TP-inference.

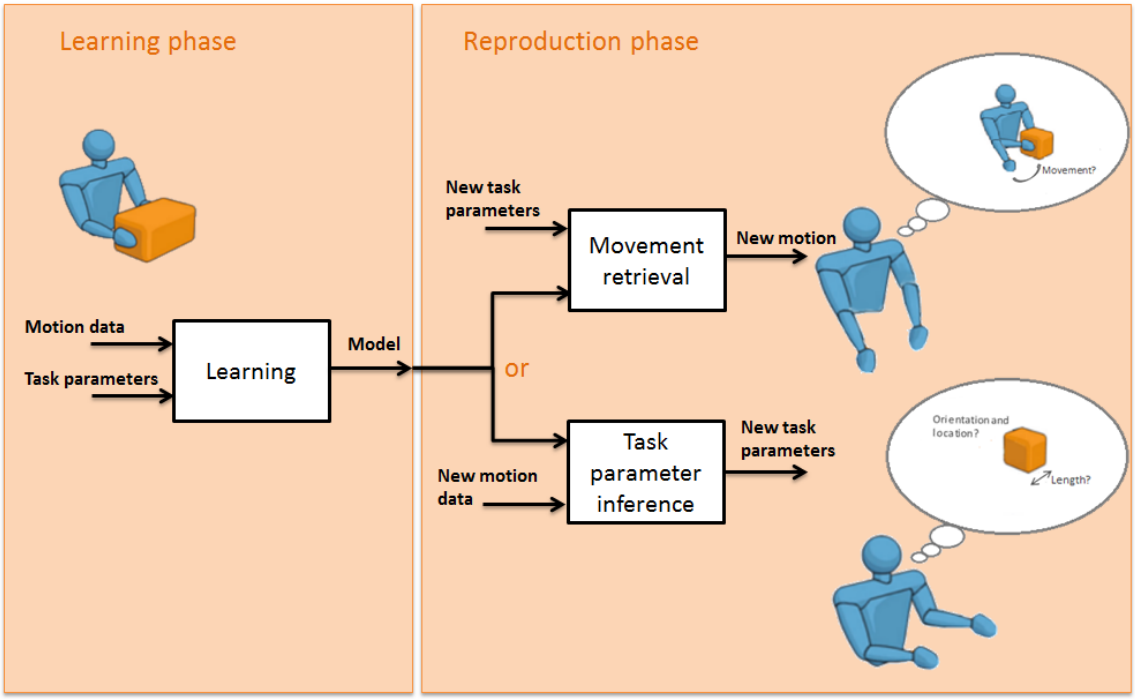


Figure 2.5: A schematic representation of the two phases in Programming by Demonstration (PbD). The left side illustrates the learning phase, with an example of picking up a box (based on the image used in [2]). For this example, the input data to make this model are: the position, the orientation and the size of the box (task parameters) and the movement of picking up the box (motion data). This model will be used as an input in the reproduction phase, to either retrieve the model when given new task parameters, or to estimate the task parameters when given a new movement, shown on the right side of the image.

2.2. PROCRUSTES ANALYSIS

The Procrustes analysis is a form of statistical shape analysis, used to compare the shape of two objects. The transformation between the two shapes is calculated by first optimally "superimpose" the shapes. This superimposition is achieved by optimally translating, rotating and uniformly scaling the one object to the other, to obtain position, orientation and size, by minimizing the measure of the shape difference (the Procrustes distance) between the objects. The Procrustes analysis is used in various studies, such as computer vision[22], pattern recognition [25], medical data analysis [20] and geoinformatics [18], to infer the transformation between shapes/ data sets. In this thesis the Procrustes analysis is employed to infer the transformation between the data from the local frame and the data from the global frame, in the equations indicated by $\mathbf{X}$ and $\mathbf{Y}$ respectively.
A distinction can be made between full Procrustes superimposition (PS), in which both placement and scaling are taken into account, and partial PS, where scaling is not performed. In both cases PS can perform reflection if needed. The orthogonal Procrustes analysis is a mathematical method to calculate the optimal rotation and/ or reflection, needed to superimpose two objects. The calculations for the orthogonal Procrustes analysis will be described below.

ORTHOGONAL PROCRUSTES ANALYSIS

The analysis will be explained by using the two data sets: $\mathbf{X} \in \mathbb{R}^{N \times D^I}$ describing the input data and $\mathbf{Y} \in \mathbb{R}^{N \times D^0}$ describing the output data. This means that $\mathbf{X}$ will be transformed to $\mathbf{Y}$. To do so, a standard regression problem consists on finding $\mathbf{A}$ is used, such that $\mathbf{Y}$=$\mathbf{AX}$. The least squares solution of the problem is given by [29]:

$$\hat{\mathbf{A}} = \mathbf{X}^{\dagger}\mathbf{Y} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}. \tag{2.1}$$

In the orthogonal Procrustes analysis, is searched for an orthogonal $\mathbf{A}$ such that $\mathbf{A}\mathbf{A}^T = \mathbf{I}$

$$\hat{\mathbf{A}} = \arg\min_{\mathbf{A}} ||\mathbf{Y} - \mathbf{X}\mathbf{A}||_F^2 \quad \text{s.t. } \mathbf{A}^T\mathbf{A} = \mathbf{I}. \tag{2.2}$$

When p=q=2 for the $\mathbf{L}_{p,q}$ norm (for a $m \times n$ matrix $\mathbf{A}$, $||\mathbf{A}||_{p,q} = [\sum_{j=1}^{n}(\sum_{i=1}^{m}|\mathbf{a_{ij}}|^p)^{q/p}]^{1/q}$), it is called the Frobenius norm (subscribt F).
By minimizing the equation above and applying Singular Value Decomposion (SVD), a solution can be found for the rotation matrix $\mathbf{A}$. Because $\hat{\mathbf{A}}$ is a rotation matrix (with $D^I = D^O$), it must be ensured that its columns form a right-handed coorinate system. For $D^I = D^O = 3$, this can be computed with:

$$\hat{\mathbf{A}} = \mathbf{V}\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & d \end{bmatrix}}_{\mathbf{I}}\mathbf{U}^T, \tag{2.3}$$

where $d = \text{sign}(\det(\mathbf{V}\mathbf{U}^T))$.

LEAST-SQUARES RIGID MOTION

The problem given above can be extended to find the best rotation $\mathbf{A}$ and translation $\mathbf{b}$, that map the two data sets $\mathbf{X}$ and $\mathbf{Y}$ [30]

$$\min_{\mathbf{A},\mathbf{b}} ||\mathbf{Y} - \mathbf{X}\mathbf{A} - \mathbf{1}_N\mathbf{b}^T||_F^2 \; s.t. \; \mathbf{A}^T\mathbf{A} = \mathbf{I}. \tag{2.4}$$

An estimation can be made by first solving the orthogonal Procrustes problem for centered $\mathbf{X}$ and $\mathbf{Y}$, and the evaluating $\mathbf{b}$ assuming $\mathbf{A}$ is known, using the following equations:

$$\boldsymbol{\mu}^X = \frac{1}{N}\sum_{n-1}^{N}\mathbf{X}, \qquad \boldsymbol{\mu}^Y = \frac{1}{N}\sum_{n-1}^{N}\mathbf{Y}, \tag{2.5}$$

$$\tilde{\mathbf{X}} = \mathbf{X} - \mathbf{1}_N\boldsymbol{\mu}^{X^T}, \qquad \tilde{\mathbf{Y}} = \mathbf{Y} - \mathbf{1}_N\boldsymbol{\mu}^{Y^T}, \tag{2.6}$$

$$\hat{\mathbf{A}} = \mathbf{V}\tilde{\mathbf{I}}\mathbf{U}^T, \text{ with } \tilde{\mathbf{X}}^T\tilde{\mathbf{Y}} = \mathbf{U}\mathbf{S}\mathbf{V}^T, \quad \tilde{\mathbf{b}} = \boldsymbol{\mu}^Y - \tilde{\mathbf{A}}\boldsymbol{\mu}^X, \tag{2.7}$$

μ calculates the mean coordinate of the data set, in order the be able to center the data (e.g. $\tilde{\mathbf{X}}$). The calculation of the rotation is explained in Equations 2.2-2.3. The translation can be calculated by rotating the mean matrix of data set $\mathbf{X}$ with $\hat{\boldsymbol{A}}$ and calculating the difference between the mean of data set $\mathbf{Y}$ and the rotated mean of data set $\mathbf{X}$.
When a part of the shape data is uncertain, a weight function can be introduced to give priority to transform the data according to the certain part of the data. In the Procrustes analysis the weight can be added by weighting each pair of datapoint $\boldsymbol{X_n}$ and $\boldsymbol{Y_n}$ by a scalar w_n with $\sum_{n=1}^{N} w_n = 1$ [18].

FULL PS

The problem can be extended by taking the relative scaling of X and Y into account. But for these equations the data sets have to be centred in advance, resulting in a translation of zero in these equations [31]

$$\min_{\mathbf{A},\mathbf{b}} ||\mathbf{Y} - \beta\mathbf{X}\mathbf{A} - \mathbf{1}_N\mathbf{b}^T||^2, \tag{2.8}$$

where $\beta > 0$ is the scale parameter. The full PS solution to the minimization Equation 2.8, is given by $(\hat{\mathbf{b}}, \hat{\beta}, \hat{\mathbf{A}})$ where:

$$\hat{\boldsymbol{b}} = \mathbf{0}, \tag{2.9}$$

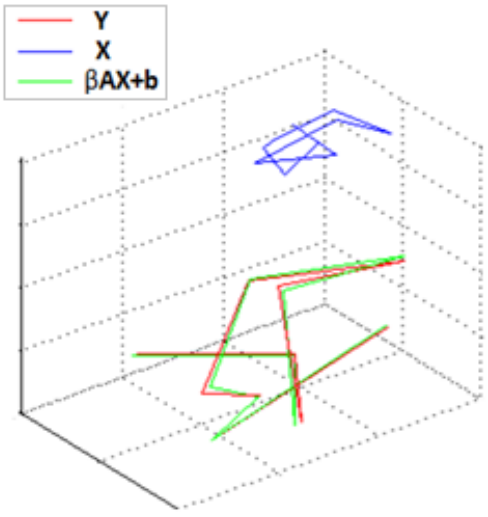


Figure 2.6: Full Procrustes analysis

$$\hat{\boldsymbol{A}} = \boldsymbol{U}\boldsymbol{V}^T, \tag{2.10}$$

where

$$\boldsymbol{Y}^T\boldsymbol{X} = ||\boldsymbol{X}||||\boldsymbol{Y}||\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{U}^T, \boldsymbol{U}, \boldsymbol{V} \in SO(m), \tag{2.11}$$

with Λ a diagonal $m \times m$ matrix of positive elements except possibly the last element when reflection occurs

$$\hat{\beta} = \frac{\text{trace}(\boldsymbol{Y}^T\boldsymbol{X}\hat{\boldsymbol{A}})}{\text{trace}(\boldsymbol{X}^T\boldsymbol{X})}. \tag{2.12}$$

2.3. INTERACTIVE PRIMITIVE

Another approach, that is used within imitation learning to generalize a demonstrated movement to new situations, is Interactive Primitives (IP). Interactive Primitives is used to generalize the demonstrations of both the human and the robot, such that the important aspects of the individual tasks are stored, as well as the correlation between the robot and the human, vice versa. There are two types of IPs: IP based on dynamic motor primitives (DMP) and IP based on probabilistic movement primitives (ProMP).

2.3.1. DMP

DMP is an adaptive representation of the movement of a human or a robot. The general idea behind DMP is to encode a recorded movement as dynamic systems, that can be used to generate different variations of the original movement. Generally a simple dynamical system, a damped spring model, is used to shape the trajectory [32]:

$$\tau\ddot{y} = \alpha_y(\beta_y(g-y) - \dot{y}) + f(x), \tag{2.13}$$

which can be written as a first-order system:

$$\begin{aligned} \tau\dot{z} &= \alpha_y(\beta_y(g-y) - z) + f(x), \\ \tau\dot{y} &= z, \end{aligned} \tag{2.14}$$

where y is a state variable (e.g. the joint angle that needs to be controlled), g is the corresponding goal state, and τ is a time constant. The first set of terms represents a critically damped linear system with constant coefficients α_y and β_y. The critically damped system results that y monotonically converges towards g, where g functions as a single point attractor. The last term is the forcing function f and is added to create the desired, more complex trajectory. The forcing term can be written as:

$$f(x) = \frac{\sum_{i=1}^{N}\psi_i(x)w_i}{\sum_{j=1}^{N}\psi_j(x)}x(g-y_0) = \phi(x)^T(\boldsymbol{w}), \tag{2.15}$$

where **w** are the weight vectors and $\psi_i(x)$ the Gaussian basis function

$$\psi_i(x) = e^{(-\frac{1}{2\sigma_i^2}(x-c_i)^2)}, \tag{2.16}$$

where σ_i is the width and c_i the center of the basis function. The basis functions depend on the phase variable x and on the movement amplitude $(g-y_0)$. The phase variable is the state of the canonical system, shared by all the degrees of freedom (DoF). It has the form $\dot{x} = -\alpha_x x\tau$, where $x_0 = 1$ at the beginning of the movement, which decays towards zero when getting closer to the end of the movement. The movement amplitude is needed when scaling is applied to the trajectory. The elements of the weight vector **w** (shape-parameters), determines the acceleration pattern of the movement and so indirectly also the shape of the movement. For each DoF a separate set of **w** as well

as the goal attractor g are learned. The goal attractor can be used to change the target position and the time scaling parameter τ to change the executing time of the movement. The weight parameters $\mathbf{w}$ can be obtained from the observed trajectories $\{y_{1:T}, \dot{y}_{1:T}, \ddot{y}_{1:T}\}$, by first computing the forcing function that can reproduce the given trajectory [32]:

$$\boldsymbol{F}_i = \tau^2 \ddot{y}_i - \alpha_y(\beta_y(g - y_i) - \tau \dot{y}_i). \tag{2.17}$$

With this equation the system $\boldsymbol{\Phi w} = \boldsymbol{F}$ can be solved using a least squares algorithm [6]

$$\boldsymbol{w} = (\boldsymbol{\Phi}^{\boldsymbol{T}}\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^{T}\boldsymbol{F}, \tag{2.18}$$

where Φ is a matrix containing of basis vectors for all time steps.

The limitation of IP is formulated using DMPs, is the linear correlation between the movement of the human and the robot, due to Gaussian conditioning. This limitation is illustrated in Figure 2.7, where the model is trained by using only one Gaussian. The Gaussian used to describe the movement of the human and the robot individually and the relationship between the movement of the human and the robot, is called a multivariate Gaussian distribution. Because the movement of the robot is similar to the movement of the human, it is possible to retrieve the correct movement in the reproduction phase. If the movement of the robot is different from the movement of the human, it will not be possible to retrieve the correct movement, see Figure 2.7, second example.

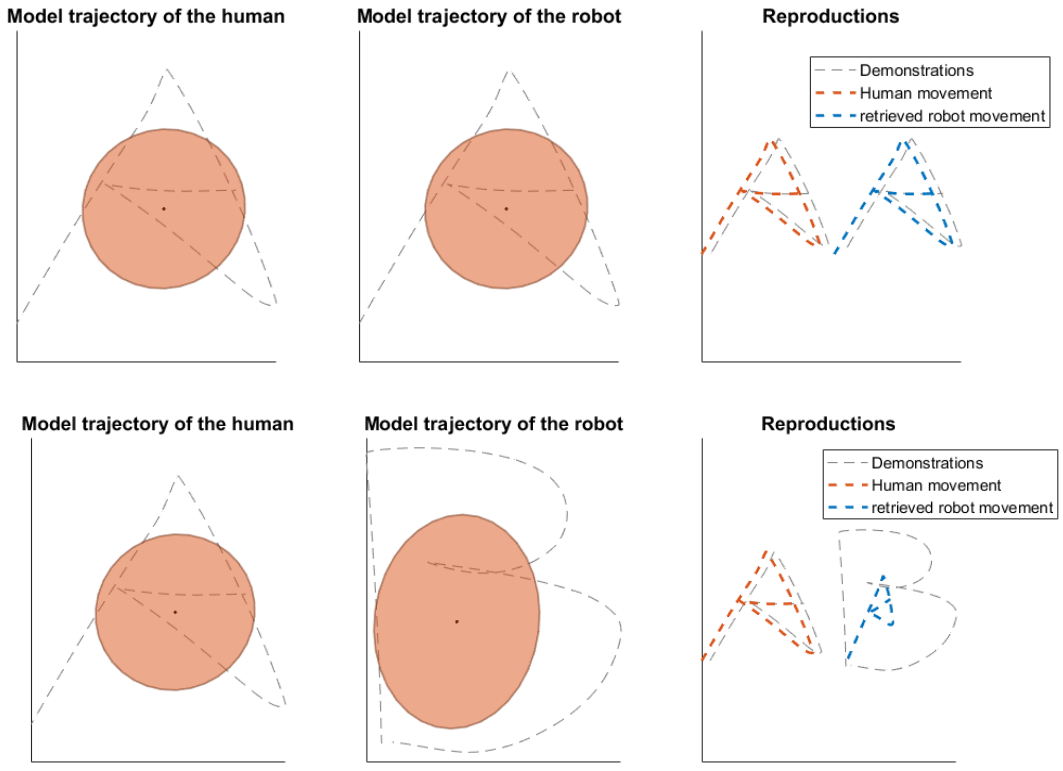


Figure 2.7: One Gaussian kernel is used to represent the movements. The part of the model, which is representing the relation between the human and robot is left out. The right image shows the reproduction for a new A movement. The data used for these plot is retrieved form the matlab code from S. Calinon. [3]

CONDITIONAL GAUSSIAN DISTRIBUTION

One of the properties of the multivariate Gaussian distribution is that if two sets of variables are Gaussian distributed, then the conditional distribution of one set conditioned on the other set, is also Gaussian distributed [33].

Having a n-dimensional vector **X** that describes the data of the human and the robot, then the mean and covariance can be written as follows

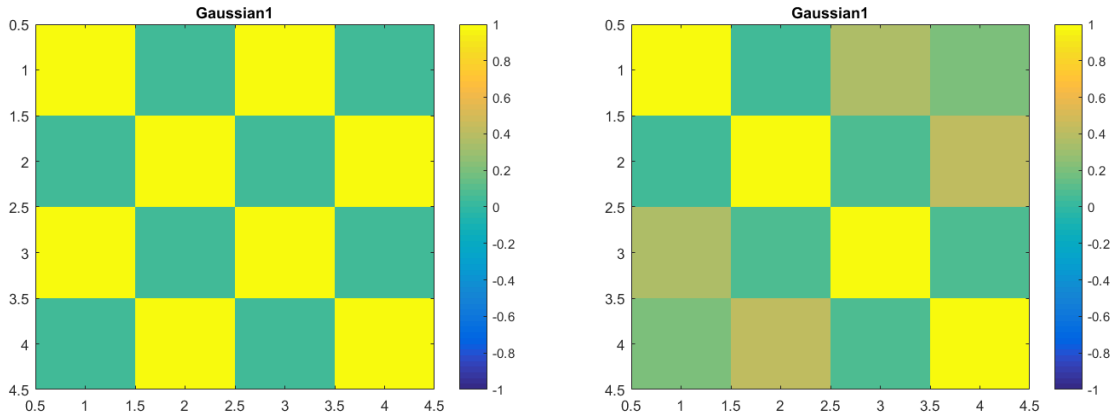
$$\mu = \begin{bmatrix} \mu_H \\ \mu_R \end{bmatrix} \text{ and } \Sigma = \begin{bmatrix} \Sigma_{HH} & \Sigma_{HR} \\ \Sigma_{RH} & \Sigma_{RR} \end{bmatrix}. \tag{2.19}$$

Here for instance μ_H gives the means for variables in set $\mathbf{x}_H$ which is the vector representing the data of the human movement and Σ_{HH} gives the covariances for set $\mathbf{x}_H$. The μ_R and Σ_{RR} represent the mean and covariances for the variables in set $\mathbf{x}_R$, the data of the movement of the robot. The matrix Σ_{HR} gives the covariances between set $\mathbf{x}_H$ and $\mathbf{x}_R$ (as does Σ_{RH}).
The conditional distribution of set $\mathbf{x}_H$, given the known values for set $\mathbf{x}_R$, is a multivariate normal distribution with [34]

$$\mu_{x_H|x_R} = \mu_H + \Sigma_{HR}\Sigma_{RR}^{-1}(x_R - \mu_R), \tag{2.20}$$

$$\Sigma_{x_H|x_R} = \Sigma_{HH} - \Sigma_{HR}\Sigma_{RR}^{-1}\Sigma_{RH}. \tag{2.21}$$

The $\Sigma_{x_H|x_R}$ term indicates the strength of a linear relation between two variable sets. In the Figure 2.7 the Σ_{HR} and Σ_{RH} could not be properly visualized. In Figure 2.8 the correlation of the whole Σ matrix of the samples AA (a) and AB (b) is plotted.



a: AA b: AB

Figure 2.8: Correlation plots of the Gaussian used to model the interaction. (a) is the situation of AA as given in Figure 2.7, first example and (b) the situation of AB as given in Figure 2.7, second example.

Figure 2.8 illustrates that even though the Gaussian, which has a low correlation when observing Σ_{HH} and Σ_{RR}, used to represent the AA (see Figure 2.7), there is a high correlation between the human and the robot movement Σ_{HR} and Σ_{RH}. For the AB there is a low correlation between the robot and the human movement. This explains the behaviour seen in Figure 2.7.

2.3.2. INTERACTIVE PRIMITIVES BASED ON PROMP

ProMP is a probability distribution over the trajectories. ProMP allows to generate the covariance between trajectories of different DoF. This can be used to couple the joints of a robot or to couple the movement of the human to the movement of the robot.
The probabilistic movement primitives define a state variable y (e.g. the joint angle that needs to be controlled), similar to DMP. The state vector y(t) can be described by [35]:

$$y(t) = \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix} = \begin{bmatrix} \psi(t) \\ \dot{\psi}(t) \end{bmatrix} w + \epsilon_y, \tag{2.22}$$

where $\Psi(t) = [\psi(t), \dot{\psi}(t)]^T$ represents the time-dependent standard normalized Gaussian basis matrix, w the weights, ϵ_y is zero-mean i.i.d. Gaussian noise and q and $\dot{q}$ are the position and velocity of the different joints. For simplicity the remaining explanation of the process will be performed using only the position q.
The probability distribution over the weights w can be inferred when multiple demonstrations of q are given. Generally one Gaussian distribution is used to describe p($\mathbf{w}$). For each single trajectory a weight vector is created. The probability distribution $p(q_{1:T})$ over the trajectories $q_{1:T}$ (where $q_{1:T} = [q_1, q_2, ..., q_T]$) is obtained by integrating $\mathbf{w}$ out,

$$p(q_{1:T}) = \int p(q_{1:T|\boldsymbol{w}}) p(\boldsymbol{w}) d\boldsymbol{w}. \tag{2.23}$$

This is called a Probabilistic Movement Primitive (ProMP) [1].

3

DESIGN OF THE TP-INFERENCE APPROACH

Chapter 2 explained the background of task-parameterized movement representation, introducing the components from which the 'building blocks' within task-parameterized movement representation are created. The motion data, task parameters and the model are the components of each building block, were two components are given and the thirds is estimated. The building blocks are Learning, Movement retrieval and Task parameter inference. As indicated in Chapter 2, many research is found on the learning and movement retrieval and the combination of those two. Task parameter inference is a new area of research. The goal of this thesis is to investigate the possibility of implementing task parameter inference in human-robot interactive tasks. To do so, the TP-inference approach was created, where existing 'building blocks'—Learning and Movement retrieval—were combined with Task parameter inference.

3.1. PRINCIPLE

The idea of the design of the implementation of TP-inference in human-robot interaction, is that the robot observes the movement of the human, predicts the task parameters corresponding that movement, such that the robot can estimate what movement to make in response. Figure 3.1 illustrates the design of the implementation of TP-inference for a human-robot interaction and shows a combination of the two possibilities within the task-parameterized movement representation (see Figure 2.5).

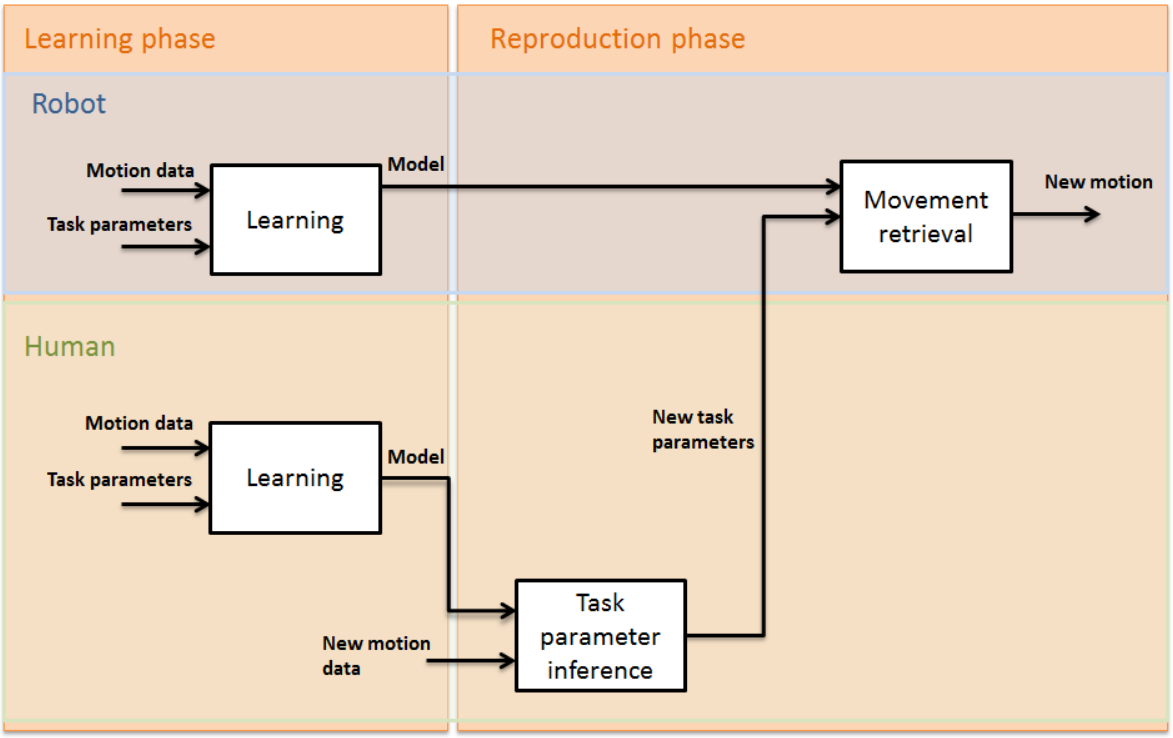


Figure 3.1: Diagram of the design of the TP-inference approach. The approach can be divided into four parts: The learning phase of the robot, the learning phase of the human, the TP-inference of the human movement and the movement retrieval of the movement of the robot. The model of the movement of the human and the model of the corresponding movements of the robot are created in the learning phase. In the reproduction phase the task parameters of the new movement of the human are inferred and used as an input to estimate the corresponding movement of the robot.

3.2. LEARNING PHASE

A task-driven model is used to represent the demonstrations data. In this case this entails that all the data is stored. The model is trained based on the demonstrations of the task, observed through multiple local frames. For the example given in Chapter 2, two local frames are considered: one frame at the location of the center of the torso of the human (p=1), and one frame at the position of the objects (p=2), as indicated in Figure 3.2 in pink and green respectively. The demonstration data can be summarized as $\{\mathbf{X}_n, \mathbf{A}_n^{(f)}, \mathbf{b}_n^{(f)}\}_{n=1,f=1}^{N,F}$. Here for each demonstration n, $\boldsymbol{X}_n$ is the movement in the global frame (in Cartesian space) and $\boldsymbol{A}_n$ and $\boldsymbol{b}_n$ are the task-parameters. The subscript N indicates the number of the demonstrations and F the number of frames. In the learning phase, the mean and covariance of the demonstration data is converted into weight values, to indicate which local frame has more importance in representing the data.

3.2.1. WEIGHT CALCULATION

To assure that the covariance matrix is strictly positive definite, the covariance matrix is reorganized. This is achieved by calculating the eigenvalues and eigenvectors of the matrix and redefining these **D** and **V** matrices, by only taking the highest eigenvalues and eigenvectors respectively and then recalculating the covariance matrix by:

$$\Sigma = \mathbf{V}\mathbf{D}\mathbf{V}^T + \mathbf{I}\lambda, \tag{3.1}$$

where λ is taken as a small number, in this case $1 \cdot 10^{-3}$.
A weight matrix is created by calculating the likelihood of the data points of all the demonstrations

to be generated by a Gaussian, parameterized by mean and (adjusted) covariance of Equation 3.1:

$$p(\mathbf{X}_n|\boldsymbol{\mu}_n^{(f)},\Sigma_n^{(f)}) = \frac{1}{\sqrt{(2\pi)^d|\Sigma_n|}}\exp(-\frac{1}{2}(\mathbf{X}_n-\boldsymbol{\mu}_n)^T\Sigma_n^{-1}(\mathbf{X}_n-\boldsymbol{\mu}_n)). \tag{3.2}$$

Here $\boldsymbol{X}_n$ are the data points of the trajectory, $\boldsymbol{\mu}$ is the mean and Σ the covariance of the Gaussian. The subscript n indicates the time step ($n \in 1:200$), f indicates which frame and d is number of dimensions. This equation can be used in this form because as is explained in the previous section, the Σ matrix is positive definite. The part, $(\mathbf{X}_n-\boldsymbol{\mu}_n)^T\Sigma_n^{-1}(\mathbf{X}_n-\boldsymbol{\mu}_n)$, represents the distance of the test point $\mathbf{x}$ from the mean $\boldsymbol{\mu}$. For the given example—with 4 demonstrations of 200 data point each—the weight is a 2×800 matrix (number of frames $\times$ the number demonstrations multiplied by the number of data points). The weight matrix is than normalized over 800 values, such that the summation the of these values results in 1. This weight matrix is stored as a part of the model parameters.

For the design introduced in Figure 3.1, two models are trained, one for the movement of the human and one for the movement of the robot. Each model is observed from two local frames.

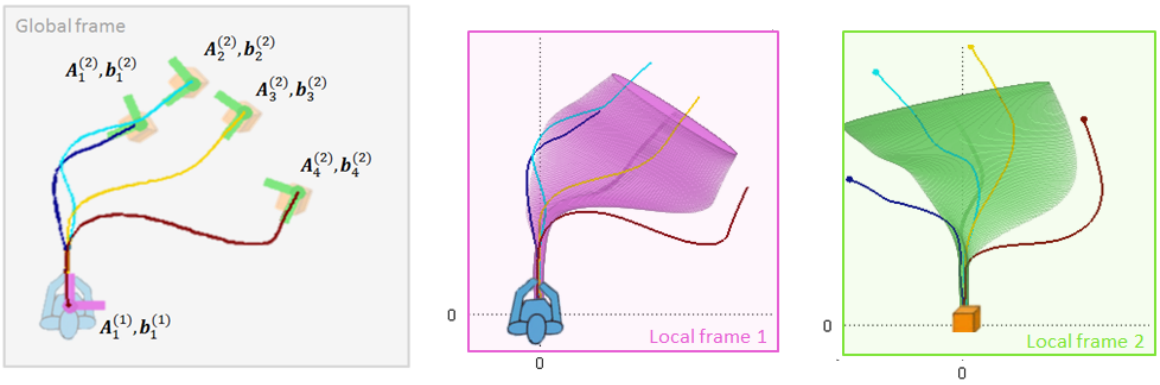


Figure 3.2: Task-driven model. The model is based on the mean and covariance of the demonstrations, calculated from the different local frames. The model is indicated as the pink and green shape in the second and third image respectively.

3.3. TASK-PARAMETER INFERENCE

The model of the movement of the human is used in combination with a new movement of the human to infer the task parameters. The weight matrix is used to adjust the model such that, when the orthogonal Procrustes analysis is executed, the importance of the different parts of the movement is taken into account. The Procrustes analysis is executed to infer the task parameters. These task parameters are then in combination with the robot model, given as an input to retrieve the robot's movement.

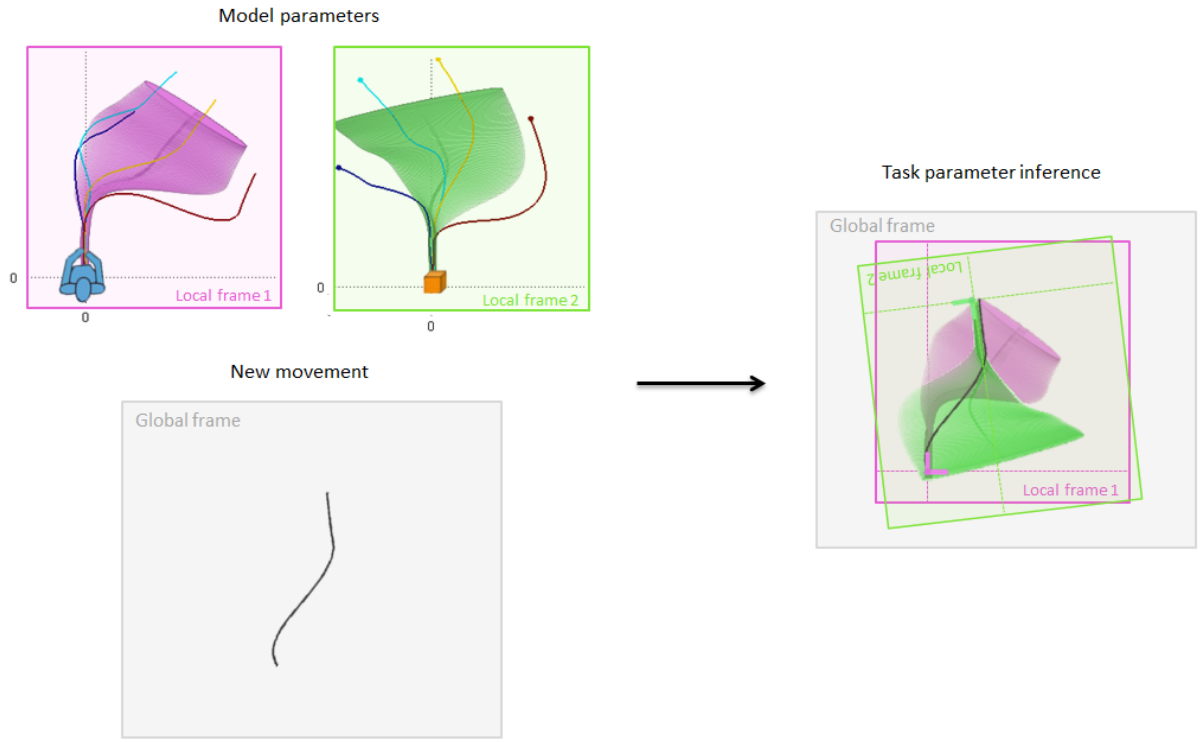


Figure 3.3: The orthogonal Procrustes analysis is used to match the models of the two frames (upper left image) with a new trajectory (lower left image) to infer the task parameters (right image).

3.4. MOVEMENT RETRIEVAL

The model of the robot is rotated and translated with the inferred task parameters to retrieve the movement of the robot

$$\hat{\boldsymbol{\mu}}_{t,i}^{(j)} = \mathbf{A}_{t,j}\boldsymbol{\mu}_i^{(j)} + \mathbf{b}_{t,j}, \tag{3.3}$$

$$\hat{\Sigma}_{t,i}^{(j)} = \mathbf{A}_{t,j}\Sigma_i^{(j)} = \mathbf{A}_{t,j}^T. \tag{3.4}$$

The model observed from the global frame is retrieved with the product of Gaussians

$$\hat{\boldsymbol{\mu}}_{t,i} = \hat{\Sigma}_{t,i}\sum_{j=1}^{P}\hat{\Sigma}_{t,i}^{(j)^{-1}}\hat{\boldsymbol{\mu}}_{t,i}^{j}, \tag{3.5}$$

$$\hat{\Sigma}_{t,i} = (\sum_{j=1}^{P}\hat{\Sigma}_{t,i}^{(j)^{-1}})^{-1}. \tag{3.6}$$

Performing linear regression will result in the most likely movement seen from the global frame.

4

BENCHMARK APPROACH: PROMP

Movement Primitive is a frequently used approach in PbD. Chapter 2 introduced the Interactive Primitive (IP), which is based on these Movement Primitive. The IP is used to model the interactive tasks, by storing the important aspects of the individual tasks as well as the important aspects of the interaction. The Probabilistic Movement Primitive (ProMP) used in the research of Erwerton et al. [1] is a variation on the IP and is chosen as benchmark to compare the TP-inference approach to.

4.1. PRINCIPLE

The benchmark approaches, to which the TP-inference approach is compared, is the Probabilistic Movement Primitive (ProMP). ProMP is an approach where primitives are determined that capture the correlation between the movement of two agents (the human and the robot). By observing the human, it is then possible to infer the movement robot, such that interactive task can be achieved. The ProMP approach applies probabilistic movement primitives to generalize the training data. In the next sections the two approaches will be discussed in more detail.

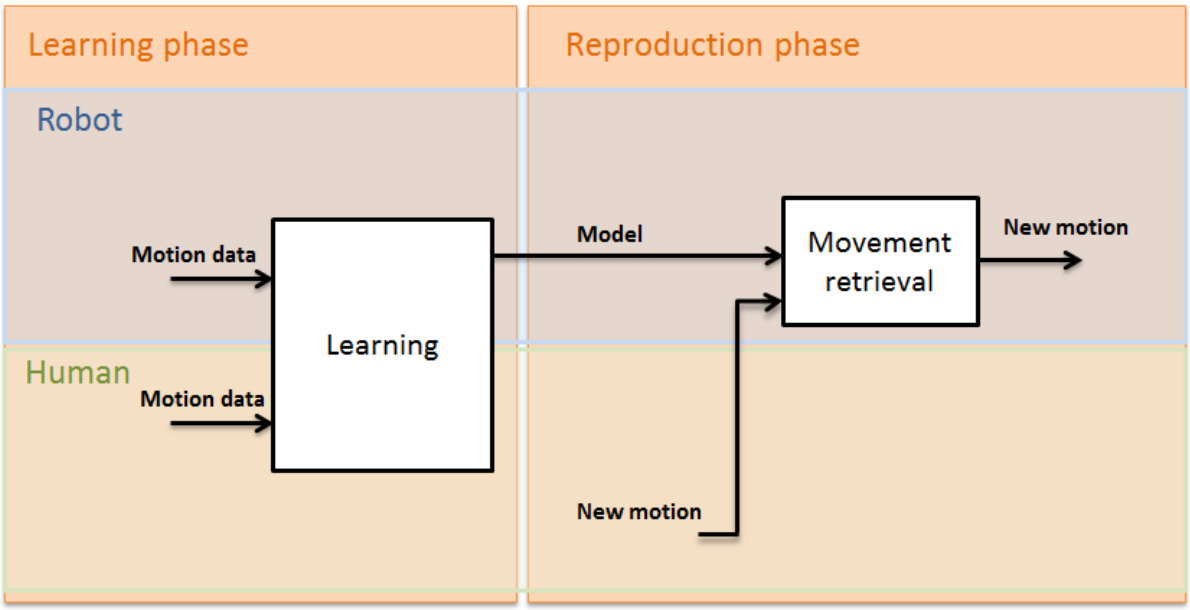


Figure 4.1: A schematic representation of the ProMP approach, to visualize the difference between the proposed problem statement of the thesis (Figure 3.1) and the ProMP approach. Here a model is made of the human and robot interaction movements. This is used in combination of new human movement data to infer the corresponding robot movement. In the learning phase a joint probability function is used to model the interactive task. In the reproduction phase the model, in combination with a new movement of the human, is used to retrieve the movement of the robot, by applying a regression method.

4.2. PROMP APPROACH

The ProMP approach discussed in this section, originates from the research of Maeda et al. [35] and Ewerton et al. [1]. This approach is an extension of the Interactive Primitive approach, described in the previous section. In the ProMP approach, the IP approach is adopted, with the difference that it is based on probabilistically modeling the interaction, using a distribution of the observed trajectories (probabilistic movement primitives). The method Ewerton et al. [1] propose is based on a Gaussian Mixture model.

4.2.1. PREDICTIVE DISTRIBUTION

To predict the movement of the human, after observing a partial trajectory, a probabilistic approach is used, where a distribution $p(\boldsymbol{q}_{1:T}|D)$ over the parameters of the trajectories is modeled

$$p(\boldsymbol{q}_{1:T}|D) = \int p(\boldsymbol{q}_{1:T}|\overline{\boldsymbol{w}})\, p(\overline{\boldsymbol{w}}|D)\, d\overline{\boldsymbol{w}}, \tag{4.1}$$

with D the given observations of the human, $\boldsymbol{q}_{1:T}$ the trajectories of the human and the robot (were T indicates the time) and $\overline{\boldsymbol{w}}$ is a vector containing the weights of the human and the robot for one demonstration.
The conditional distribution can be described as Gaussian with a mean and variance. A posterior distribution after observing D is obtained with:

$$\begin{aligned} \boldsymbol{\mu}_w^{new} &= \boldsymbol{\mu}_w + \boldsymbol{\Sigma}_w \boldsymbol{H}_t^T (\boldsymbol{\Sigma}_D + \boldsymbol{H}_t^T \boldsymbol{\Sigma}_w \boldsymbol{H}_t)^{-1} (D - \boldsymbol{H}_t^T \boldsymbol{\mu}_w), \\ \boldsymbol{\Sigma}_w^{new} &= \boldsymbol{\Sigma}_w - \boldsymbol{\Sigma}_w \boldsymbol{H}_t^T (\boldsymbol{\Sigma}_D + \boldsymbol{H}_t^T \boldsymbol{\Sigma}_w \boldsymbol{H}_t)^{-1} (\boldsymbol{H}_t^T \boldsymbol{\Sigma}_w), \end{aligned} \tag{4.2}$$

where $\boldsymbol{\Sigma}_D$ is the observation noise and $\boldsymbol{H}_t$ is the observation matrix, with on its diagonal the basis functions for the observed states and zeros for the states of the robot.

4.2.2. Phase estimation with dynamic time warping

Phase Estimation is implemented to temporally align the movements of the human and the robot. A Dynamic Time Warping (DTW) algorithm is used to determine the phase variable x. Given two time dependent data sets (one of the human movement and the other of the robot), DTW finds the optimal correlations between the data points, hereby minimizing the distance function D [6]. Phase estimation is used in the reproduction phase to predict the movement of the human, when just observing a part of the movement. This makes it possible for the robot to start its corresponding movement, while the movement of the human is still not finished.

5

EXPERIMENTAL SETUP

To test and compare the TP-inference approach with the ProMP approach, a couple of tasks were recorded by using OptiTrack. To limit the scope of the thesis, the focus of the comparison between the TP-inference approach and the ProMP approach, was on tasks where the movement of the human is fully observed, before the robot starts performing its response. A simple static interactive task—such as dealing with changing position and orientation in the reproduction phase, showing longer of shorter movements, inferring the correct aspect of the task—already provides a lot of challenges. In this chapter the choice of the tasks and the setup is described.

5.1. CHOICE OF TASKS

Four different situations were simulated: placing cups on top of each other (Cuppile), building a tower of 3 cups (Tower), pouring a drink (Drink) and placing a signature (Signature). The four static interactive tasks were constructed, with varying complexity to investigate the boundaries of the approach.

5.1.1. TASK 1: CUPPILE

The first task, the cup piling scenario, a simple similar movement was chosen to test the performance of the TP-inference approach and to compare the limitations of the TP-inference approach and the ProMP approach, when it comes to orientation and scaling of the movements. The task was recorded such that the movement ended at the end position of the cup and thereby making, in this case, the end position the most important aspect during the movement.

5.1.2. TASK 2: TOWER

The first task was expanded to a non similar movement in the second task, the building a tower scenario. For this task the movement of the human, placing two cups on the table and moving the hand back to neutral position, is recorded. Then the movement of the robot, placing a cup on top of the two cups and moving the hand back to neutral position, is also recorded. In this case the important aspect of the task is in the middle of the movement. The task was chosen to compare if the two approaches handle an interactive task that has a less similar movement and were the task parameters—that have to be inferred—are placed in the middle of the movement.

5.1.3. TASK 3: DRINK

The third task, pouring a drink, was chosen to illustrate a completely different task, where the complexity lies in observing the trajectory of the bottle rotation and inferring the place of the cup. The previous two tasks differ in complexity, but only in 2 dimensions. Observing the movement toward

the cup, the sharp curve movement, when pouring the drink and then returning to neutral position, is a movement with complexity in 3 dimensions.

5.1.4. TASK 4: SIGNATURE

In the last task, placing a signature, the complexity lies in the orientation of the signature and the signing itself. The movement of the robot is increased instead of the movement of the human. The movement of the human is recorded until the location of the signature and the movement of the robot until the signature is done. The challenge of this task is to model the components of the movement, such that the signature can be replicated correctly and reproduced in a new situation.

5.2. SETUP

The four tasks were recorded with a motion capturing system (OptiTrack) in the same setup, using a table, two chairs, a couple of cups and a bottle, see Figure 5.1. Four to five OptiTrack markers were used to record the tasks. Three of the markers were used to identify the different frames and one (Cuppile and Tower) or two (Drink and Signature) markers to track the hand of the demonstrator.

Figure 5.1: Setup used to record the demonstrations. Three local frames were attached to the table, the local frame for the human perspective, the local frame for the robot and one for the perspective of the object. One OptiTrack marker was attached to the back of the hand to record the movements.

Because the TP-inference approach is modeled in a Cartesian coordinate system, the tasks were created in a way that there would be no necessity to compensate for the hand orientation changes during the movement. This is the reason for choosing to work with cups instead of blocks. The orientation of the cup itself in the xy-axis is not relevant.

Another problem that occurs when recording the movement in Cartesian space, is that the location of the cup can not be accurate described due to the flexion of the wrist. To overcome this problem, the center of the hand marker is translated to the center of the cup, before starting the demonstrations, see Figure 5.2. Because of this center translation, two markers were used in the Drink and Signature task. In the Drink task the center is translated to the pouring end of the bottle for the movement of the human. In the Signature task the center is translated to the

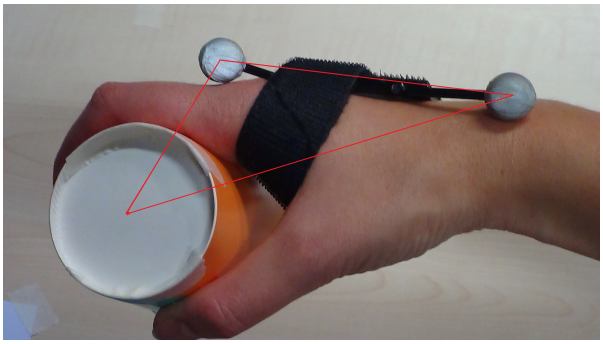


Figure 5.2: Image of the OptiTrack marker on the hand with a translated center.

position of the thumb and index finger. During pretesting this solution gave the most smooth and accurate recordings of the demonstrations.
The use of Cartesian space in the TP-inference approach limits the research to position changes only. To extent the research to orientation changes in the movement, it is recommended to transform the TP-inference approach to use a quaternion system. This would also result in more realistic demonstrations of the task, because it eliminates the necessity of the center translation of the marker.
In appendix A is given a detailed description of the setup and the performed demonstrations.

6

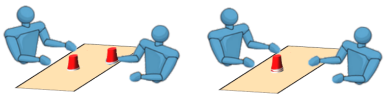
EXPERIMENTAL RESULTS

In this section the results of the four different static tasks and the analysis of the comparison between the TP-inference approach and the ProMP are presented.
The tests that were performed for each task are:

- Test 1: Given 9 demonstrations, retrieve one of them in the reproduction
- Test 2: Given 8 demonstrations, retrieve a new movement in the reproduction

Test 1 is performed to investigate the reproduction performance of the demonstration data and Test 2 to test the performance of the approaches in new situations. These tests were performed for the nine variations per position.
For each of the tests the error deviation at the position of the object, the root mean square error of the trajectory and the smoothness of the trajectory were calculated. The error deviation gives an indication if the important aspect of the task (e.g. the position of the cup on the table) is inferred correctly and if the task will succeed. The root mean squared error of the trajectory is calculated to investigate if the trajectory is reproduced in a similar manner as the demonstrations. To investigate how smooth the trajectory is performed and to analyze the movements, the smoothness is calculated.

6.1. LEARNING TASKS: CUPPILE



In the first task, where two cups are placed on top of each other, the movement of the human and the robot are quite similar. This task was chosen to focus the comparison on the precision and accuracy of the retrieved trajectory and the end position.
Figure 6.1 is a schematic representation of the demonstrations, to support the results.

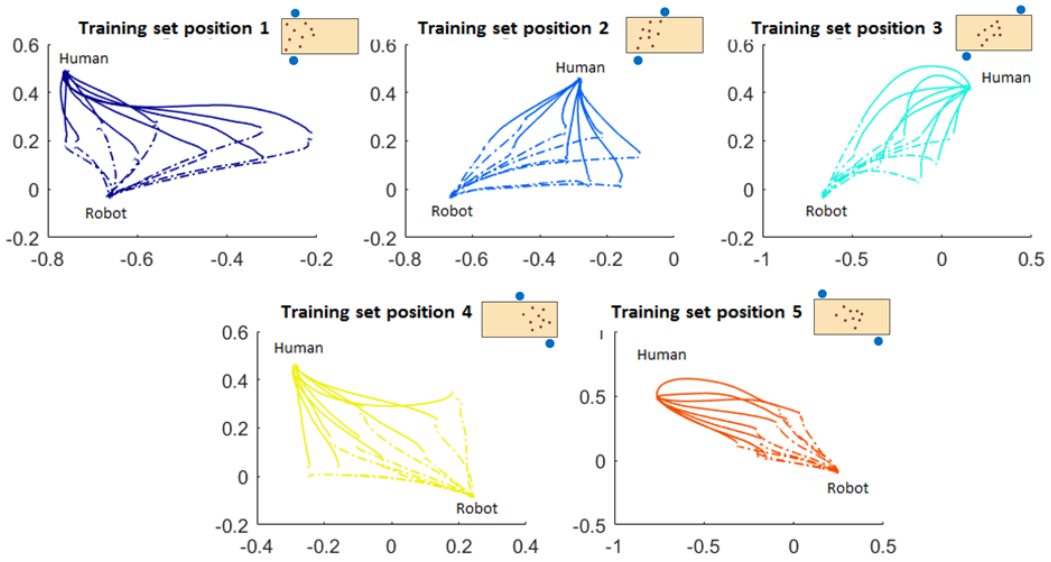


Figure 6.1: Demonstrations of the Cuppile task, seen from above. Nine demonstrations are performed for each position. At the upper right corner of each graph a schematic representation of the setup is given. The orange square represent the table. The blue circles represent the position om the human and robot, and the red the position of the cups.

6.1.1. ERROR DEVIATION OF THE END POSITION

Figure 6.2 illustrates the errors of the end position. Test 1 shows that for the position 3 the performance of the TP-inference had a high variance in the outcome. This behavior can be explained by observing Figure 6.1, training set position 3. Here the demonstrations of the human movement given in the middle, are shorter and more curvy in comparison with the outer demonstrations. The short curvy trajectories and the overall variety of trajectories, in this situation, can not be represented correctly with the task-driven model, see Figure B.1. The outliers in position 2, 4 and 5 could also be explained by looking at the difference in trajectory shapes of the nine demonstrations. At these positions the trajectories of the demonstrations are more similar to each other, except for one or two trajectories. In position 2, for example, there are two demonstrations of the human movement that are much bigger than the rest of the demonstrations. Figure 6.3 shows, the orientation that best illustrates the deviation of the demonstration of the outlier (observed in figure 6.2). For the other positions see Appendix B.1.1.

In Test 2, the variance of the outcome of the TP-inference and the ProMP approach is higher than in Test 1, excluding the variance outcome of TP-inference approach at position 3. In this case the variance is smaller but with an extreme outlier at 8.7 cm. The TP-inference has in position 2 to 4 extreme outliers. This could be explained by the choice of demonstrations. The outlier could be a result of a trajectory—that had to be retrieved—outside the range of the model. An example is given in Figure 6.4, which shows the demonstrations given in black. The trajectory used for the reproduction is illustrated in red. The human movement of the reproduction is the shortest trajectory and lies outside the range of the demonstrations. For both approaches it is therefore difficult to correctly reproduce the movement.

The comparison of the errors of Test 1 and Test 2, shows that the variance of the outcome of Test 2 is increased compared to Test 1.

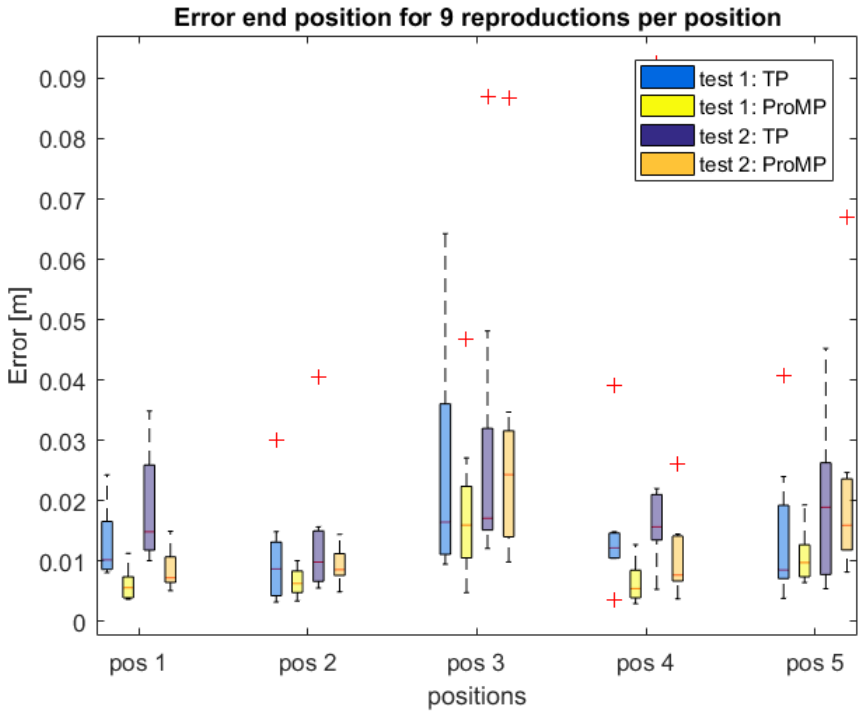


Figure 6.2: The error deviation of the end position, for the Cuppile task. Each position consists of nine trajectories. For each trajectory a calculation was made of what the error of that trajectory, if it was given as input in the reproduction and all 9 trajectories or 8 trajectories (all the demonstrations minus the one used in the reproduction) were given in the learning phase.

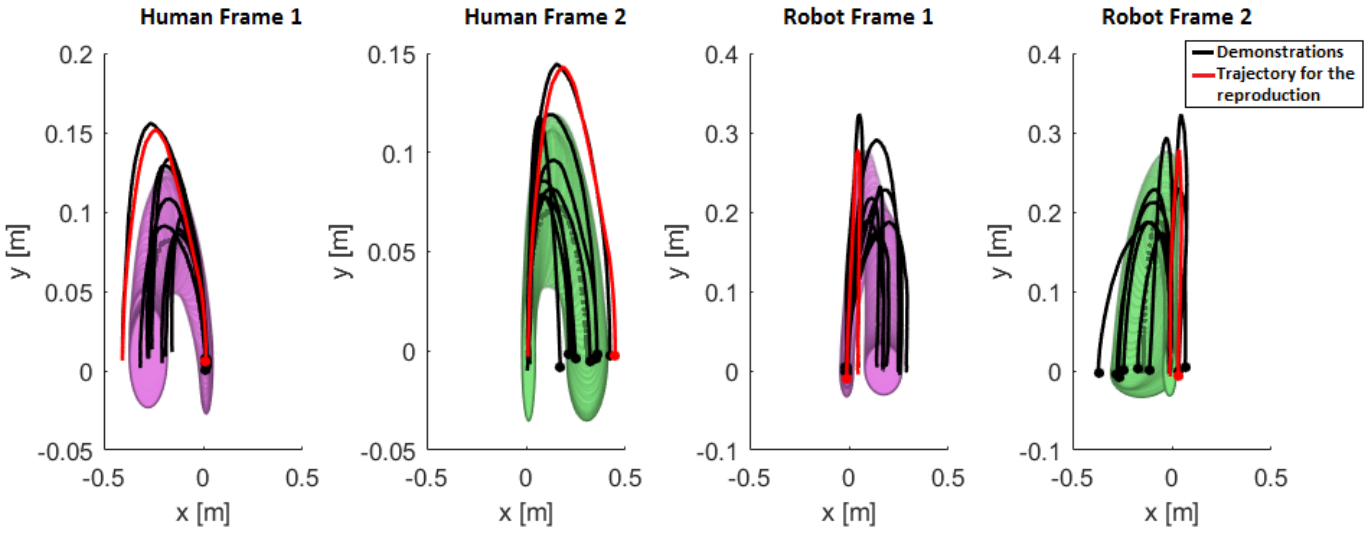


Figure 6.3: An illustration of Cuppile task for Test 1, at position 2. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory is indicated which gave the outlier, seen Figure 6.2 position 2. For the movement of the human this red trajectory has a higher overshoot in vertical direction than the other demonstrations.

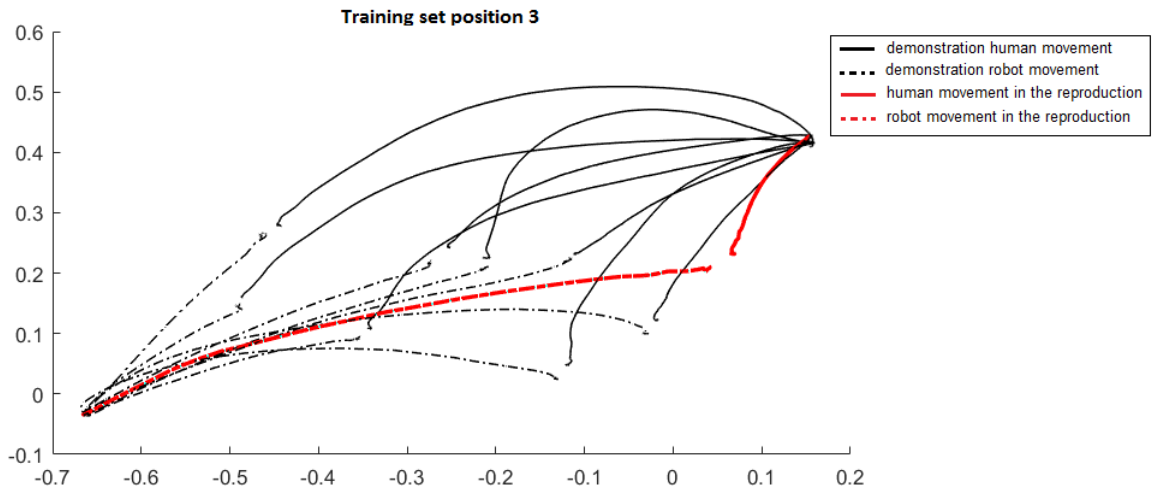


Figure 6.4: The trajectories used in Test 2 for the Cuppile task, position 3. The black lines indicate the demonstrations and the red is the trajectory given in the reproduction phase. The demonstrated trajectories of the human are much longer the trajectory shown in the reproduction phase.

6.1.2. ROOT MEAN SQUARED ERROR TRAJECTORY

The root mean squared error of the trajectory is calculated to gain an indication of how well the trajectory of the robot is reproduced, in comparison with the original movement. In Figure 6.5 the root mean squared error of both Test 1 and Test 2 are displayed. For Test 1, in the first two positions, the results for both approaches are quit similar. In the other three positions the ProMP approach performs better and with a lower variance in outcome. The outcome for the ProMP approach, over all the positions, is quit consistent. There is a high correlation between the movement of human and robot, because of the simplicity and the similarity of the movements.

In Test 2, the variances of the outcome for both approaches are increased. Positions 1, 2 and 4 have the highest variance increase. This could be explained by the wide range of locations that are demonstrated for these three positions. When one of the nine trajectories is not given, during the learning phase the size of the Gaussians describing the model decreases (in both the TP-inference and ProMP approach). This could therefore result in less adequate mapping between the model and the trajectory that is left out. See Appendix B.1.2 for the figures regarding these position for both the TP-inference approach and the ProMP approach.

When comparing the results of the two tests, it can be observed that the overall performance is deteriorated with an error of 2 to 4 cm for both approaches, and in position 1 even more, with circa 9 cm for the TP-inference approach.

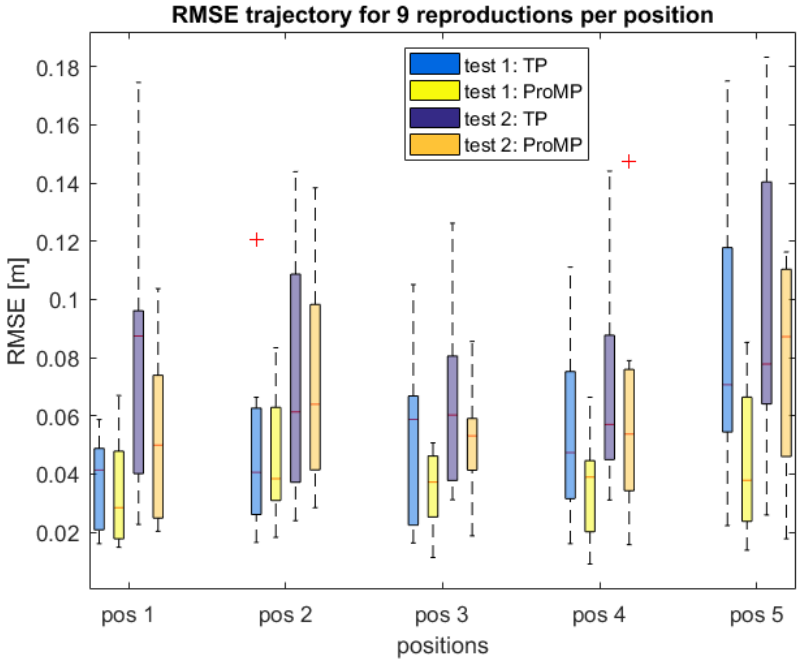


Figure 6.5: The root means squared error of the trajectory, for the Cuppile task. Each position consists of nine trajectories. For each trajectory a calculation was made, of the error of that trajectory, if it was given as input in the reproduction and all 9 trajectories or 8 trajectories (all the demonstrations minus the one used in the reproduction) were given in the learning phase.

6.1.3. SMOOTHNESS OF THE TRAJECTORY

Figure 6.6 represents the smoothness of the trajectories for position 1. The other smoothness figures can be found in Appendix B. The smoothness was calculated to investigate the jerk behavior of the retrieved trajectories, but also to observe the anomalies in the data. Both approaches smoothen the trajectories more in comparison with the original data; the trajectory jerk of the TP-inference approach is more similar to the original data. To smoothen the trajectory can be beneficial, but it could also result in the loss of information. For this task, because a simple smooth movement towards the cup is required, the loss of information will be minimal.

Observing the smoothness for Test 2, it can be seen that for the TP-inference approach some of the trajectories have a higher jerk behavior than in Test 1, and that these trajectories mimic the peaks in the acceleration rate of the given trajectories, see Appendix B.

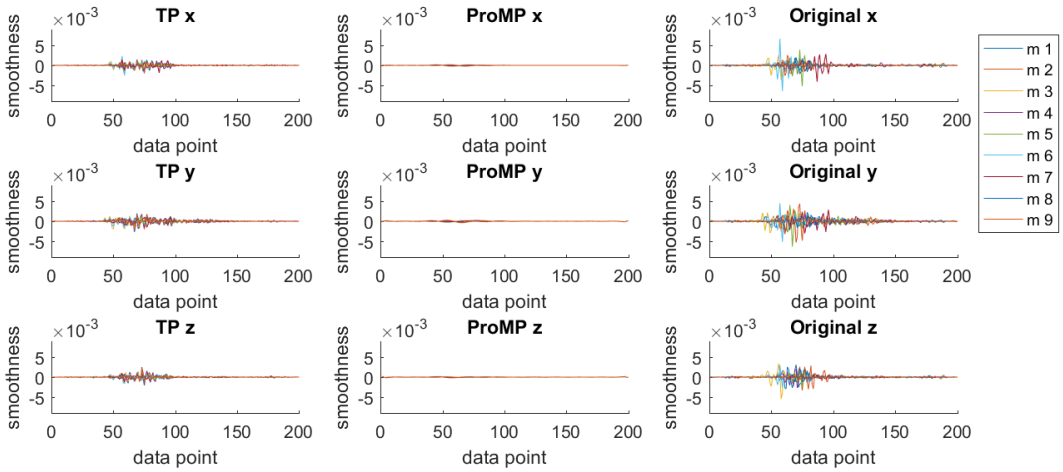


Figure 6.6: The smoothness per trajectory (m) in Test 1 for the Cuppile task at position 1. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

6.1.4. CONCLUSION

For Test 1 the end position errors, taken over all the position, have a mean and standard deviation of: $\mu = 0.0156$ m and $\sigma = 0.0125$ m and $\mu = 0.0096$ m and $\sigma = 0.0077$ m, for the TP-inference and the ProMP approach respectively. For Test 2, the error of the end position is slightly higher: $\mu = 0.0186$ m and $\sigma = 0.0086$ m and $\mu = 0.0089$ m and $\sigma = 0.0036$ m, for the TP-inference and the ProMP approach respectively.

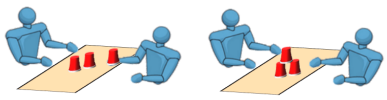
The root mean squared error of the trajectory in Test 1, taken over all the position, has a mean and standard deviation of: $\mu = 0.0550$ m and $\sigma = 0.0346$ m and $\mu = 0.0386$ m and $\sigma = 0.0197$ m, for the TP-inference and the ProMP approach respectively. In test 2 these values increase with circa 2 cm, $\mu = 0.0761$ m and $\sigma = 0.0475$ m and $\mu = 0.0521$ m and $\sigma = 0.0301$ m, for the TP-inference and the ProMP approach respectively.

For this task the values of the end position is most important. The robot will not be able to place the cup on top of the other cup if the retrieved coordinates deviate to much. If the error is lower than 1 cm the Cuppile will still succeed. When the error is between 1 and 2 cm the chance of succeeding is low, and when the error is higher than 2 cm the task will not succeed, due to passing the radius of the cup (2.5 cm).

For the ProMP approach, the task in both Test 1 and 2 will most likely succeed for positions 1, 2 and 4. For position 3 and 5 the task will most likely false.

For the TP-inference approach, the task will mostly likely succeed for both Test 1 and 2, position 2. Position 1 and 4 have a lower chance of succeeding and position 3 and 5 will most likely fail.

6.2. LEARNING TASKS: TOWER



The second task, building a tower of cups together, is an extension of the Cuppile task. The human places two cups next to each other and the robot has to place a cup on top, in the middle of the two cups. In the task the parameters that need to be inferred, from observing the movement of the human, are somewhere in the middle of the movement instead at the end.

Figure 6.7 shows a schematic representation of the demonstrations, to support the results.

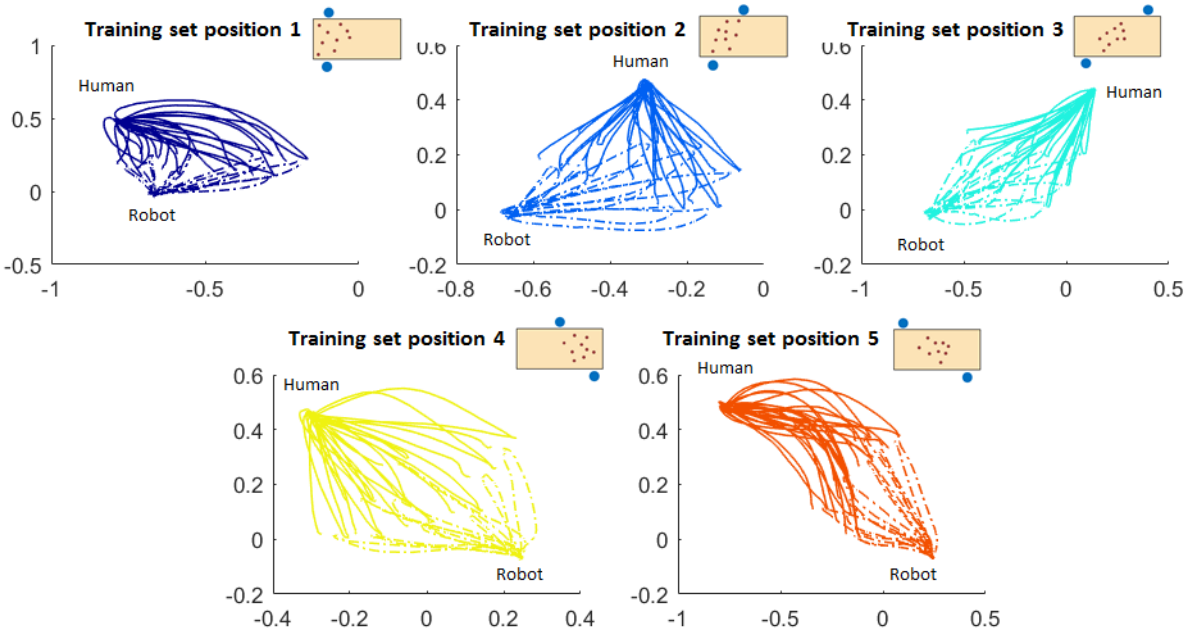


Figure 6.7: Demonstrations of the Tower task seen from above. Nine demonstrations are performed for each position. At the upper right corner of each graph a schematic representation of the setup is given. The orange square represent the table. The blue circles represent the position om the human and robot, and the red the position of the cups.

6.2.1. ERROR DEVIATION OF THE MID POSITION

In Figure 6.8, the errors for the cup position are given. For Test 1 the overall performance of the ProMP approach is higher in comparison with the TP-inference approach. For the TP-inference approach the variance of the outcome is smaller in positions 4 and 5, compared to the other positions, but still has a mean error of circa 2 cm. At position 3 there is an extreme outlier for the TP-inference approach of 8 cm error. The trajectory responsible for this outlier is a trajectory that differs from the other demonstrations, see Figure B.19.

For Test 2, the error slightly increases for both approaches. The outlier observed in Test 1, at position 3 is the same outlier observed in Test 2, but increased to 13 cm.

The ProMP approach performs better than the TP-inference approach, even tough the movements of the human and the robot are different.

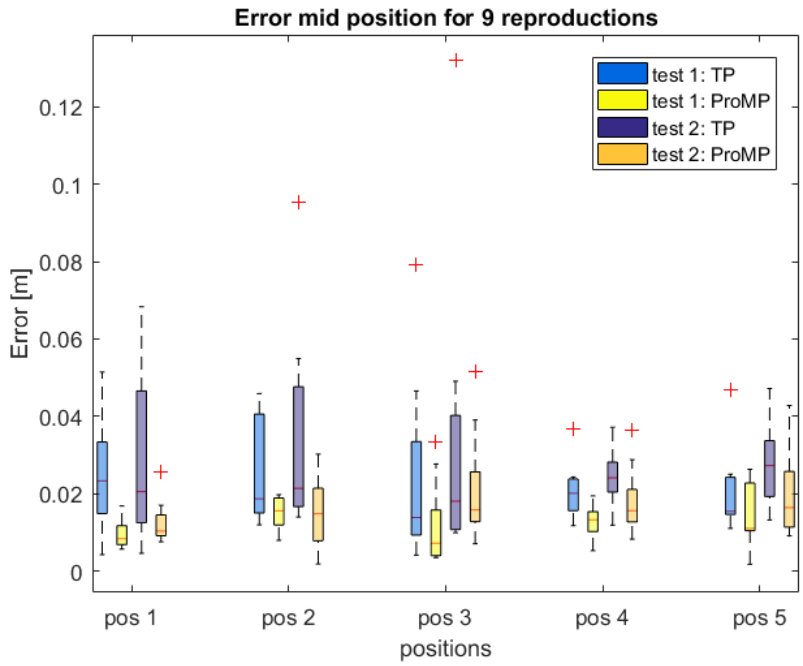


Figure 6.8: The error deviation of the end position, for the Tower task. Each position consists of nine trajectories. For each trajectory a calculation was made, of the error of that trajectory, if it was given as input in the reproduction and all 9 trajectories or 8 trajectories (all the demonstrations minus the one used in the reproduction) were given in the learning phase.

6.2.2. ROOT MEAN SQUARED ERROR OF THE TRAJECTORY

In Figure 6.9, the root mean squared error of the two test are displayed. In Test 1 the overall performance of the ProMP approach is slightly higher in comparison with the TP-inference approach. The performance of the TP-inference approach is the lowest at position 2, regardless that the error of the end position is comparable to position 1 and 3, see Figure 6.8. For this position the demonstrations differ too much from each other and are therefore the trajectories are not represented correctly by the model, see Figure B.20.

For Test 2, the results for both approaches, except for position 3, lie in the same range, but most reproductions for the ProMP approach have a lower mean value and standard deviation. For position 3 the root mean squared error for the TP-inference approach is increased with an maximum value of 16.5 cm. The problem in recording the Tower task could be that each position also differs in orientation (see Figure A.1), resulting in shorter and longer curves in the human movement, which makes the trajectories more difficult to model correctly. A solution could be to add an extra local frame at the position of the second cup.

Comparing the outcome of Test 1 and Test 2, the root mean squared error in Test 2 is increased for both approaches. The movement of the human is more complex than in the Cuppile task. This in combination with wide spread positions of the demonstrations, results in a higher error for Test 2.

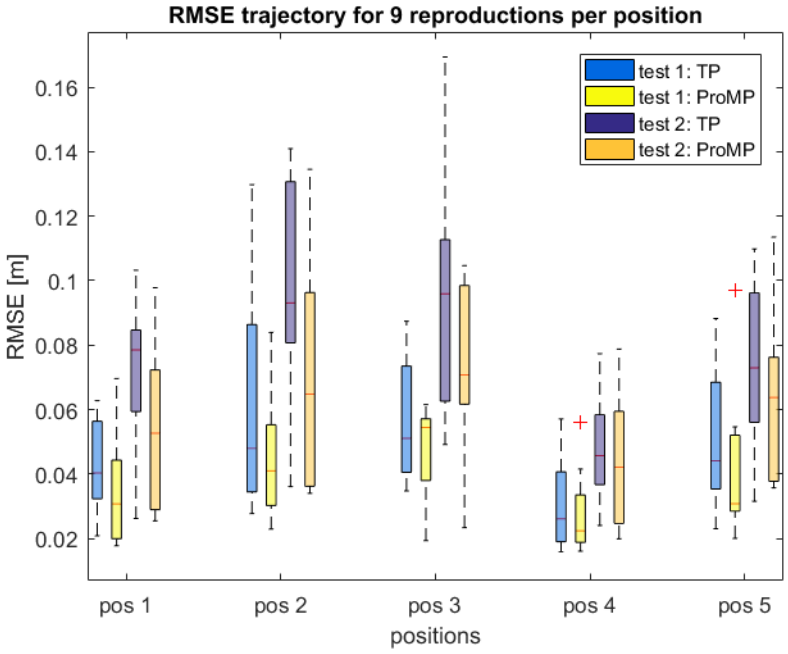


Figure 6.9: The root means squared error of the trajectory, for the Tower task. Each position consists of nine trajectories. For each trajectory a calculation was made, of the error of that trajectory, if it was given as input in the reproduction and all 9 trajectories or 8 trajectories (all the demonstrations minus the one used in the reproduction) were given in the learning phase.

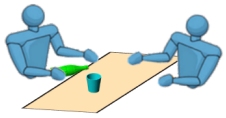
6.2.3. SMOOTHNESS OF THE TRAJECTORY

The figures regarding the smoothness of the trajectories can be found in Appendix B. Both The TP-inference and ProMP approach smoothen the trajectories more in comparison with the original data. The peaks in the change of acceleration rate, of the TP-inference approach is more similar to the original data. For this task, 2 points in the trajectory have an amplitude in the smoothness graph, this is due to the back an forth movement of the robot. The amplitude is at the highest point of lifting up the cup to the correct position, and lifting up the hand towards the neutral position. These amplitudes are also visible, but small for the ProMP approach. For Test 2, the peaks in the acceleration rate slightly increases for both approaches.

6.2.4. CONCLUSION

For Test 1, the cup position errors, taken over all the position, have a mean and standard deviation of: $\mu = 0.0244$ m and $\sigma = 0.0145$ m and $\mu = 0.0098$ m and $\sigma = 0.0042$ m, for the TP-inference and the ProMP approach respectively. For Test 2: $\mu = 0.0295$ m and $\sigma = 0.0228$ m and $\mu = 0.0128$ m and $\sigma = 0.0056$ m, for the TP-inference and the ProMP approach respectively.
The root mean squared error of the trajectory in Test 1, taken over all the position, have a mean and standard deviation of: $\mu = 0.0485$ m and $\sigma = 0.0246$ m and $\mu = 0.0399$ m and $\sigma = 0.0189$ m, for the TP-inference and the ProMP approach respectively. In Test 2 these values increase with circa 1 to 4 cm: $\mu = 0.0715$ m and $\sigma = 0.0244$ m and $\mu = 0.0536$ m and $\sigma = 0.0248$ m, for the TP-inference and the ProMP approach respectively.
For this task, the values of the position of the cup are most important. The robot will not be able to place the cup on top of the other two cups, if the retrieved coordinates deviate too much. If the error is lower than 1 cm, the Tower will still succeed. When the error is between 1 and 2 cm, the chance of succeeding is low and if it is higher than 2 cm, the task will most likely not succeed, depending on the direction of the error.
Only the ProMP approach in Test 1 has a slight change of succeeding to build the tower for position 1 and 3.

6.3. LEARNING TASKS: DRINK



In the third task, the human pours a drink and the robot picks up the cup. The movement of the human is a back and forth movement. The important aspects of the tasks are at the middle of the pouring movement.

6.3.1. ERROR DEVIATION OF THE MID POSITION

Figure 6.10 presents the error of the inferred position of the cup. For Test 1, the overall performance of inferring the accurate position of the cup, is for the TP-inference approach much higher compared to the ProMP approach. Observing the model used by the ProMP approach for this task (see Figure 6.11), seems to have a low variance for the human movement on the y-axis. The amplitudes visible at time step 0.25 and 0.75 are the upward movements, from the neutral position to the cup and vice versa. The amplitude at 0.5 is the moment when the drink is poured. When observing the movement from x and y-axis, the trajectory indicates a small deviation at the time of pouring, but due to the high variance at this point, the ProMP is not able the represent the location of the cup.

Test 2 shows similar results in comparison with Test 1.

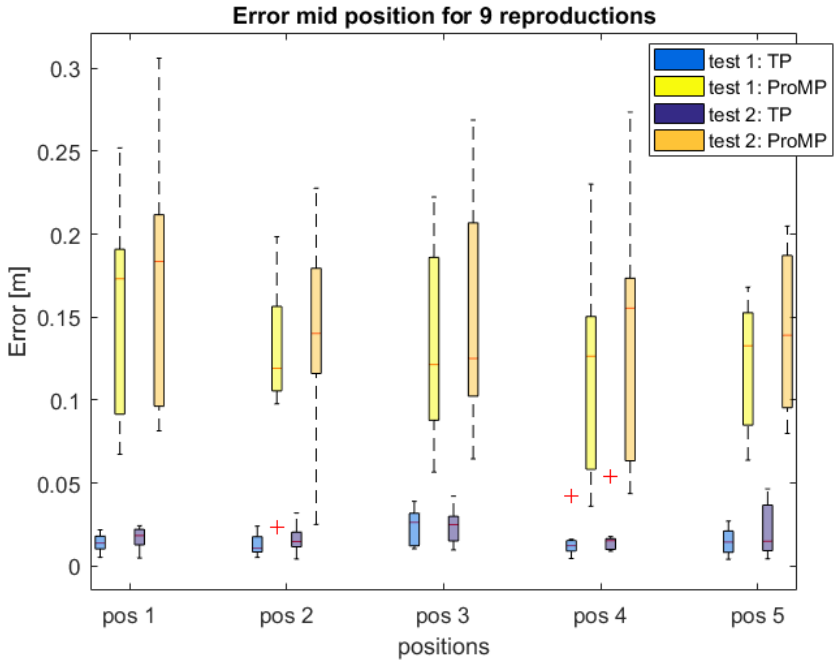


Figure 6.10: The error deviation of the end position, for the Drink task. Each position consists of nine trajectories. For each trajectory a calculation was made, of the error of that trajectory, if it was given as input in the reproduction and all 9 trajectories or 8 trajectories (all the demonstrations minus the one used in the reproduction) were given in the learning phase.

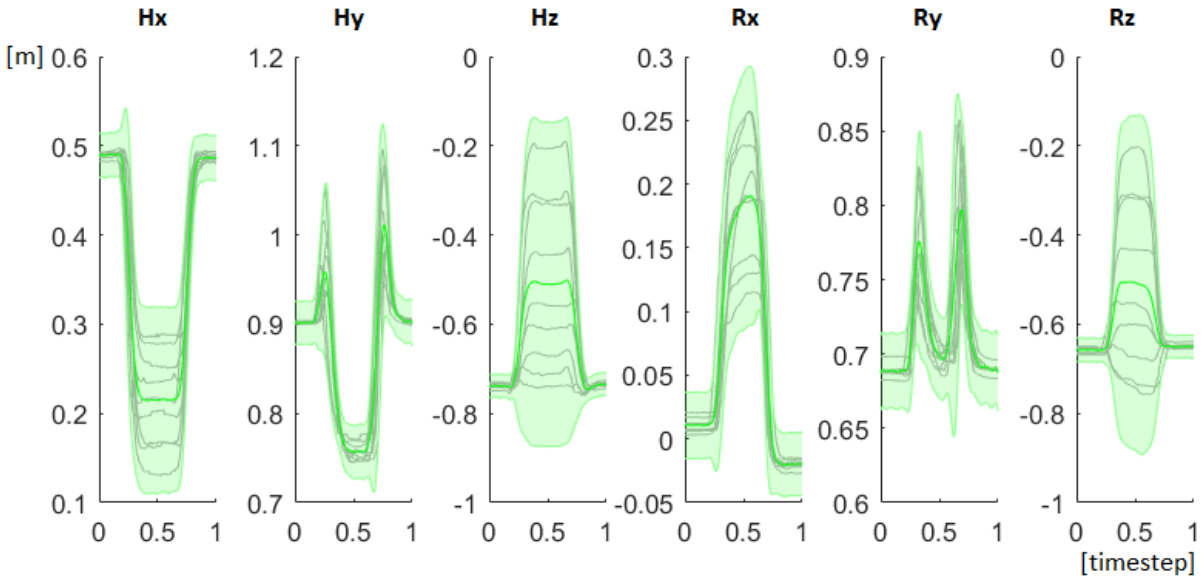


Figure 6.11: Model used by the ProMP approach for the Drink task, for position 1. The green area indicates the model and the grey lines show the demonstrations used to generate this model.

6.3.2. ROOT MEAN SQUARED ERROR OF THE TRAJECTORY

Figure 6.12 presents the root mean squared error of the trajectory for both Test 1 and Test 2. For Test 1, the performance of the TP-inference approach is higher compared to the ProMP approach. The performance of the TP-inference approach is at its highest at position 1, with an error of circa 2,5 cm. Even though not all the demonstrations are within the range of the model, the trajectories are evenly distributed in shape, resulting in a small deviation of the reproduction, see Figure B.32. At position 2, TP-inference approach has one outlier, due to the dissimilarity of this trajectory compared to the other demonstrations.

For Test 2 the TP-inference approach performs less in comparison with Test 1, but still better than the ProMP approach. Even though the performance of the TP-inference over the trajectory has decreased, the performance of inferring the task parameters remained similar.

6.3.3. SMOOTHNESS OF THE TRAJECTORY

The figures regarding the smoothness of the trajectories can be found in Appendix B. For this task the observations are comparable to those of the Tower task, because the movement of the robot in both tasks is similar.

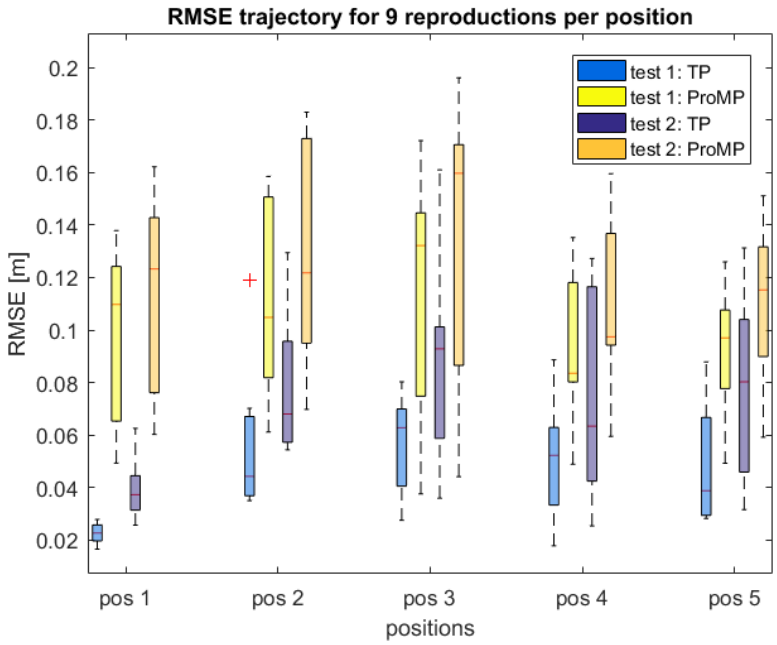


Figure 6.12: The root means squared error of the trajectory, for the Drink task. Each position consists of nine trajectories. For each trajectory a calculation was made, of the error of that trajectory, if it was given as input in the reproduction and all 9 trajectories or 8 trajectories (all the demonstrations minus the one used in the reproduction) were given in the learning phase.

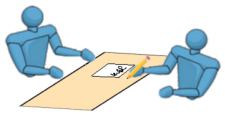
6.3.4. CONCLUSION

For Test 1, the cup position errors, taken over all the position, have a mean and standard deviation of: $\mu = 0.0135$ m and $\sigma = 0.0055$ m and $\mu = 0.1560$ m and $\sigma = 0.0616$ m, for the TP-inference and the ProMP approach respectively. For Test 2, the error of the mid position is similar: $\mu = 0.0174$ m and $\sigma = 0.0063$ m and $\mu = 0.1738$ m and $\sigma = 0.0729$ m, for the TP-inference and the ProMP approach respectively.

The root mean squared error of the trajectory in Test 1, taken over all the position, has a mean and standard deviation of: $\mu = 0.0463$ m and $\sigma = 0.0231$ m and $\mu = 0.1017$ m and $\sigma = 0.0342$ m, for the TP-inference and the ProMP approach respectively. In Test 2 these values increase with circa 1 to 4 cm for the TP-inference. The mean and standard deviation taken over all the position for Test 2 are: $\mu = 0.0385$ m and $\sigma = 0.0114$ m and $\mu = 0.1145$ m and $\sigma = 0.0383$ m, for the TP-inference and the ProMP approach respectively.

For this task the important aspect is the position of the cup. By observing the human pouring the drink, the robot can infer the location of the cup. For both tests the TP-inference approach is most likely to succeed in picking up the cup, but not with high certainty, because the error is still high for this task (4 cm). The ProMP approach will certainly not be able to perform the task.

6.4. LEARNING TASKS: SIGNATURE



In the fourth task, placing a signature, the human points to a specific position and the robot writes a name at that position. This is the first task in which the movement of the robot is more complex than the movement of the human. For this task, the correct reproducing the signature is most important, therefore the tests are analyzed by using the root mean squared error of the trajectory and the smoothness only.

6.4.1. ROOT MEAN SQUARED ERROR OF THE TRAJECTORY

Figure 6.13 represents the root mean squared error of both Test 1 and Test 2. For Test 1 the TP-inference approach performs better in comparison with the ProMP approach. The reason why the ProMP approach performs insufficiently in this task is because the ProMP model does not represented correctly all the curvy lines of the signature (see Rx and Ry in Figure 6.14). The variance at Rx and Ry are to high. The shape of the signature disappears.

In Test 2, the TP-inference approach also performs better in comparison with the ProMP approach. The performance decreased, but not in the same order as for the ProMP approach.

The root mean squared error gives an indication if the trajectory resembles the original movement, but it does not show if the signature is correctly reproduced. To check the signature, a plot of a reproduction is given in Figure 6.15.

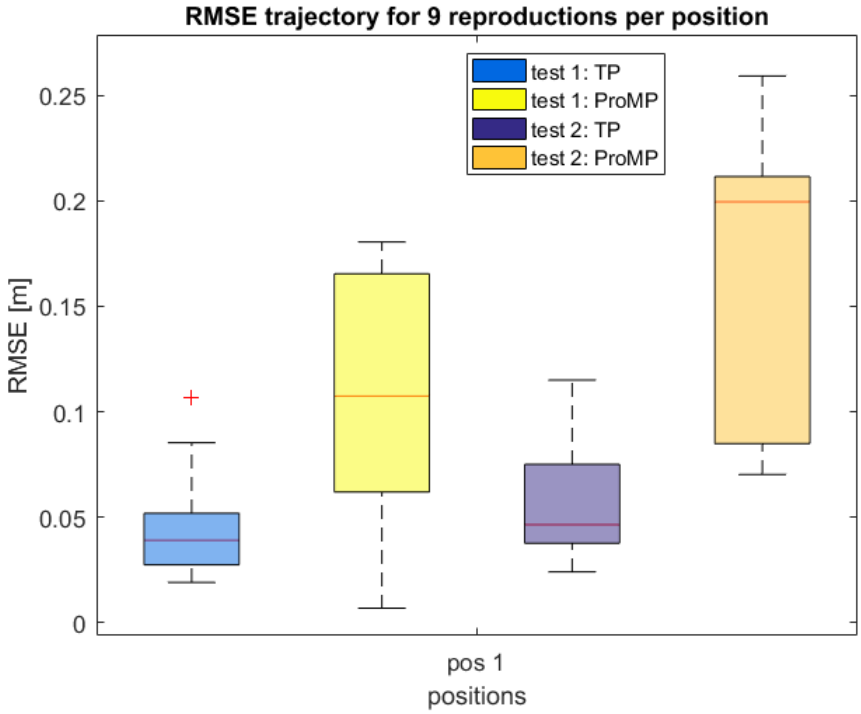


Figure 6.13: The root means squared error of the trajectory, for the Signature task. The training set consists of nine trajectories. For each trajectory a calculation was made, of the error of that trajectory, if it was given as input in the reproduction and all 9 trajectories or 8 trajectories (all the demonstrations minus the one used in the reproduction) were given in the learning phase.

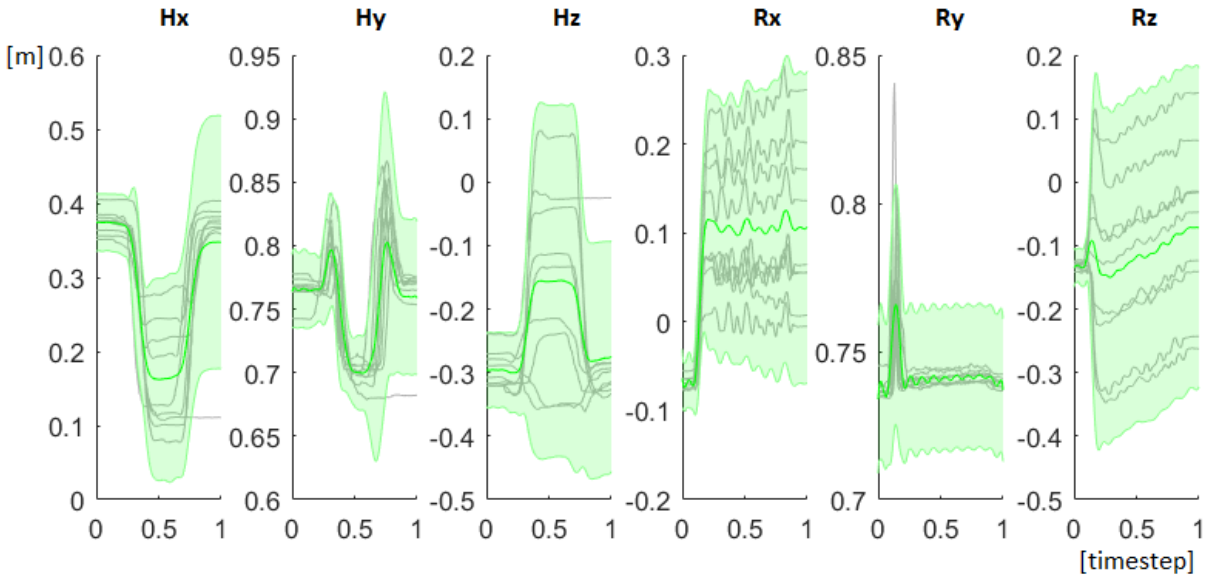


Figure 6.14: Model used by the ProMP approach for the Signature task, for position 1. The green area indicates the model and the grey lines show the demonstrations used to generate this model.

6.4.2. SMOOTHNESS OF THE TRAJECTORY

The figures regarding the smoothness of the trajectories can be found in Appendix B. Both approaches smoothen the trajectories more in comparison with the original data. The peaks in the change of acceleration rate, of the TP-inference approach is more similar to the original data. Smoothing the trajectory can be beneficial, but is could also result in the loss of information. For this task, because it is a complex movement, smoothing the trajectory can result in an inaccurate reproduction of the signature. This is visible for both approaches, especially for the ProMP approach.

6.4.3. CONCLUSION

The root mean squared error of the trajectory in Test 1, taken over all the position, has a mean and standard deviation of: $\mu = 0.0459$ m and $\sigma = 0.0300$ m and $\mu = 0.1086$ m and $\sigma = 0.0617$ m, for the TP-inference and the ProMP approach respectively. And for Test 2: $\mu = 0.0574$ m and $\sigma = 0.0310$ m and $\mu = 0.1680$ m and $\sigma = 0.0698$ m, for the TP-inference and the ProMP approach respectively. For this task the most important aspect was to perform the signature correctly. Figure 6.15 shows an example of the reproduction of the signature. The TP-inference has a better performance in reproducing the correct the position and reproducing a signature that comes close to the original shape.

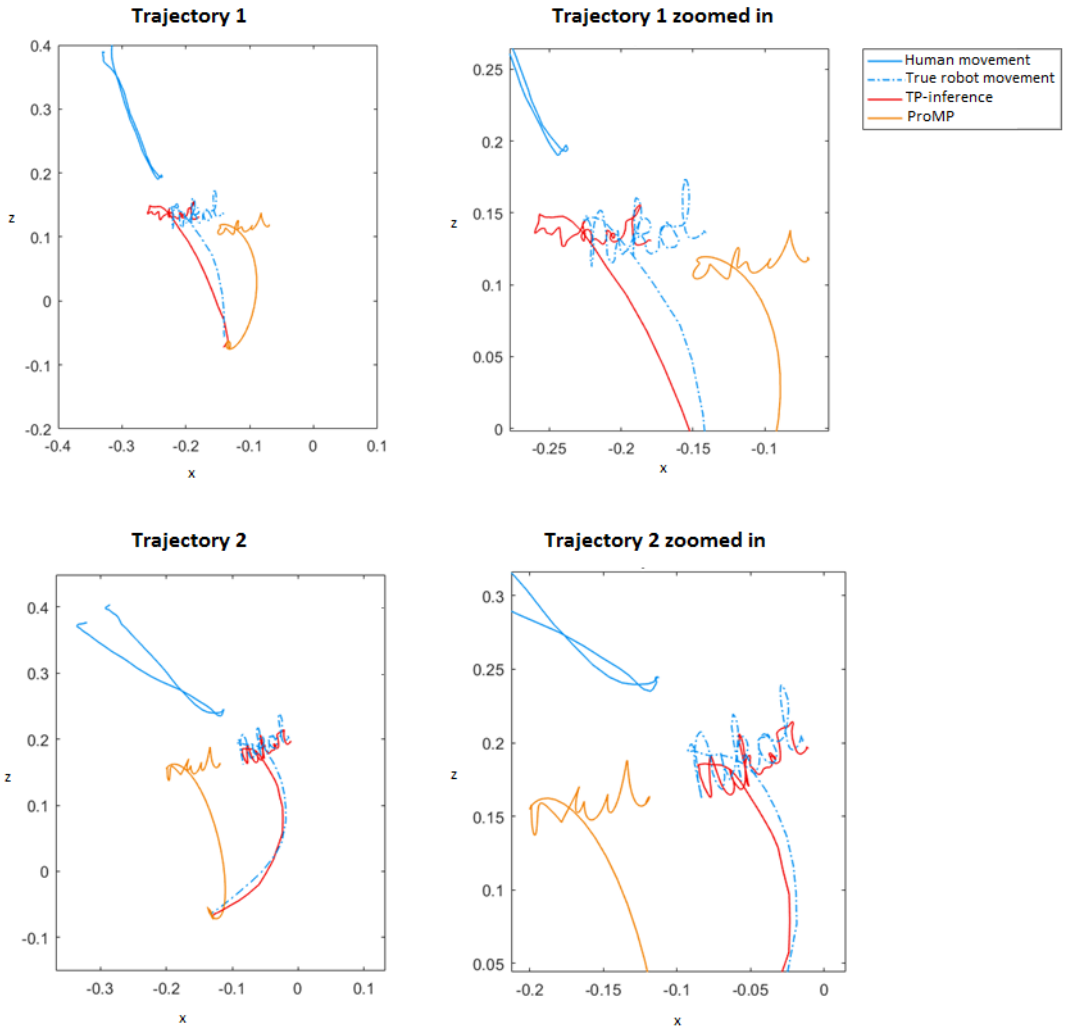


Figure 6.15: Two reproductions of the Signature task, visualizing the accuracy of the reproduction of the signature movement.

6.5. DISCUSSION

Regarding the four different tasks that were performed, in the Cuppile and the Tower task the ProMP approach performed better, compared to the TP-inference approach. In these tasks, the issue for the TP-inference approach was the high variety in length and shape of the trajectories of the nine given demonstrations.

During the recording of the demonstrations of the Cuppile task, the movement was performed with a different end location of the cup. Because the cup was light weighted, different overshoots in vertical direction occurred. The variety in shapes caused by the inconsistency of the trajectories, is not modeled correctly by the task-driven model, because the model calculates the mean and covariance of the training set. Implementing a GMM-GMR instead of a task-driven model could increase the performance. The GMM-GMR model has as a benefit that it iteratively positions the Gaussians over the demonstrations such that, within its capacity, it 'best' can describe the model. In a GMM-GMR the Gaussians are not fixed to represent specific data points—as is the case for the task-driven model—because the demonstration data is no longer required as a part of the model.

The issue of the inability of reproducing trajectories with different lengths, is because the TP-inference approach is not able to rescale or re-adjust the model. If the model is not rescaled, an overlab of the models of the two frames (for shorter movements) or a the gab between the models (for longer movements) causes distortion in the retrieved trajectory. To improve this in further research, the full Procrustes can be employed in order the scale the model. Another improvement could be to implement EM to optimize the inference of the task parameters by iteratively revalue the weight matrix, thereby re-adjusting the matching of the model to the new movement. This improvement is not suitable for big differences in length when using the task-driven model, because EM does not solve the distortion.

In the Tower task the loop-shaped movement of the human results in a covariance pattern, where at two parts of the movement the covariance is small, seen from the different perspectives. Because during the demonstration the orientation of the cups was altered, as well as the positions, it is difficult to correctly model the behaviour of the task by the task-driven model. For inferring the task parameters, the generalized model is matched using the two low variance parts in the movement. Because different orientations were used, the model is not matched correctly. Another type of model, such as the GMM-GMR could help to increase the performance. Another possibility is to investigate if the performance would increase by adding an extra local frame. Instead of one frame in the middle of the two cups, each cup could have its own frame. By keeping the task-driven model this would still result in incorrect orientations in the mapping, due to the different orientation of the demonstration samples.

For the Drink and the Signature task the TP-inference approach performs significantly better than the ProMP approach. In these tasks the demonstrations were more consistent and more complex than in the Cuppile and Tower task.

This consistency in the trajectories of the Drink task is caused by the pouring movement. The weight of the bottle, caused that the movements were performed with less overshoot in the vertical direction. The sharp curvy movement when pouring the drink also helped in the matching of the model of the second frame. For the ProMP approach the sharp curvy movement is lost in the high variance at this part of the movement, making it difficult to infer the position of the cup.

In the Signature task the movement towards the position of the signature differ, but the signature itself was in each demonstrations the same. For this task, the task-driven model is beneficial because it can describe the detailled curved shape of the signature and because the second frame is at the position of the signature. In this case the covariance at the signature part of the movement, is small. For the ProMP approach the curved shape of the signature is lost in the high variance at this part of the movement. In this Signature task different positions of signing were used in the training set causing a high variance at the position of the signature. An interesting aspect to investigate is the performance of the ProMP approach when the same position of signing is demonstrated multiple times.

7

CONCLUSION

In this thesis the use of task parameter inference in human robot interaction was presented. There were two main questions that were addressed:
RQ1. Can task parameter inference be applied in human-robot interaction?
RQ2. How does the task parameter inference approach compare with a common approach in this field?

To find an answer to the question if task parameter inference can be applied in human interactive tasks, a TP-inference approach—consisting of a combination of task parameter inference and task parameter movement retrieval—was created. A task-driven model was used to generalize the demonstration data and the task parameter inference was achieved by using the orthogonal Procrustes analysis.

The Procrustes analysis has the benefit that it can be used to retrieve rotation, translation, scale factor and can recognize mirrored data. In this research the orthogonal Procrustes was chosen as a starting point, because the algorithm keeps the perpendicular frame intact. A task-driven model was used to provide the Procrustes analysis with enough data points and shape to infer the task parameters.

To limit the scope of the research, the focus was on tasks where the movement of the human is fully observed before the robot starts performing its response. Static tasks were chosen as a starting point, to investigate the possibility of the task parameter inference in human-robot interaction.

Four static interactive tasks were constructed, with varying complexity—placing cups on top of each other (Cuppile), building a tower of three cups (Tower), pouring a drink (Drink) and placing a signature (Signature)—to investigate the boundaries of the approach. The Cuppile task describes a simple similar movement for both the human and the robot. This task is expanded to a non similar movement in the second task, Tower, where the important aspects of the movement were in the middle of the movement. The third task was chosen to illustrate a completely different task, where the complexity lies in observing the trajectory of the bottle rotation and inferring the place of cup. In the last task, Signature, the complexity of the movement of the robot was increased instead of the movement of the human.

Because the TP-inference approach was based on a Cartesian coordinate system, the tasks were created such that there would be no need to compensate for the changes in hand orientation during

the movement.

In the search for an answer how the task parameter inference compare to a common approach in this field, first the most commonly used approaches have been investigated. Movement primitives is a well known and frequently used approach in PbD. The research of Amor et al. [6] had similar goals as described in this thesis, but because this approach is limited to tasks with linear correlation between the movement of the human and the robot, the Probabilistic movement primitives (ProMP) approach of the following work [1] was used to compare with the TP-inference approach. The results of the performance of the TP-inference approach and the ProMP approach in the four static interactive tasks were compared.

Two tests were performed for each of the four interactive tasks. The first test was to give 9 demonstrations and to retrieve one of those trajectories in the reproduction phase. The second test was to give 8 demonstrations and to retrieve a new trajectory in the reproduction phase. For each test the error at the position of the object, the root mean squared error of the trajectory and the smoothness of the trajectory, was calculated. For the first three tasks the tests were repeated for five positions. For each position 9 reproductions were performed. For the last task the tests were performed for one position, due to the complexity of the movement of the robot. The arm's reach was not long enough to perform the task from the different positions. For this position 9 reproductions were performed.

This comparison resulted in the following conclusions. For tasks with simple straightforward movements the TP-inference approach has no significant benefit in comparison with the ProMP approach. However, for more complex tasks, where the movement is more complex and curvy, the TP-inference approach performs with much higher accuracy and more precise compared to the ProMP approach.
For the Drink task on an table of 1.4 by 0.7 m the TP-inference approach has a average error of 0.0155 m with a standard deviation of 0.0061 m and for the ProMP approach an average error of 0.1649 m with a standard deviation of 0.0661 m, when inferring the location of the cup.
For the Signature task the root mean squared error of the trajectory, the TP-inference approach has and average of 0.0516 m with a standard deviation of 0.0302 m and the the ProMP approach an average of 0.1383 m with a standard deviation of 0.0708 m.

In the simple tasks was observed that the TP-inference approach had a lower performance when the given demonstrations differ in trajectory shape and length. These differences were not represented correctly by the task-driven model.

When in the reproduction phase a shorter trajectory compared to the demonstrations is given, the TP-inference approach is not able to rescale the model, resulting in an overlap of the models of the local frames causes distortion of the retrieved trajectory.

7.1. FUTURE WORK

A method to improve the inference accuracy of the TP-inference approach could be to implement the full Procrustes analysis, in order to scale the model, and to employ Expectation Maximization (EM) to optimize the task parameters by re-adjusting the weights.

The model of the TP-inference approach is much bigger than the model of the ProMP approach, which makes it slower in computation. A next step could be to use a Gaussian Mixture Model-

Gaussian Mixture Regression (GMM-GMR) based model instead of a task-driven model as used in these experiments, in order to decrease the size of the model.

Another part that could be interesting for further research, is the application of the TP-inference approach in non-static tasks, also referred to as online movement retrieval. The ProMP approach is able to predict the intention of the human during the observation, making it possible to start the movement of the robot while the human is still performing his own part of the task. The ProMP approach uses a combination of Dynamic Time Warping (DTW) and a predictive distribution to estimate and to update the prediction during the observation. The first step for the TP-inference approach to achieve this online retrieval, could be to implement the EM mentioned above.

A

FULL EXPERIMENTAL DESIGN

A.1. EXPERIMENTAL SETUP

The focus of the comparison between the TP-inference approach and the ProMP approach, was on tasks where the movement of the human is fully observed, before the robot starts performing its response. Doing so, 4 situations were constructed: placing cups on top of each other (Cuppile), building a tower of 3 cups (Tower), pouring a drink (Drink), placing a signature (Signature). The four tasks were recorded using a motion capturing system (OptiTrack) in the same setup.

A.1.1. SETUP

The interactive tasks were static (first the movement of the human, then the movement of the robot) and therefore the demonstrations could be performed by one person.
Used equipment:

- One person
- One table (of 1.4 by 0.7 m), two chairs and three cups and one bottle.
- OptiTrack system,computer with OptiTrack software
- Two OptiTrack markers for the demonstrator's active hand: one for the human movement and one for the robot movement.
- Three OptiTrack frame markers, that indicate the frames of reference of the human and robot position, and the position and orientation of the cup.
- Tape and a pen to mark the positions and orientations of the local frames on the table (this was necessary to reproduce the demonstrations).

Figure A.1 shows images of the setup. The left image illustrates how the tape is used to indicate both the location and the orientation of the second local frame. Pretests where performed to determine the shape of the local frame markers, to minimize the interference of the different markers during the recording of the movement.

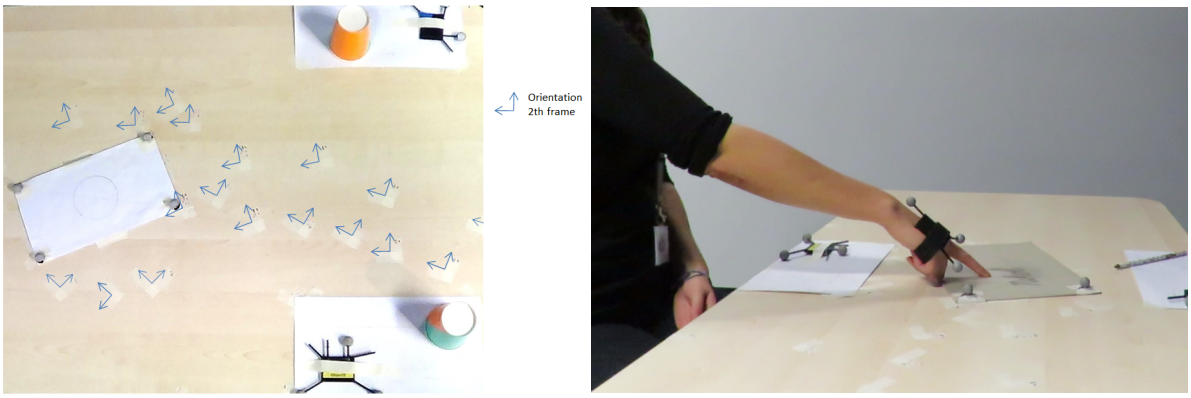


Figure A.1: Images of the setup during the demonstrations. Three frames are recorded for the interactive task. One for the human position, one for the robot position and one for the position of the object. During the TP-inference approach and the task parameter movement retrieval 2 frames are used: for TP-inference approach, the local frame of the human and the object, for the task parameter movement retrieval the local frame of the robot and the object.

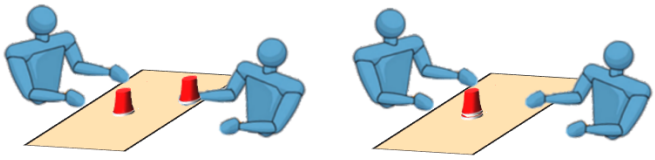
The global frame of reference is set at a fixed position at the beginning of the calibration. In this case the axis of the global frame was positioned to right of the person (seen in figure A.1), at half width of the table, on the floor.

To reproduce the same movement more than ones, a fixed human and robot local frame is used. This is achieved by using a fixed marker position in front of the demonstrator on the table.

To overcome overshoots in the movement due to wrist flexion, the center of the hand marker is translated to the center of the cup (all tasks), to the pouring end of the bottle (Drink task) and to the position of the thumb and index finger (Signature task).

SETUP 1: CUPPILE

In this task the human places a cup on the table and the robot places a cup on top of it. This is the first task and it illustrates an almost similar movement of the human and the robot.



The task is created to test the behavior of the approaches when altering the amount of demonstrations and positions of the movement.

Demonstrations:

The size of the table gave the possibility to perform the demonstration from different positions. The demonstration were given from 5 positions, with for each position 9 trajectories, resulting in a data set of 45 demonstrations. One test set was created where the location and movement of the human is slightly altered (5 more movements), see Figure A.2. Figure A.2 shows in the left image the setup and in the right image all the demonstration movements.

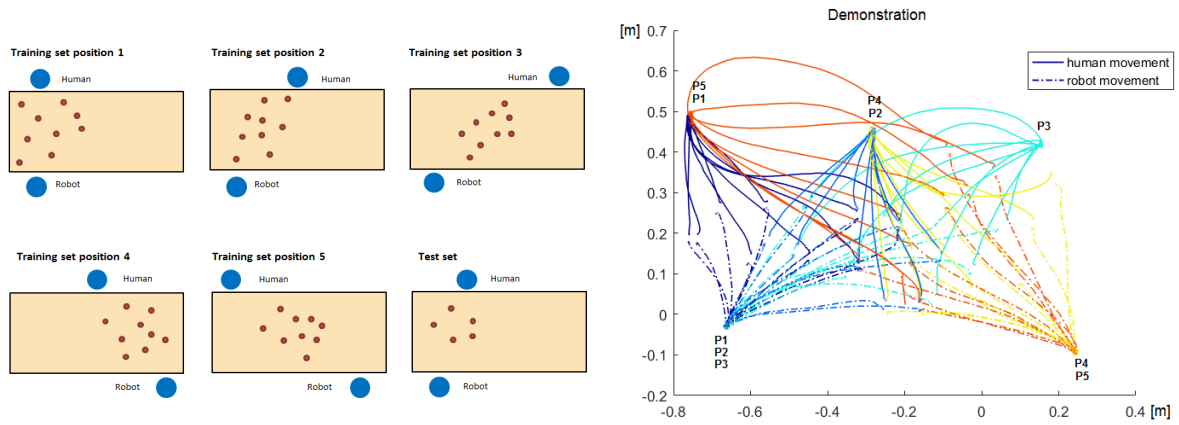
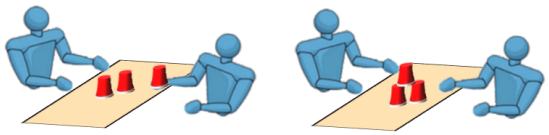


Figure A.2: Setup and demonstration data for the cuppile task, seen from above. The orange square represents the table. The blue circles represent the position om the human and robot, and the red the position of the cups.

SETUP 2: TOWER

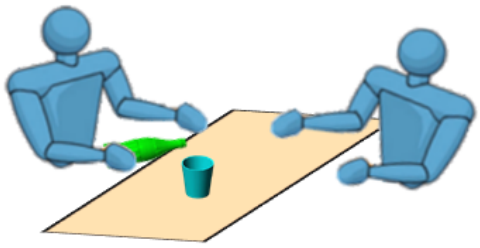


In this task the human places two cups next to each other on the table and the robot has to place another cup on top of these two cups. This can be seen as an extension of the previous task, with the difference that the movement is recorded until the hand is back to neutral position, therefore placing the important task parameters in the middle of the movement.

Demonstrations:

For this task, the same setup is used as described in setup 1, with the same position and orientations of the local frames and the same amount of demonstrations, see the left image of Figure A.2.

SETUP 3: DRINK

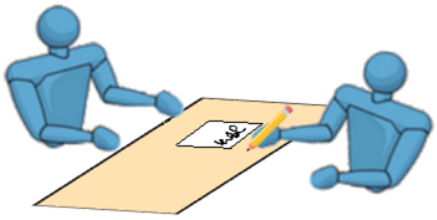


For this task, the human pours the drink and the robot picks up the glass afterwards. This task has a more complex movement due to the curves of the pouring movement.

Demonstrations:

For this task, the same setup is used as described in setup 1, with the same position and orientations of the local frames and the same amount of demonstrations, see the left image of Figure A.2.

SETUP 4: SIGNATURE



For this task the human point to a location on the table and the robot has to write down a name. This is the first task where the complexity of the movement of the robot is enhanced.

Demonstrations:

Because of the limitation of the length of the arm, it was not possible to perform the demonstrations in the same manner as in the previous tasks. Therefore is chosen to perform the task from one position for the human and robot with 9 different demonstrations. This is reproduced 4 times, see figure A.3.

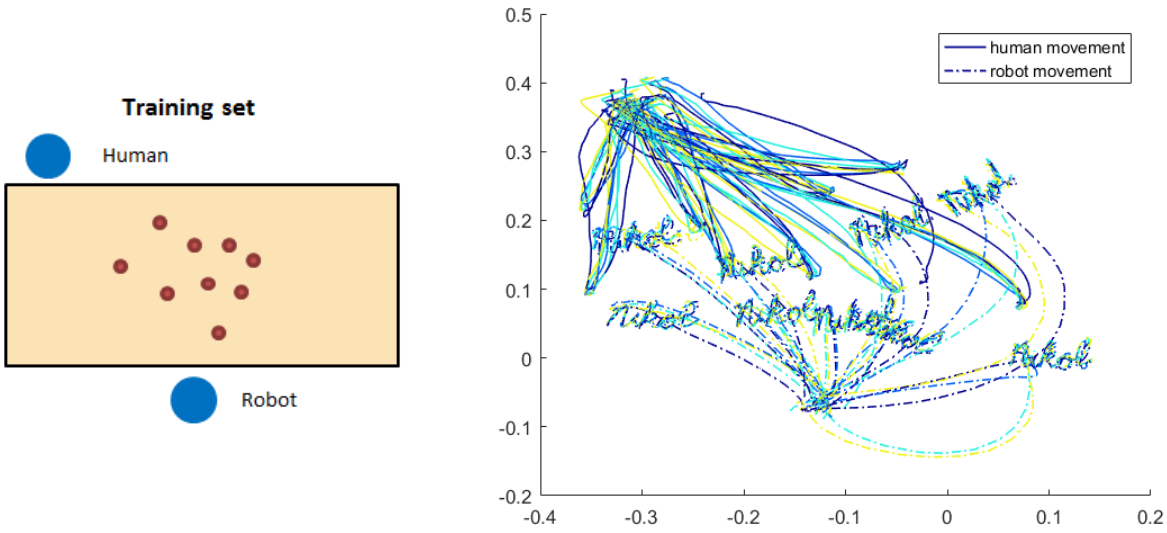


Figure A.3: Setup and demonstration data for the signature task, seen from above. The orange square represents the table. The blue circles represent the position om the human and robot, and the red the position of the cups.

B

FULL EXPERIMENTAL RESULTS

B.1. CUPPILE RESULTS

B.1.1. TEST 1

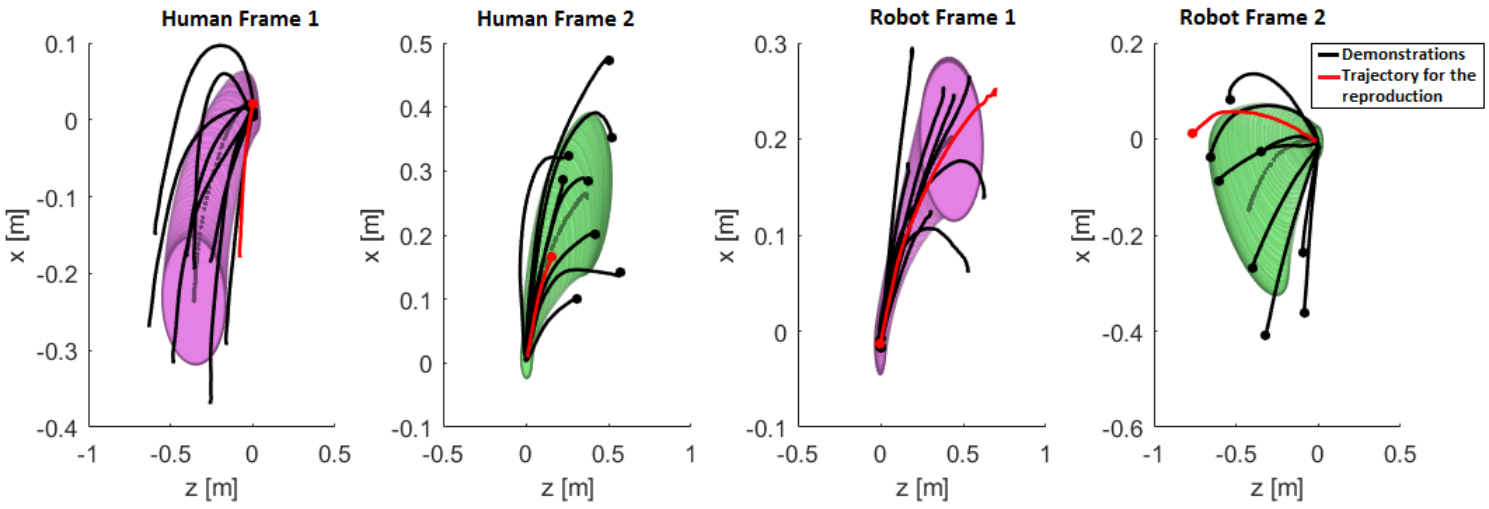


Figure B.1: An illustration of the task-driven model for the Cuppile task for Test 1 at position 3. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory is indicated which gave the highest error, see Figure 6.2 position 3. For the movement of the human this red trajectory is much smaller than the other demonstrations.

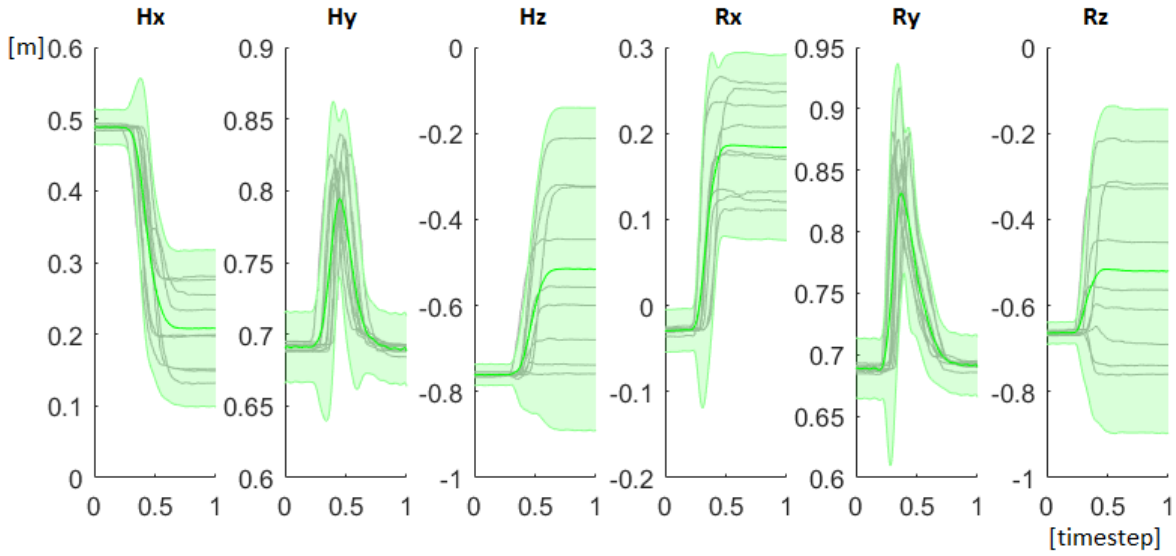


Figure B.2: Model used by the ProMP approach for the Cuppile task, for position 1. The green area indicates the model and the grey lines show the demonstrations used to generate this model.

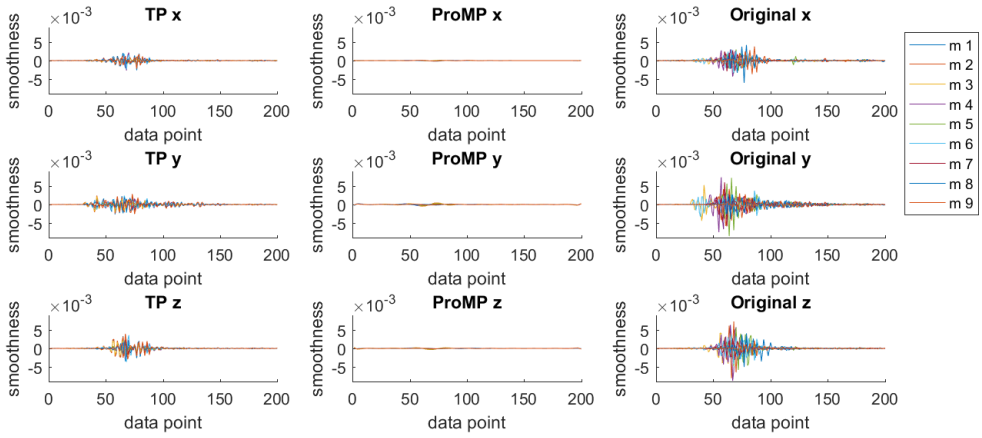


Figure B.3: The smoothness per trajectory (m) in Test 1 for the Cuppile task at position 2. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

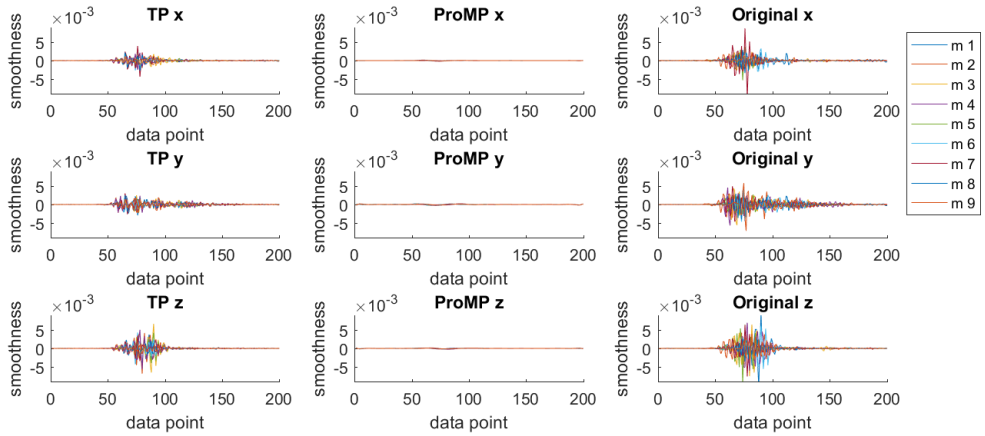


Figure B.4: The smoothness per trajectory (m) in Test 1 for the Cuppile task at position 3. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

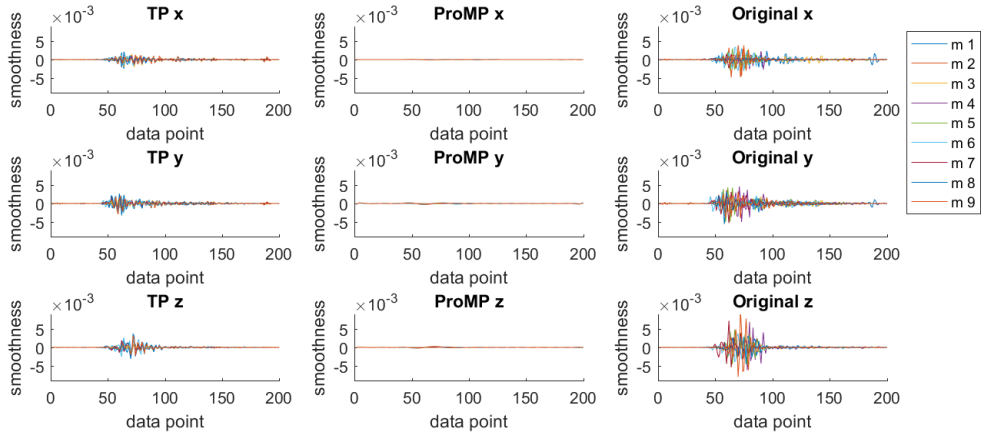


Figure B.5: The smoothness per trajectory (m) in Test 1 for the Cuppile task at position 4. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

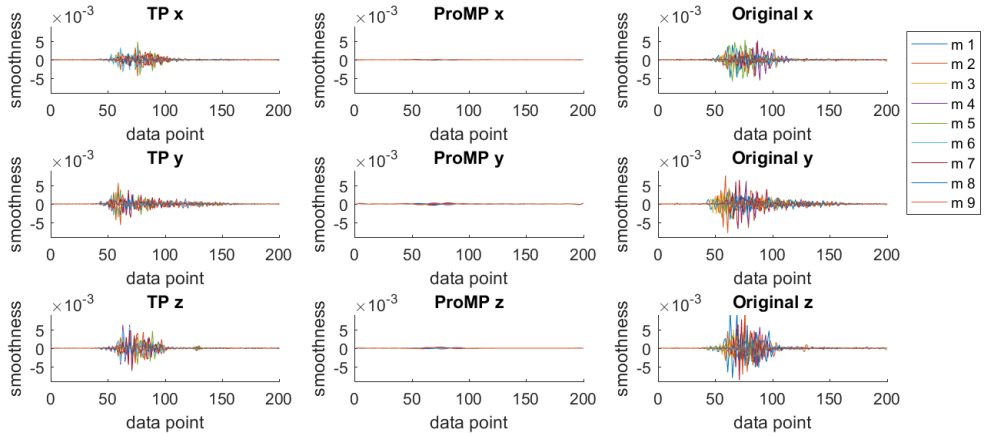


Figure B.6: The smoothness per trajectory (m) in Test 1 for the Cuppile task at position 5. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

B.1.2. TEST 2

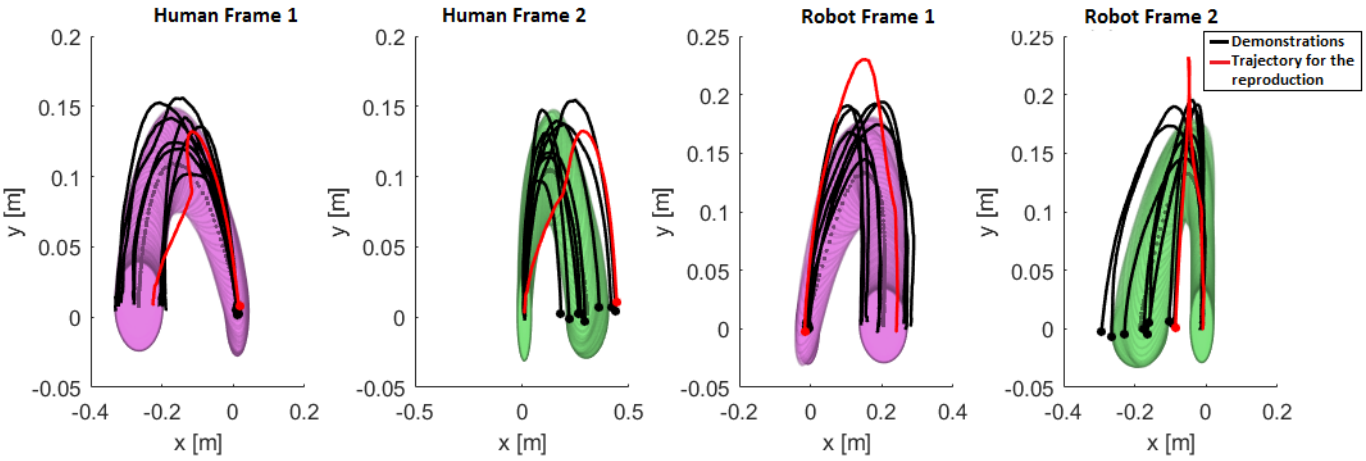


Figure B.7: An illustration of the task-driven model for the Cuppile task for Test 2 at position 1. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory that is given in the reproduction phase.

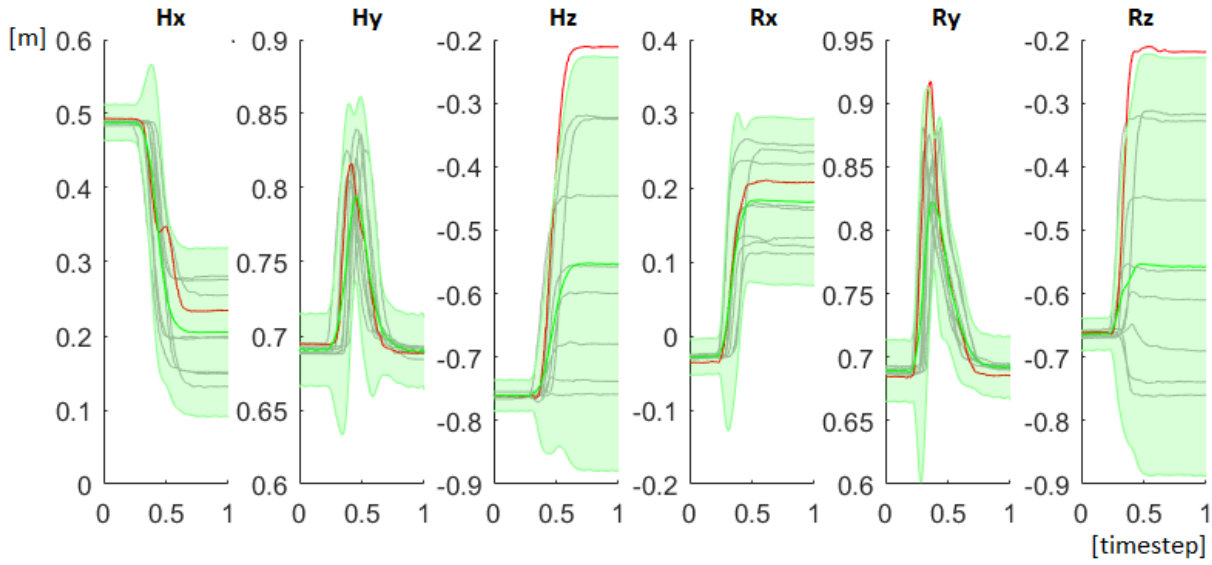


Figure B.8: An illustration of the model of the ProMP approach, for the Cuppile task for Test 2 at position 1. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory that is given in the reproduction phase. The red line in this figure is the same trajectory as the red line in Figure B.7.

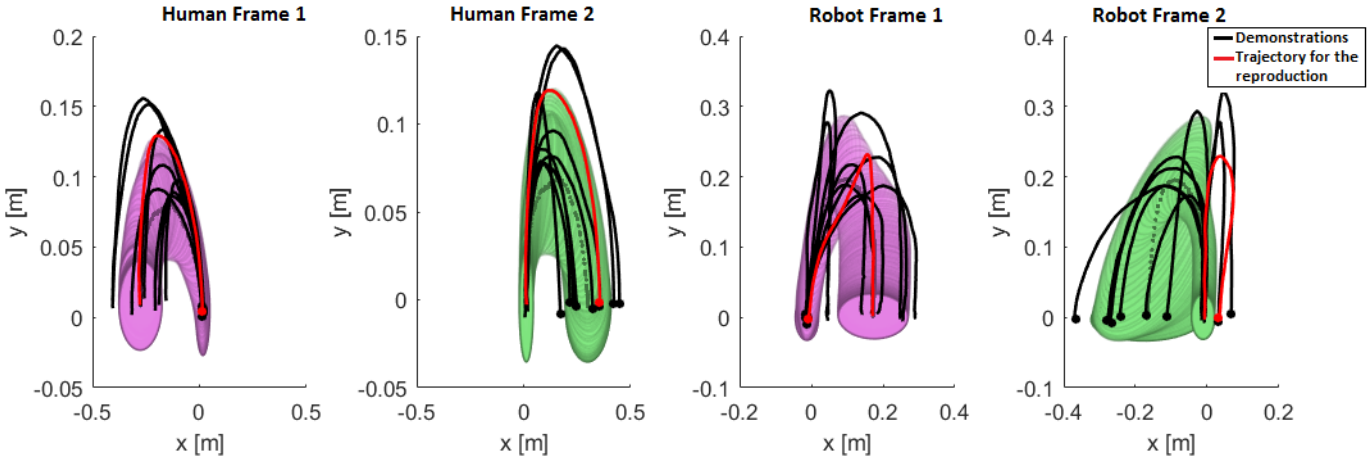


Figure B.9: An illustration of the task-driven model for the Cuppile task for Test 2 at position 2. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory that is given in the reproduction phase.

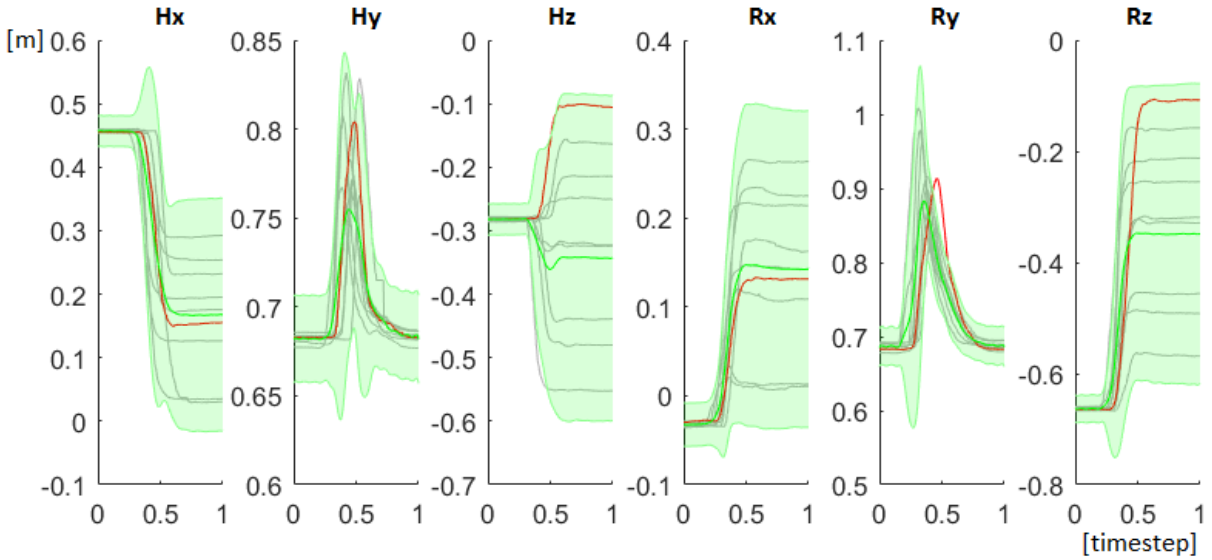


Figure B.10: An illustration of the model of the ProMP approach, for the Cuppile task for Test 2 at position 2. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory that is given in the reproduction phase. The red line in this figure is the same trajectory as the red line in Figure B.9.

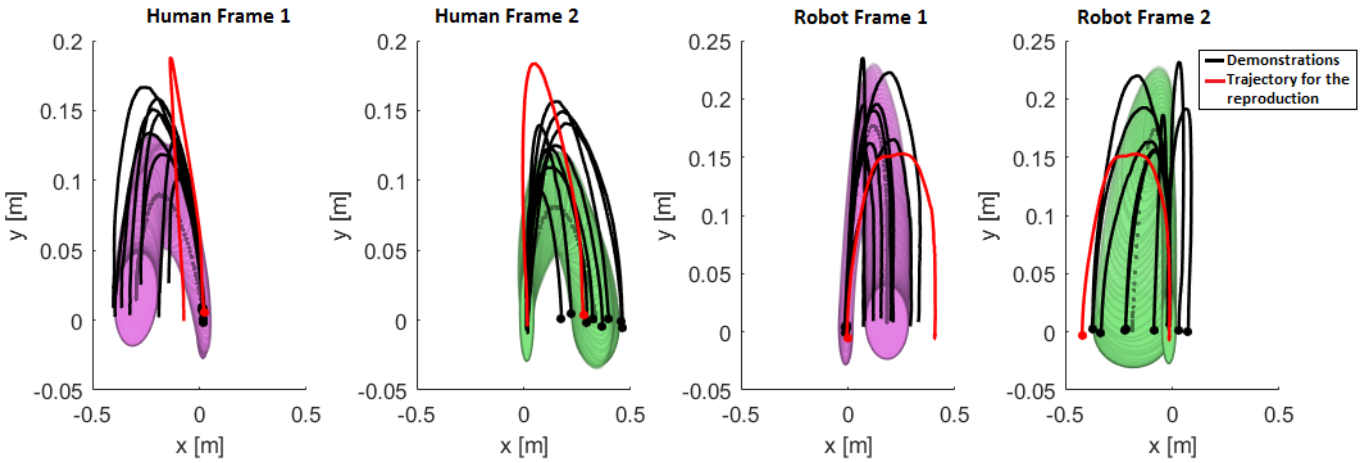


Figure B.11: An illustration of the task-driven model for the Cuppile task for Test 2 at position 4. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory that is given in the reproduction phase.

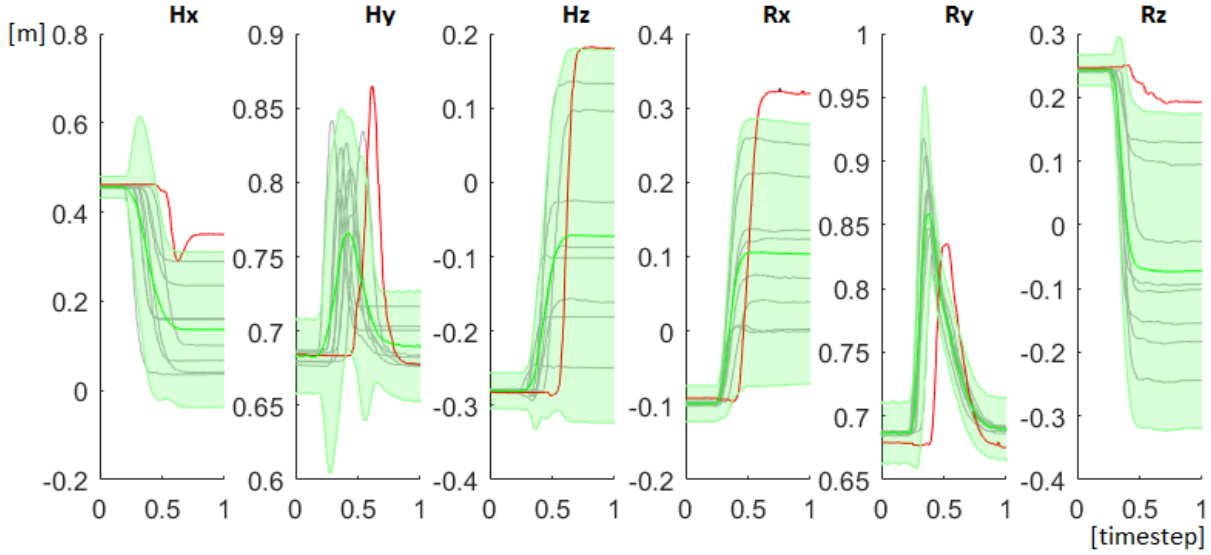


Figure B.12: An illustration of the model of the ProMP approach, for the Cuppile task for Test 2 at position 4. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory that is given in the reproduction phase. The red line in this figure is the same trajectory as the red line in Figure B.11.

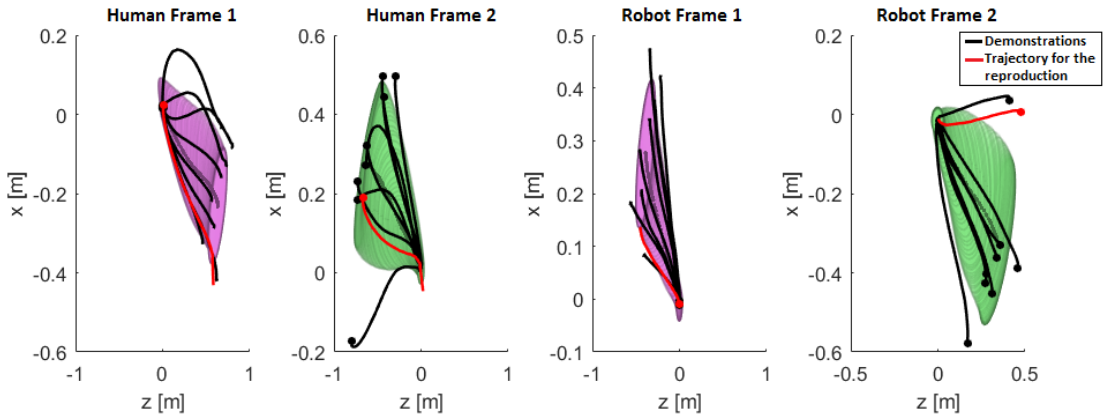


Figure B.13: An illustration of the model of the ProMP approach, for the Cuppile task for Test 2 at position 5. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory that is given in the reproduction phase.

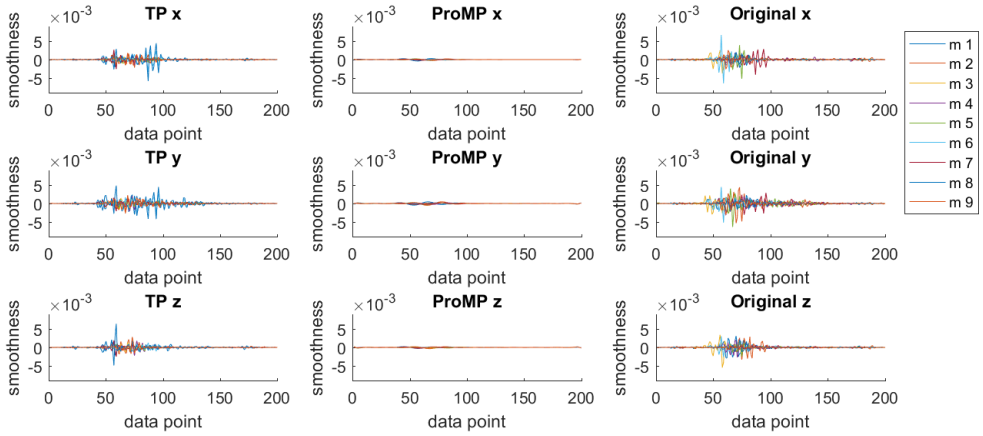


Figure B.14: The smoothness per trajectory (m) in Test 2 for the Cuppile task at position 1. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

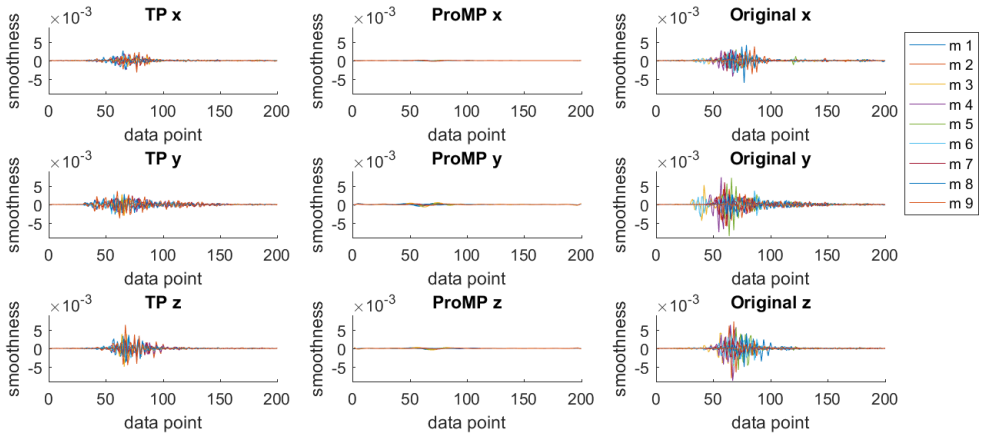


Figure B.15: The smoothness per trajectory (m) in Test 2 for the Cuppile task at position 2. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

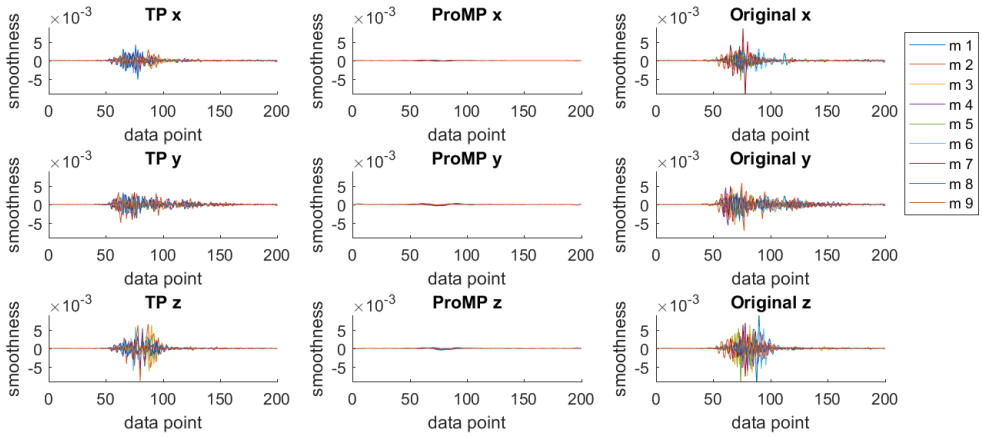


Figure B.16: The smoothness per trajectory (m) in Test 2 for the Cuppile task at position 3. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

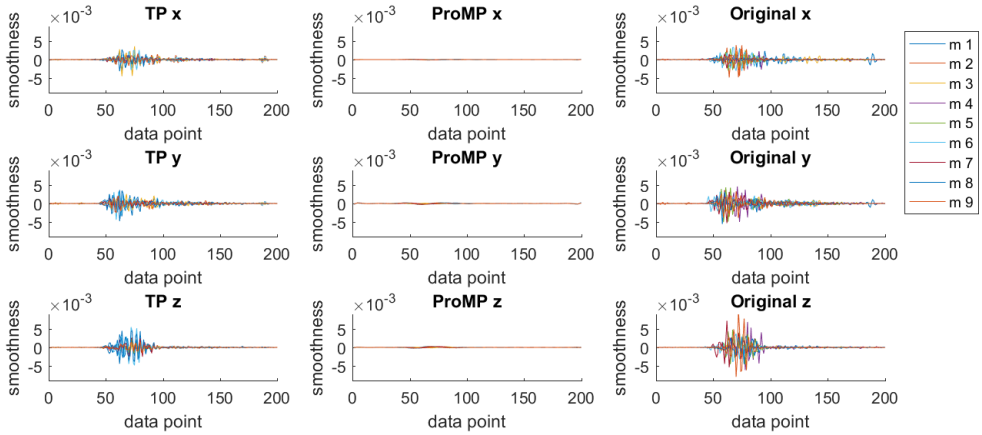


Figure B.17: The smoothness per trajectory (m) in Test 2 for the Cuppile task at position 4. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

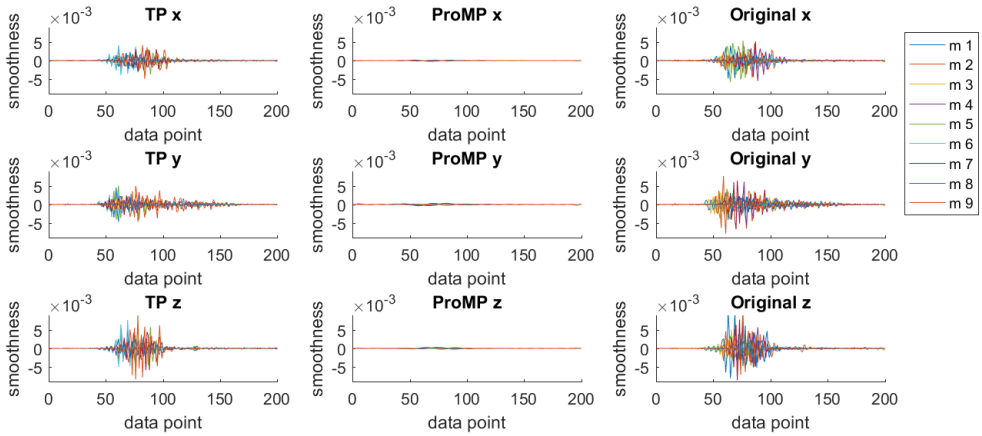


Figure B.18: The smoothness per trajectory (m) in Test 2 for the Cuppile task at position 5. One part of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup.

B.2. TOWER RESULTS

B.2.1. TEST 1

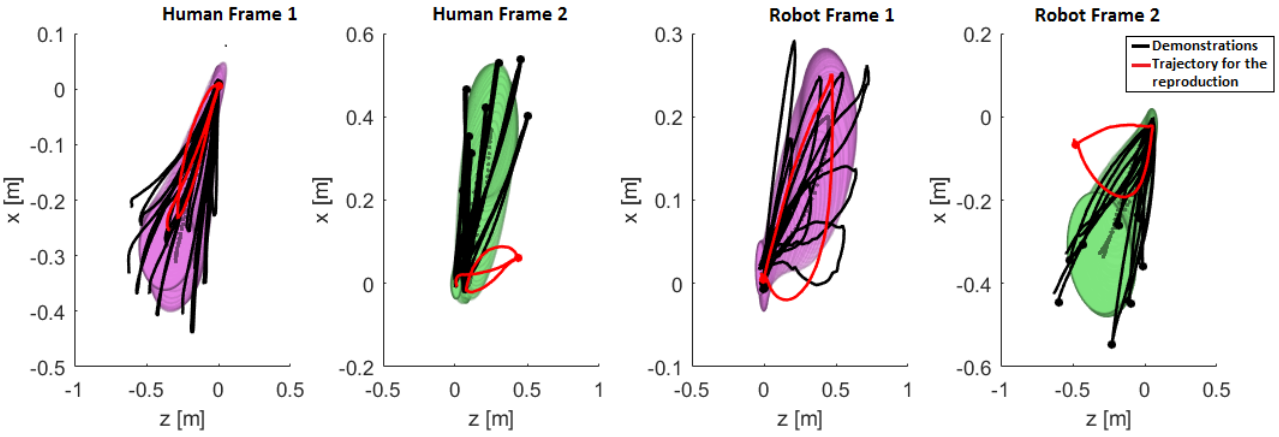


Figure B.19: An illustration of the task-driven model for the Tower task for Test 1 at position 3. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory is indicated which gave the highest error, see Figure 6.8 position 3. In the second frame for both the human movement and the movement the robot is visible that the red movement deviates from the rest of the trajectories.

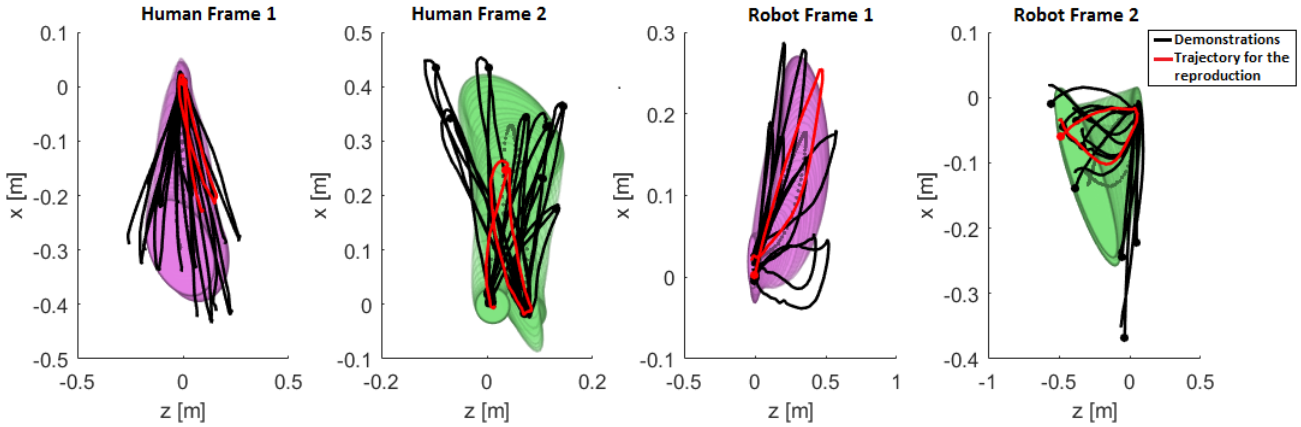


Figure B.20: An illustration of the task-driven model for the Tower task for Test 1 at position 2. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory is indicated which gave the highest error, see Figure 6.8 position 2. For the movement of the human this red trajectory is much smaller then for the other demonstrations.

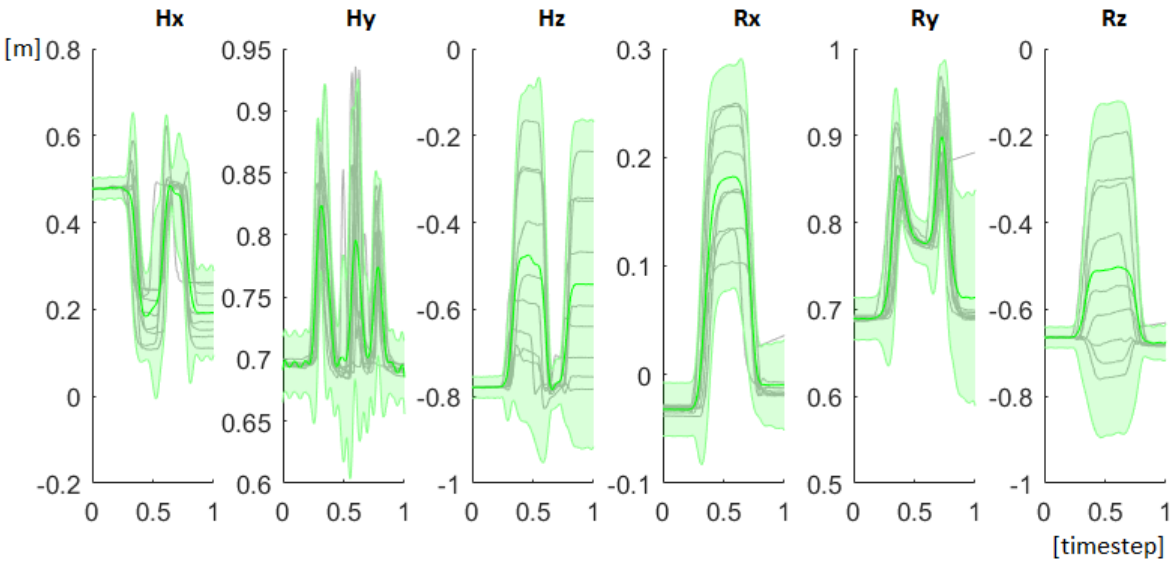


Figure B.21: Model used by the ProMP approach for the Tower task, for position 1. The green area indicates the model and the grey lines show the demonstrations used to generate this model.

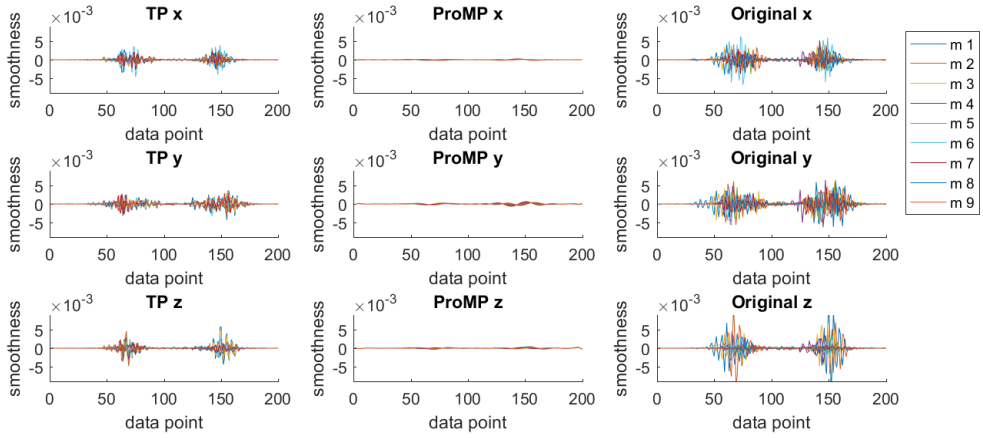


Figure B.22: The smoothness per trajectory (m) in Test 1 for the Tower task at position 1. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup and lifting the hand to go back.

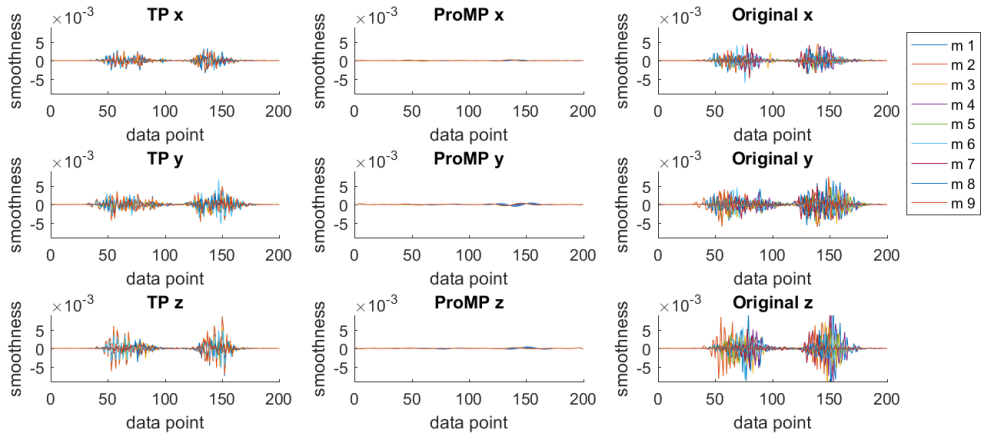


Figure B.23: The smoothness per trajectory (m) in Test 1 for the Tower task at position 2. Two parts of the movement creates a lot of the jerk. This is the part of lifting up the cup and lifting the hand to go back.

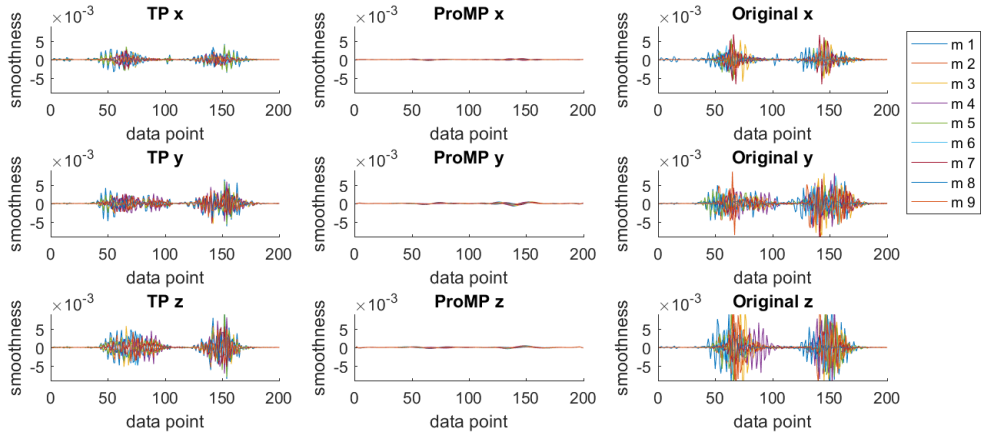


Figure B.24: The smoothness per trajectory (m) in Test 1 for the Tower task at position 3. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup and lifting the hand to go back.

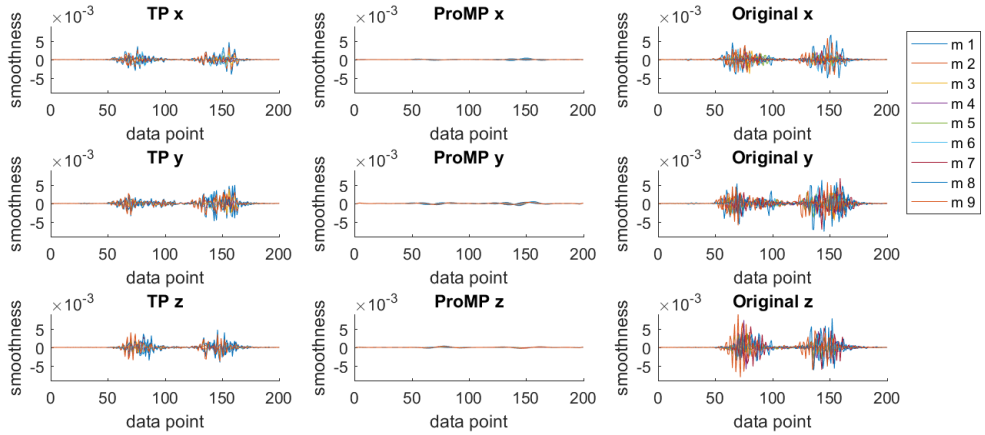


Figure B.25: The smoothness per trajectory (m) in Test 1 for the Tower task at position 4. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup and lifting the hand to go back.

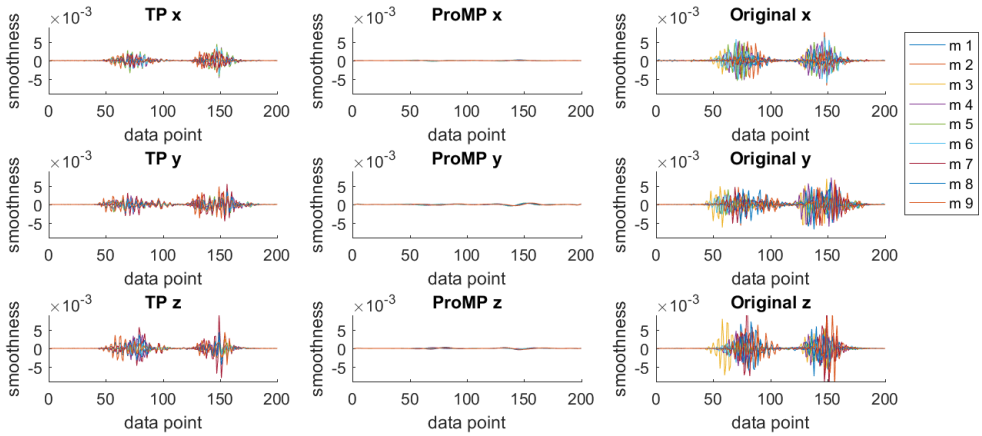


Figure B.26: The smoothness per trajectory (m) in Test 1 for the Tower task at position 5. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup and lifting the hand to go back.

B.2.2. TEST 2

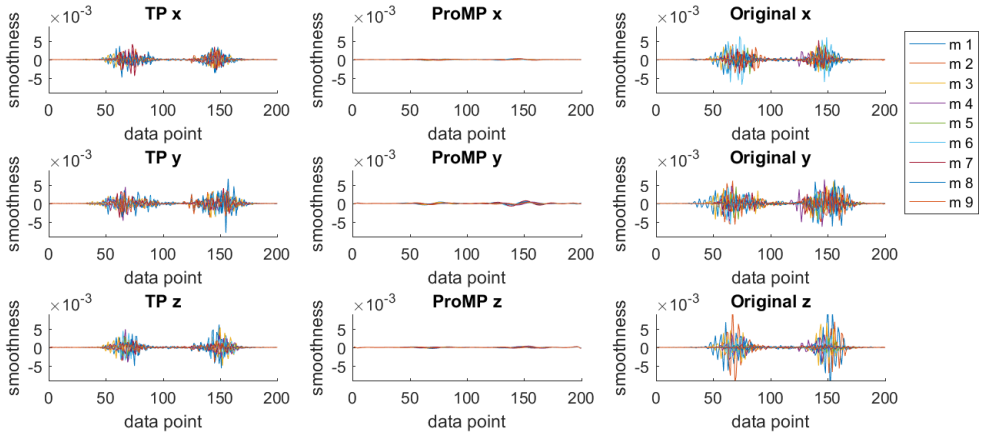


Figure B.27: The smoothness per trajectory (m) in Test 2 for the Tower task at position 1. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup and lifting the hand to go back.

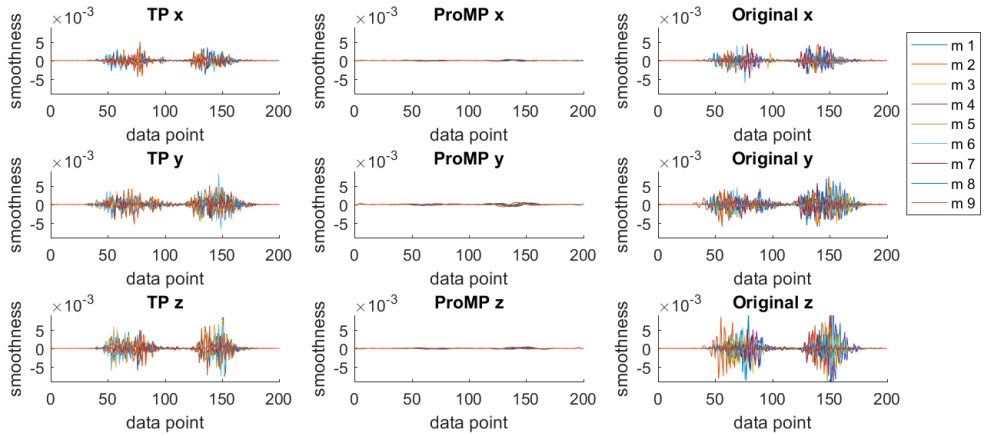


Figure B.28: The smoothness per trajectory (m) in Test 2 for the Tower task at position 2. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup and lifting the hand to go back.

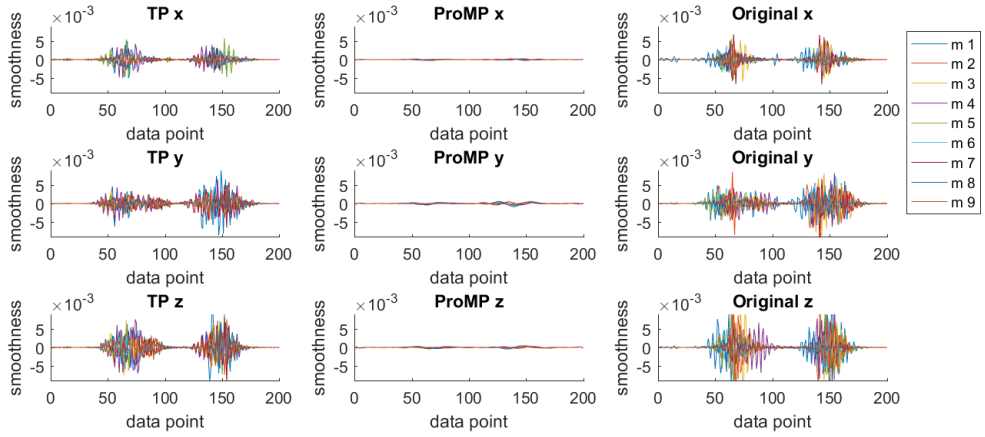


Figure B.29: The smoothness per trajectory (m) in Test 2 for the Tower task at position 3. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup and lifting the hand to go back.

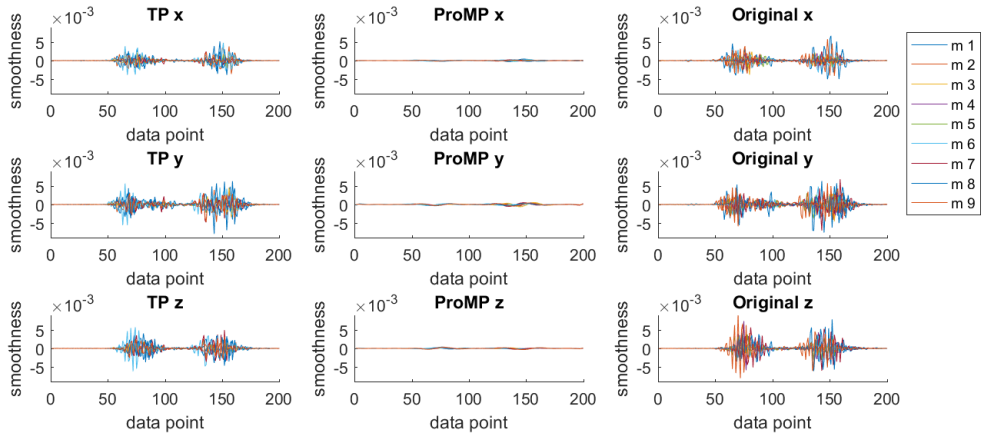


Figure B.30: The smoothness per trajectory (m) in Test 2 for the Tower task at position 4. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup and lifting the hand to go back.

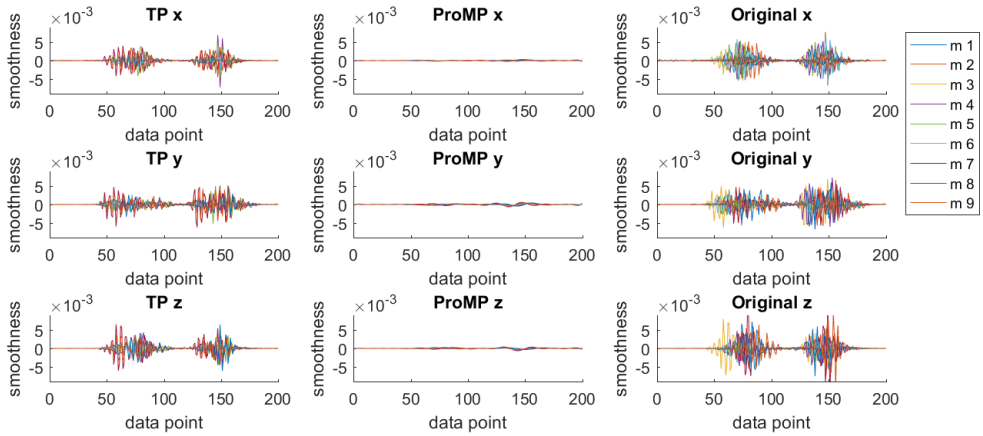


Figure B.31: The smoothness per trajectory (m) in Test 2 for the Tower task at position 5. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the cup and lifting the hand to go back.

B.3. BOTTLE RESULTS

B.3.1. TEST 1

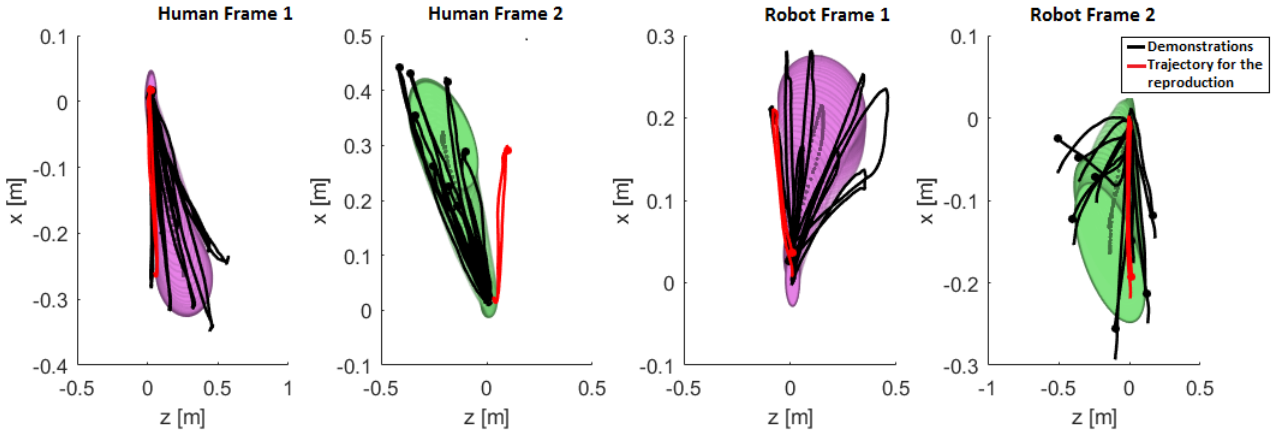


Figure B.32: An illustration of the task-driven model for the Bottle task for Test 1 at position 1. The first two graphs show the model and demonstrations of the human movements and the third and fourth graph show the model and movements of the robot. The black lines are the demonstrations and in red the trajectory is indicated which gave the highest error, see Figure 6.10 position 1. In the second frame for the human movement is visible that the red movement deviates from the rest of the trajectories.

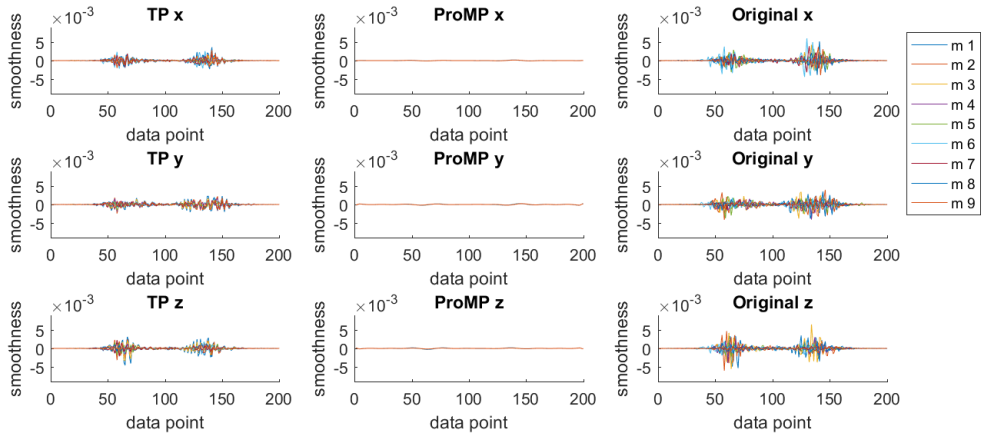


Figure B.33: The smoothness per trajectory (m) in Test 1 for the Drink task at position 1. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

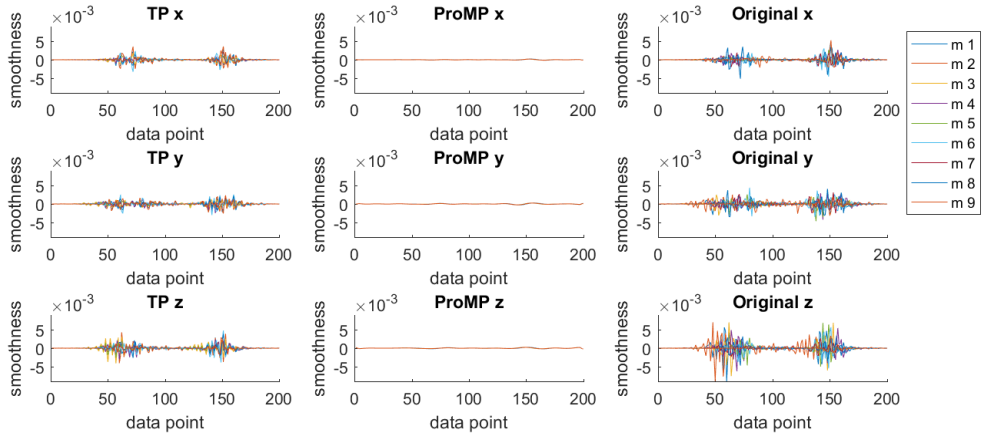


Figure B.34: The smoothness per trajectory (m) in Test 1 for the Drink task at position 2. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

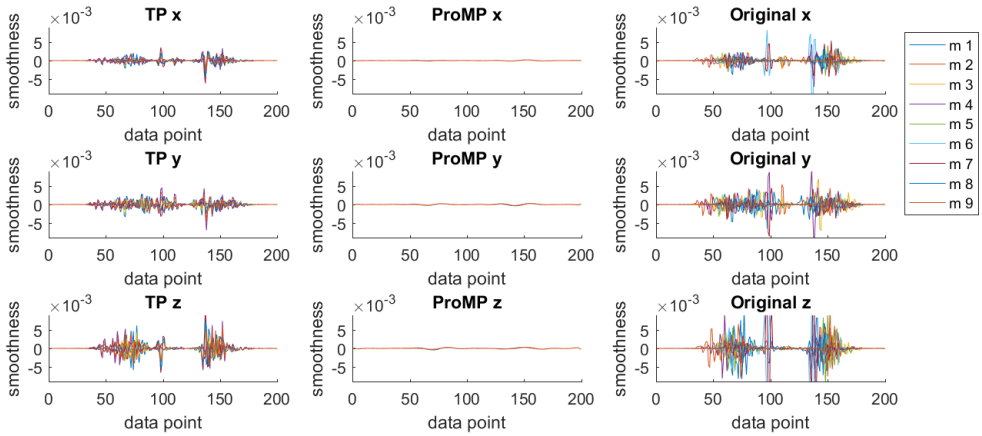


Figure B.35: The smoothness per trajectory (m) in Test 1 for the Drink task at position 3. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

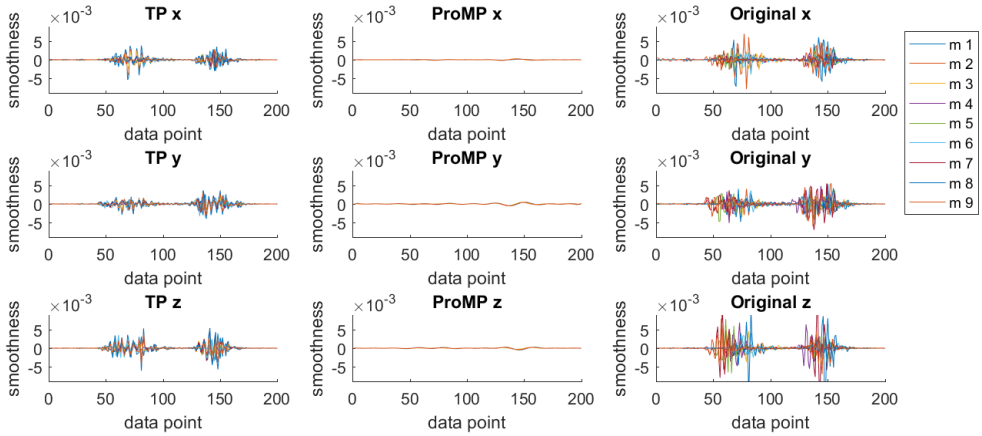


Figure B.36: The smoothness per trajectory (m) in Test 1 for the Drink task at position 4. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

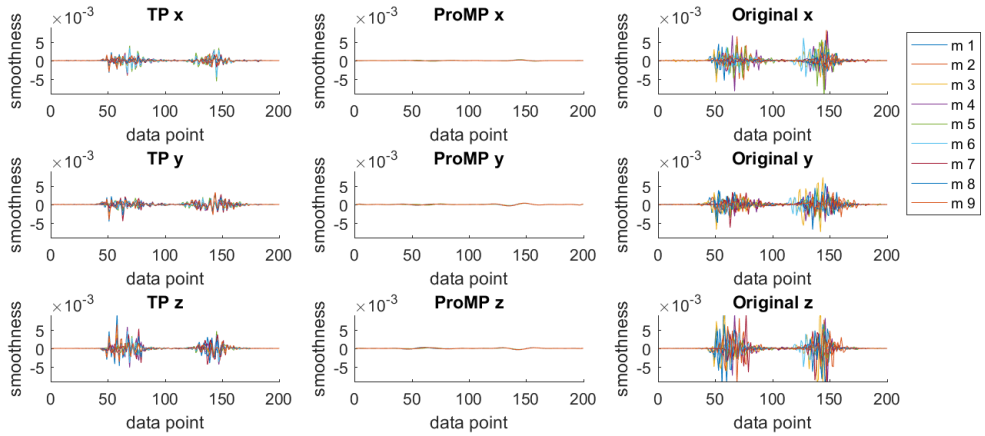


Figure B.37: The smoothness per trajectory (m) in Test 1 for the Drink task at position 5. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

B.3.2. TEST 2

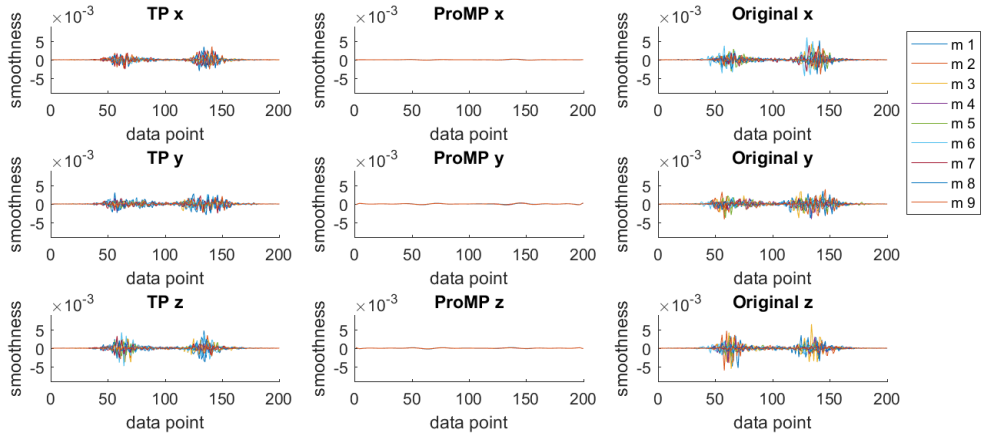


Figure B.38: The smoothness per trajectory (m) in Test 2 for the Drink task at position 1. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

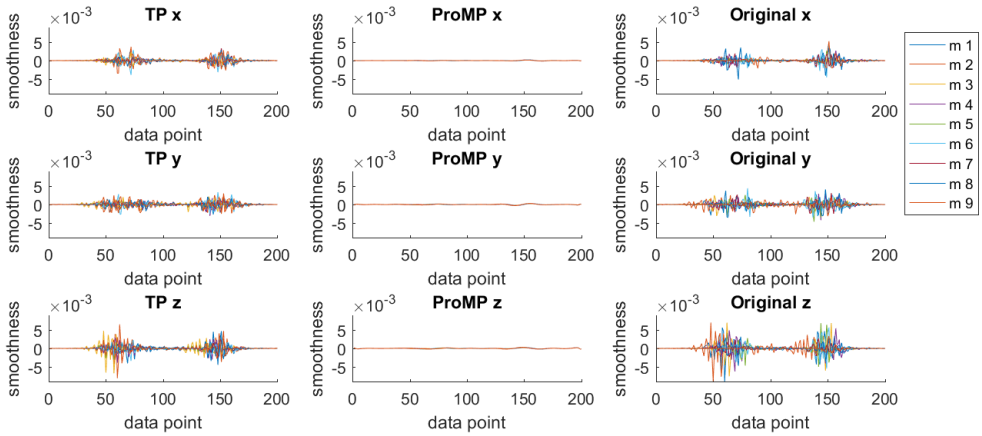


Figure B.39: The smoothness per trajectory (m) in Test 2 for the Drink task at position 2. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

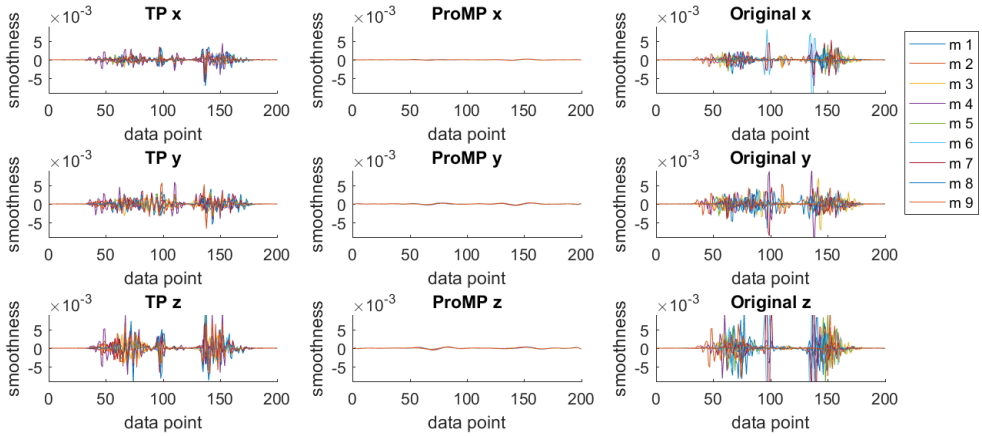


Figure B.40: The smoothness per trajectory (m) in Test 2 for the Drink task at position 3. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

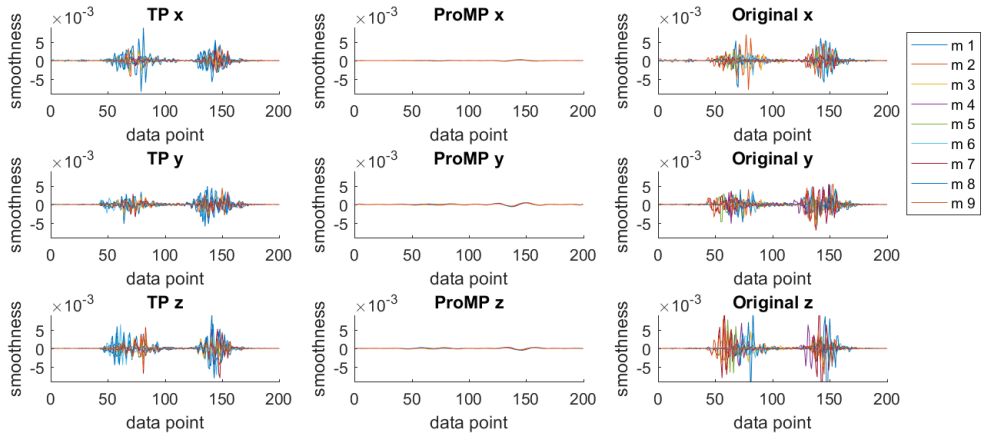


Figure B.41: The smoothness per trajectory (m) in Test 2 for the Drink task at position 4. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

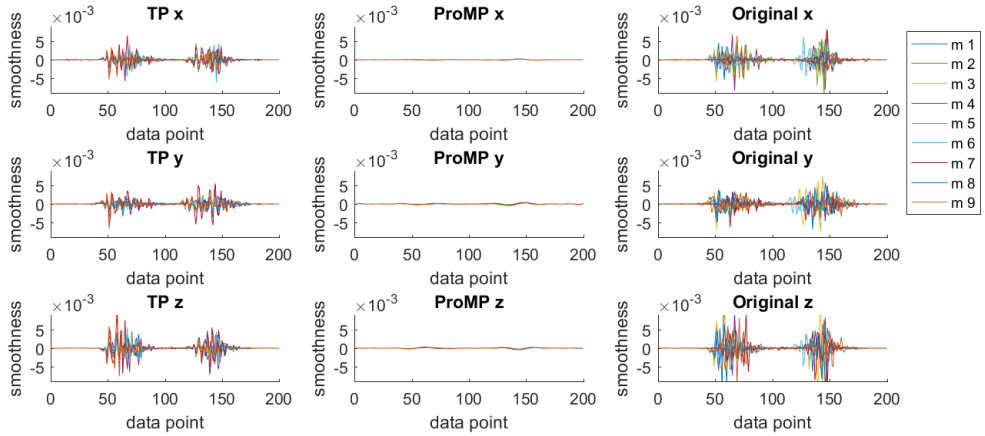


Figure B.42: The smoothness per trajectory (m) in Test 2 for the Drink task at position 5. Two parts of the movement creates high peaks in the change of acceleration rate. This is the part of lifting up the hand towards the cup and lifting up the cup.

B.4. SIGNATURE RESULTS

B.4.1. TEST 1

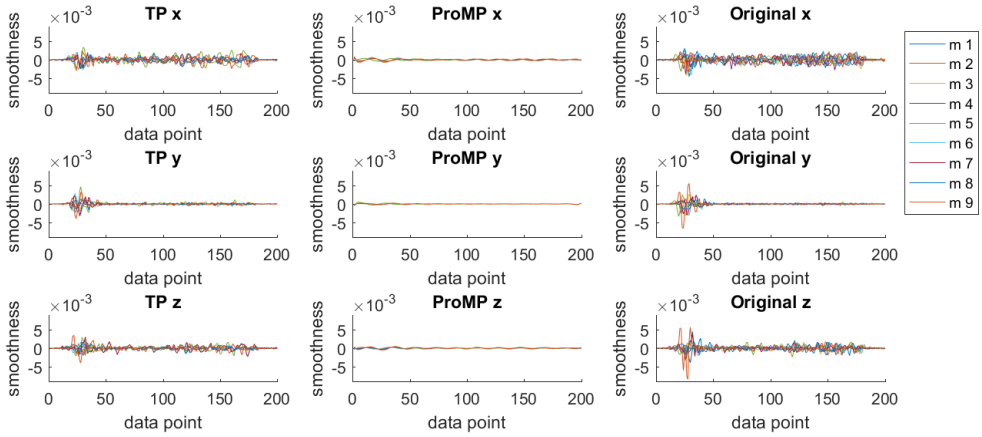


Figure B.43: The smoothness per trajectory (m) in Test 1 for the Drink task. Over the whole trajectory peaks in the change of acceleration rate are visible, in the y direction because of the lifting up the hand the the position and in x and z because of the signature movement.

B.4.2. TEST 2

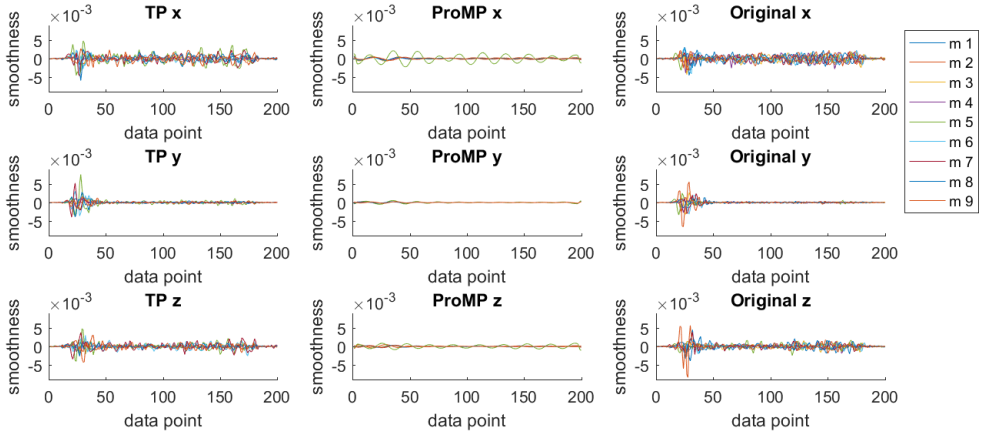


Figure B.44: The smoothness per trajectory (m) in Test 1 for the Drink task. Over the whole creates peaks in the change of acceleration rate are visible, in the y direction because of the lifting up the hand the the position and in x and z because of the signature movement.

BIBLIOGRAPHY

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

[2] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, *Robot programming by demonstration,* in *Handbook of Robotics*, edited by B. Siciliano and O. Khatib (Springer, Secaucus, NJ, USA, 2008) pp. 1371–1394.

[3] S. Calinon, *A tutorial on task-parameterized movement learning and retrieval,* Intelligent Service Robotics **9**, 1 (2016).

[4] D. Lee, C. Ott, and Y. Nakamura, *Mimetic communication model with compliant physical contact in human—humanoid interaction,* The International Journal of Robotics Research **29**, 1684 (2010).

[5] S. Ikemoto, H. B. Amor, T. Minato, B. Jung, and H. Ishiguro, *Physical human-robot interaction: Mutual learning and adaptation,* IEEE robotics & automation magazine **19**, 24 (2012).

[6] H. B. Amor, G. Neumann, S. Kamthe, O. Kroemer, and J. Peters, *Interaction primitives for human-robot cooperation tasks,* in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2014) pp. 2831–2837.

[7] S. Calinon, *A Programming by Demonstration Model combining Satistics, Dynamical Systems and Optimal Control- Technical report,* (2015).

[8] J. Silvério, L. Rozo, S. Calinon, and D. G. Caldwell, *Learning bimanual end-effector poses from demonstrations using task-parameterized dynamical systems,* in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2015).

[9] S. Calinon, D. Bruno, and D. G. Caldwell, *A task-parameterized probabilistic model with minimal intervention control,* in *2014 IEEE International Conference on Robotics and Automation* (2014).

[10] L. Rozo, S. Calinon, D. Caldwell, P. Jimenez, and C. Torras, *Learning collaborative impedance-based robot behaviors,* in *Twenty-Seventh AAAI Conference on Artificial Intelligence* (2013) pp. 1422–1428.

[11] M. S. Malekzadeh, S. Calinon, D. Bruno, and D. G. Caldwell, *Learning by imitation with the stiff-flop surgical robot: a biomimetic approach inspired by octopus movements,* Robotics and Biomimetics **1**, 1 (2014).

[12] J. Hoyos, F. Prieto, G. Alenyà, and C. Torras, *Incremental learning of skills in a task-parameterized gaussian mixture model.* Journal of Intelligent and Robotic Systems **82**, 81 (2016).

[13] F. Stulp, G. Raiola, A. Hoarau, S. Ivaldi, and O. Sigaud, *Learning compact parameterized skills with a single regression,* in *Humanoid Robots (Humanoids), 2013 13th IEEE-RAS International Conference on* (IEEE, 2013) pp. 417–422.

[14] L. Shang and A. B. Chan, *On approximate inference for generalized gaussian process models,* arXiv preprint arXiv:1311.6371 (2013).

[15] Q.-T. Luong and O. D. Faugeras, *The fundamental matrix: Theory, algorithms, and stability analysis,* International journal of computer vision **17**, 43 (1996).

[16] Z. Zhang, *Determining the epipolar geometry and its uncertainty: A review,* International journal of computer vision **27**, 161 (1998).

[17] N. Tsunashima, T. Awakura, M. Koide, J. Yamaguchi, and M. Nakajima, *Measuring distances to objects using rotating mirror system,* Systems and Computers in Japan **30**, 43 (1999).

[18] B. Paláncz, P. Zaletnyik, J. Awange, and B. Heck, *Extension of the abc-procrustes algorithm for 3d affine coordinate transformation,* Earth, planets and space **62**, 857 (2010).

[19] P. H. Schönemann, *A generalized solution of the orthogonal procrustes problem,* Psychometrika **31**, 1 (1966).

[20] A. Dotterer, *Using procrustes analysis and principal component analysis to detect schizophrenic brains,* (2006).

[21] C. Goodall, *Procrustes methods in the statistical analysis of shape,* Journal of the Royal Statistical Society. Series B (Methodological) , 285 (1991).

[22] H. Ragheb and N. A. Thacker, *Morphometric shape analysis with measurement covariance estimates.* in *BMVC* (2011) pp. 1–11.

[23] M. B. Stegmann and D. D. Gomez, *A brief introduction to statistical shape analysis,* Informatics and mathematical modelling, Technical University of Denmark, DTU **15** (2002).

[24] J. Verbeek, N. Vlassis, and B. Krose, *Procrustes analysis to coordinate mixtures of probabilistic principal component analyzers,* (2002).

[25] L. Igual, X. Perez-Sala, S. Escalera, C. Angulo, and F. De la Torre, *Continuous generalized procrustes analysis,* Pattern Recognition **47**, 659 (2014).

[26] H. Li, X. Huang, J. Huang, and S. Zhang, *Feature matching with affine-function transformation models,* IEEE transactions on pattern analysis and machine intelligence **36**, 2407 (2014).

[27] A. D. J. Cross and E. R. Hancock, *Graph matching with a dual-step em algorithm,* IEEE Transactions On Pattern Analysis And Machine Intelligence **20**, 1236 (1998).

[28] T. K. Dey, F. Janoos, and J. A. Levine, *Meshing interfaces of multi-label data with delaunay refinement,* Engineering with Computers **28**, 71 (2012).

[29] S. Calinon, *Linear Regression,* `http://calinon.ch/misc/EE613/EE613-slides-8.pdf` (Nov. 2015), [Lecture slides course Machine learning for engineers].

[30] O. Sorkine, *Least-squares rigid motion using svd,* Technical notes **120** (2009).

[31] I. L. Dryden and K. V. Mardia, *Statistical shape analysis*, Vol. 4 (J. Wiley Chichester, 1998).

[32] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, *Dynamical movement primitives: learning attractor models for motor behaviors,* Neural computation **25**, 328 (2013).

[33] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis* (Pearson, 2014).

[34] C. M. Bishop, *Pattern Recognition and Machine Learning* (Springer, 2006).

[35] G. Maeda, M. Ewerton, R. Lioutikov, H. B. Amor, J. Peters, and G. Neumann, *Learning interaction for collaborative tasks with probabilistic movement primitives,* in *2014 IEEE International Conference on Humanoid Robots (Humanoids)* (IEEE-RAS, 2014) pp. 527–534.

GLOSSARY

LIST OF ACRONYMS

DMP	Dynamic Motor Primitive
DoF	Degrees of Freedom
DTW	Dynamic Time Warping
GMM	Gaussian Mixture model
IIT	Istituto Italiano di Tecnologia
IP	Interaction Primitive
PbD	Programming by Demonstration
ProMP	Probabilitic Movement Primitives
PS	Procrustes superimposition
RMSE	Root mean squared error
SVD	Singular Value Decomposition
TP-inference	Task Parameter inference
TUDelft	Delft University of Technology

LIST OF SYMBOLS

α_y	Constant coefficient of a critically damped system	9,10
β	Scaling parameter	8,9,10
β_y	constant coefficient of a critically damped system	9
ϵ_y	Zero-mean i.i.d. Gaussian noise	12
Λ	Diagonal matrix with positive elements, except possibly the last element	9
λ	Constant value	14
μ	Mean	30,33,36,38
$\boldsymbol{\mu}^X$	Mean coordinate of the data set	8
$\boldsymbol{\mu}$	Mean vector of a Gaussian	11,15,16,18
$\boldsymbol{\Sigma}$	Covariance matrix of a Gaussian	11,14,15,16,18
$\boldsymbol{\Sigma}_D$	Observation noise	18
σ	Standard deviation	30,33,36,38
σ_i	Width of the basis function i	9
τ	Time constant	9,10
$\boldsymbol{\Phi}$	Matrix consisting of basis vectors for all time steps	10
Ψ_i	Gaussian basis function i	9,11,12
$\dot{\Psi}_i(t)$	Gaussian basis function of the velocity i	11,12

$\mathbf{A}$	Rotation matrix	4,7,8,9
$\boldsymbol{A}_p^{(q)}$	Rotation matrix for demonstration q and frame p	5,14,16
$\mathbf{b}$	Translation vector	4,8
$\boldsymbol{b}_p^{(q)}$	Translation vector	5,14,16
c_i	Center of the basis function i	9
$\mathbf{F}$	Forcing function matrix	10
f(x)	Forcing function	9
g	Goal state	9
$\boldsymbol{H}_t$	observation matrix, with on its diagonal the basis functions for the observed stated and zeros for the unobserved stated	18
$p(\cdot)$	Gaussian probability density function	15,18
$\boldsymbol{q}(t)$	Position vector of different joints	11,12,18
$\dot{\boldsymbol{q}}(t)$	Velocity vector of different joints	11
$\mathbf{S}$	Rectangular diagonal matrix with non-negative real numbers on the diagonal in SVD	8

U	Real of complex unitary matrix, final rotation in SVD	8,9
V	Real of complex unitary matrix, initial rotation in SVD	8,9
V	Matrix of which the columns are eigenvectors	14
V	Diagonal matrix of which the eigenvalues on its diagonal	14
w	Weight vector	9,10,11,12,„18
X	Data set $\in \mathbb{R}^{N \times D^I}$ describing the input data	7,8,9,14,15
Y	Data set $\in \mathbb{R}^{N \times D^O}$ describing the output data	7,8,9
y	State variable (e.g. the joint angle that needs to be controlled)	9,10,11
$\dot{y}$	Velocity of the state variable	9,10
$\ddot{y}$	Acceleration of the state variable	9,10