# Robot Cooperative Behavior Learning using Single-Shot Learning from Demonstration and Parallel Hidden Markov Models

Jean-Francois Lafleche, Shane Saunderson, *Student Member*, *IEEE*, and Goldie Nejat, *Member, IEEE*

***Abstract*— For robots to become collaborative assistants, they need to be capable of naturally interacting with users in real environments. They also need to be able to learn new skills from non-expert users. In this paper, we present a novel Parallel Hidden Markov Model (PaHMM) architecture for Learning from Demonstration (LfD), which allows a robot to learn a sequence of cooperative and non-cooperative behaviors from a single demonstration (single-shot) of a task-based human-robot interaction from two interacting teachers. During teaching, the robot learns both a human-robot interaction model and an object interaction model in order to be able to effectively determine its own behaviors. Experiments with a Baxter robot and several teachers were conducted to validate the ability of the robot to learn both its cooperative and non-cooperative behaviors during a task-based interaction. Comparison experiments also show the robustness of our approach to spatial variations from the demonstrated behaviors and tracking errors when compared to other approaches.**

***Index Terms*—Learning from Demonstration, Physical Human-Robot Interaction, Model Learning for Control**

## I. Introduction

As robots are being used for diverse roles in society, their need to learn from non-expert users is increasing [1]. They need to be capable of learning interactive behaviors in order to engage in human-robot cooperative tasks such as assisting the elderly with activities of daily living [2], [3], supporting nurses/doctors [4], and collaborative assembly [5], [6]. The numerous behaviors that are required by different tasks in each application make it impossible for each potential scenario to be anticipated and pre-programmed by expert designers. Therefore, there is a growing need for robots to learn new interactive behaviors from non-expert users to engage in human-robot interactions (HRI).

Learning from Demonstration (LfD) can be used to teach a robot interactive behaviors either through observations of a single human teacher [3], [7], [8] or interactions between two teachers [9]–[14]. Once the robot learns a task, it can then become an interaction partner in HRI scenarios with human users. The advantage of LfD using two teachers allows a robot to learn the relationships between the behaviors of the collaborative partners directly. Previous work in this area has largely focused on learning interactions composed of single cooperative gesture-response behaviors [7]–[11], where the robot learns a specific behavior as a response to a single human behavior. Alternatively, in [14], a robot learned a sequence of behaviors, but every behavior was assumed to be cooperative, where the robot responded to user actions.

In contrast to the aforementioned literature, our research uniquely focuses on a robot learning a sequence of both cooperative and non-cooperative behaviors. Cooperative behaviors consist of a robot directly responding to human partner behaviors, while non-cooperative behaviors are independent of partner actions. The importance of learning both cooperative and non-cooperative behaviors is that a number of tasks can consist of both types of behaviors. For example, in an airport environment, a robot may be required to retrieve a bag from a human traveler, scan its tag to record information and then give the bag back to the traveler. This interaction is composed of a sequence of three distinct behaviors: human-to-robot handover, robot scanning a tag, and robot-to-human handover. While the two handover behaviors require the robot to respond to the human partner's movements using cooperative behaviors, scanning is performed independently in a non-cooperative behavior.

In this paper, we present the development of a novel Parallel Hidden Markov Model (PaHMM) architecture in order for a robot to learn both cooperative and non-cooperative behaviors. The unique use of two HMMs allows to the robot to distinguish between cooperative and noncooperative behaviors in order to engage in tasks requiring both types of behaviors. During the learning stage, one HMM is used to model the behaviors displayed during the interaction and a second HMM is used to model the relationship between the teachers and the object to handover. By modeling the object itself we are able to distinguish between different behaviors with similar joint trajectories. Our technique is used with a two teacher, single demonstration LfD approach in order for a robot to learn interaction tasks consisting of a temporal sequence of cooperative and non-cooperative behaviors. The advantage of using a PaHMM structure is that during the interaction phase the robot's behaviors are robust to variations in the demonstrated behaviors and sensory noise/tracking errors by considering predictions from two different models.

## II. Related Work

LfD approaches can be used to teach new skills to robots without requiring robotics expertise. The approaches can be categorized as either: 1) learning from a single teacher, or 2)

Manuscript received June 22, 2018; revised September 28, 2018; accepted November 9, 2018; current version November 24, 2018.

This paper was recommended for publication by Editor Dongheui Lee upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), the Canada Foundation for Innovation (CFI), and the Canada Research Chairs (CRC) program.

J. F. Lafleche, S. Saunderson, and G. Nejat are with the Autonomous Systems and Biomechatronics Lab in the Department of Mechanical and Industrial Engineering at the University of Toronto (email {jf.lafleche, shane.saunderson}@mail.utoronto.ca, nejat@mie.utoronto.ca).

Digital Object Identifier (DOI): see top of this page.

learning from cooperation of two teachers. Regardless of the number of teachers, there are many Gaussian, trajectory-based, regression, and mixed approaches to LfD [15].

### A. Learning from a Single Teacher

Learning behaviors from a single teacher can occur through tele-operation or kinesthetic teaching, where a teacher directly or indirectly physically moves the robot, e.g. [3], [16], [17], or by observations, where the joint movements of a teacher are observed using sensors and then mapped onto the robot, e.g. [7], [8], [18], [19].

An example of teleoperation is presented in [3], where a teacher tele-operated the Tangy robot within a simulated environment for it to learn to facilitate a multi-user Bingo game. Through repeated demonstrations of the activity and a random forest classifier, the robot was able to learn the state-behavior pairs that represented the rules of the game as well as assistive verbal and nonverbal behaviors. After learning, Tangy was able to autonomously facilitate a Bingo game.

An example of kinesthetic LfD is demonstrated in [17], where participants directly manipulated an upper-torso humanoid robot with 7 degree of freedom (DOF) arms in order to teach it an object handling task. Using keyframe demonstrations, two separate learning models were trained. An action model generated state paths between subsequent keyframes to build the overall trajectory. Meanwhile, a goal model monitored the execution of the action model against expected keyframes to validate completion. In experiments, the goal model was on average 90% correct at monitoring task execution, however, the action model was only 66% successful at providing valid end-effector trajectories.

Kinesthetic LfD was also used in [20] to create a Gaussian HMM of probabilistic movement primitives (interaction ProMPs). A 7 DOF KUKA arm interacted with people through several cooperative tasks such as handover and assembly. Multiple training demonstrations provided diversity in the variance in task, uncertainty of execution, and exploration noise. The demonstration of multiple behaviors allowed the training of several interaction ProMPs. During interaction experiments, the model identified the most probable ProMP and then determined the robot's movements to align with the observed human movements.

Recurrent Neural Network (RNN) [7], and HMM [8], [18] techniques have also been used to model a sequence of behaviors through observations. In [7], a teacher demonstrated rhythmic hand movement patterns to the QRIO humanoid robot. The teacher repeated demonstrations of the actions, while a vision system tracked hand trajectories. A RNN with parametric bias (RNNPB) was used to learn the movement patterns to mirror. In [8], the IRT humanoid robot learned interactive behaviors such as high five gestures. The gestures were each demonstrated 5 times and detected using a motion capture system. A mimetic communication model was used where motion primitives learned using continuous HMMs and interaction primitives were learned using discrete HMMs. When learning the interaction primitives, the robot performed a learned motion pattern, and observed the user's reaction.

### B. Learning from Cooperation of Two Teachers

LfD techniques that have used two teachers to learn cooperative behaviors usually have one teacher represent the *user*, while the second teacher is the *interaction partner* whose behavior will be performed by a robot [9]–[14]. A number of different techniques have been proposed to achieve this including HMMs [9],[10], Probabilistic Principal Component Analysis (PPCA) [10], Dynamic Motion Primitives (DMPs) [11]–[13], and the combination of both DMPs and HMMs [14].

In [9], a mimetic communication model consisting of a hierarchical grouping of HMMs was used to learn the behavior of two teachers demonstrating kick-boxing. A motion capture system was used to record teacher joint positions. During the learning phase, the teachers performed repeated demonstrations of cooperative interaction patterns. Motion patterns were segmented and modeled by an HMM and the mimetic communication model was used to determine the robot's motions from the human's motion patterns. Experiments with a user interacting with a virtual robot and the UT-$\mu$2 physical humanoid robot showed that the robot could recognize the user's motions and respond.

In [10], robot learning from a dataset of multi-shot recorded interactions such as attacking and blocking between two teachers was achieved using two different techniques: a PPCA approach and HMMs representing the joint trajectories of the teachers. During interactions with the user, the observed joint trajectories were used to predict the most likely behavior and then find the corresponding *interaction partner* parameters. Once trained, a Nao robot could recognize user gestures and respond appropriately.

In [11], object handover tasks were learned from observing two teachers with both single-shot and multi-shot approaches. During a demonstration, a DMP model was learned for each motion pattern. The wrist trajectory of the *interaction partner* was represented as a second-order linear dynamical system influenced by two attractor fields, representing the demonstrated trajectory and the goal position. By varying the weights of the two fields, the model was able to generate planned trajectories that adapted to user wrist trajectories. In [12] and [13], this technique was used to learn human-robot handover tasks. The learned trajectories were applied to a KUKA manipulator. Experiments demonstrated the generation of arm trajectories that enabled fluent, successful human-robot and robot-human handovers.

The aforementioned approaches have focused on a robot learning individual cooperative gesture-response pairs. Alternatively, in [14], an approach using Interaction Meshes (IM) and an HMM was presented to teach a robot arm a *sequence* of cooperative behaviors for an assembly task. During a single-shot demonstration joint information was obtained using a motion capture system and the IMs were used to model the relationships between the joints of the two teachers. To determine which IM to apply when responding to the behavior of a user, the robot also learned a HMM-based representation of the sequence of behaviors that made up the interaction. Experiments consisted of cooperative tasks, such as Lego assembly.

In general, the majority of LfD approaches assume that a user behavior directly results in a corresponding cooperative robot behavioral response. However, interaction tasks exist that can alternate between cooperative and non-cooperative behaviors. Examples of such tasks include baggage handling and parts assembly.

In our work, we uniquely utilize LfD to teach a robot a physical, task-based interaction composed from cooperative and non-cooperative behaviors using a single-shot demonstration performed by two teachers. We implement a PaHMM structure that consists of a novel combination of a *human-robot interaction model* and *object interaction model*. As previously mentioned, the integration of the *object interaction model* allows the robot to differentiate between similar joint trajectories belonging to a diverse group of behaviors and provides robustness to variations in the observed behaviors from those demonstrated.

## III. Learning From Human Interactions

Our PaHMM structure is integrated into the proposed overall system architecture shown in Fig. 1. The architecture consists of a learning phase and an interaction phase. In the learning phase, the robot observes a task demonstrated by two human teachers in order to learn the sequence of behaviors of that interaction using a PaHMM. In the interaction phase, the robot then undertakes the learned task with a user by executing the sequence of learned behaviors. Each phase is discussed in detail below.

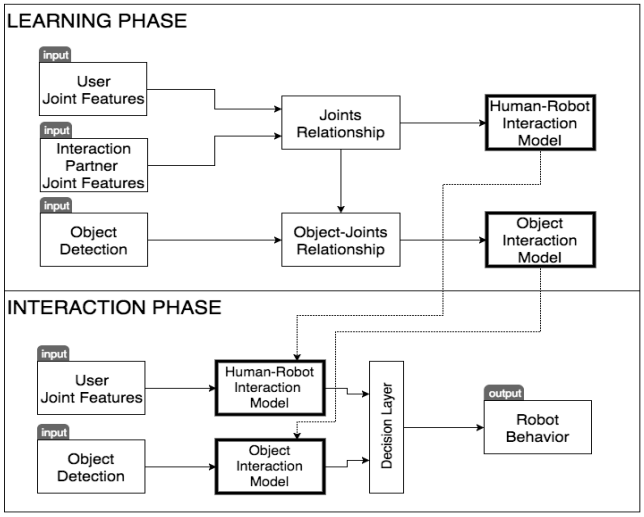



Fig. 1. System architecture for cooperative and non-cooperative behavior learning from human interactions with learning and interaction phases.

### A. Learning Phase

The goal of the *learning phase* is to model the observed interaction between two teachers. One teacher represents the *user* in the interaction and the other the *interaction partner,* which is the role of the robot. We focus on interactions which are composed of a sequence of behaviors, rather than a single behavior-response pair. Segmentation of the individual behaviors is based on both teachers returning to a resting pose after each behavior is displayed during an interaction. Our approach has been designed to be able to learn from a single demonstration, therefore, minimizing the need for multi-shot learning. This is achieved by learning two models in parallel that observe different aspects of the *user* behavior, providing complementary predictions, and combining these predictions to form a successful prediction even when variations from the demonstration behavior are observed. The two models are 1) a *human-robot interaction model*, and 2) an *object interaction model*. Our approach learns both models per behavior to obtain robust recognition and allow a robot to implement both the cooperative and non-cooperative behaviors during the interaction.

The *human-robot interaction model* receives as input the joint parameters of the two teachers which consist of both joint angles and angular velocities. The relationship between the joint parameters of the two teachers is defined as the *joints relationship* and expresses how one teacher moves in response to the other. The *object interaction model* receives as input the spatial position of a detected object in the interaction task. The object position is used with the joint parameters and relationship of the two teachers in order to define the *object-joints relationship*. Variations from the demonstration can make it difficult to distinguish between similar behaviors. However, as the two models are independently observing the demonstration using distinct features, similar behaviors in the *human interaction model* (i.e. take, give) can be represented differently by the *object interaction model* (i.e. object with robot, object with user). Therefore, these two models can complement each other when determining an interaction behavior. If one of the models is not able to distinguish between similar behaviors based on the observations, the predictions from the other model are used to eliminate low probability behaviors and increase the total prediction probability of the system.

#### 1) Human-Robot Interaction Model

The objective of the *human-robot interaction model* is to represent the sequence of joint parameters of the two teachers. Each demonstrated behavior, $\lambda_m^{HRI}$, during the interaction is represented by an HMM to form a library of $M$ learned behaviors $\{\lambda_1^{HRI}, \lambda_2^{HRI}, \dots, \lambda_M^{HRI}\}$. There are $N$ possible interaction states $S = \{S_1^{HRI}, S_2^{HRI}, \dots, S_N^{HRI}\}$. Here $N$ is set to 20. $q_t^{HRI}$ denotes the state at time $t$. Initial state probabilities are represented by $\Pi^{HRI} = \{\pi_i^{HRI}\}$, where the state probability $\pi_i^{HRI}$ is the probability that the initial state interaction state $q_1^{HRI}$ at $t = 1$ corresponds with the state $s_i^{HRI}$ according to:

$$\pi_i^{HRI} = P(q_1^{HRI} = s_i^{HRI}),\ 1 \le i \le N. \quad (1)$$

State transitions are represented by the transition probability matrix $A^{HRI} = \{a_{ij}^{HRI}\}$, where $a_{ij}^{HRI}$ represents the probability of transitioning from state $q_{t-1}^{HRI}$ at time $t-1$ to the state $q_t^{HRI}$ at time $t$ according to:

$$a_{ij}^{HRI} = P\left(q_t^{HRI} = s_j^{HRI} | q_{t-1}^{HRI} = s_i^{HRI}\right),\ 1 \le i,j \le N. \quad (2)$$

To represent the temporal nature of the behaviors, a left-right transition probability distribution is used where the hidden states can either transition to themselves or to the state directly to their right according to the constraint:

$$a_{ij}^{HRI} = 0, \quad j > i + 1. \quad (3)$$

The observation probability distribution is represented by $B^{HRI} = \{b^{HRI}(o_t^{HRI})\}$, where $b(o_t^{HRI})$ is the probability density function for an observation $o_t^{HRI}$ representing the joint parameters of the *user* (e.g., joint angles and angular velocities). Since joint observations occur in a continuous space, the observation probabilities are modeled as a Gaussian HMM. The parameters of the model are estimated using the Expectation Maximization (EM) algorithm [21]. The transition, observation, and initial state parameters for the

HMM representing a behavior, $\lambda^{HRI} = (A^{HRI}, B^{HRI}, \Pi^{HRI})$, are estimated using the Baum-Welch algorithm.

The robot begins observing the demonstration when the user makes a movement. To avoid learning unintentional movements, behaviors that are less than 1 sec in length are rejected. If the *interaction partner* behavior is accompanied by actions from the *user*, the behavior is cooperative. However, for a non-cooperative behavior, the behavior of the *interaction partner* is not dependent on any user behavior. Our approach defines a non-cooperative behavior as one where the *interaction partner* initiates a behavior while the *user* is not engaging in a behavior. Therefore, cooperative behaviors require actions taken by both the *user* and *interaction partner*. This is identified by observing the difference in joint angles of the user from time $t-1$ to $t$:

$$c_t = \begin{cases} 1, & o_t^{HRI} - o_{t-1}^{HRI} > \alpha \quad cooperative \\ 0, & o_t^{HRI} - o_{t-1}^{HRI} \leq \alpha \quad non-cooperative \end{cases}, \quad (4)$$

where $c_t$ represents the behavior type, $o_t$ represents the observed joint parameters of the *user* at time $t$, and $\alpha$ represents a limit above which a behavior is defined as cooperative. $\mathrm{t}$ and $\alpha$ are set to appropriately classify cooperative and non-cooperative behaviors while accounting for small unintentional movements as well as sensor noise when detecting joint angles. The behavior type $c_t$ is evaluated at every time step $t$ to determine whether the observed movement is performed cooperatively. While behaviors are typically entirely composed of one type (i.e. non-cooperative: robot waving at user, and cooperative: hand-over), it is possible for a behavior to mix both cooperative and non-cooperative actions as in the case where there is a delayed reaction from a teacher. There is no time limit defined for a response to a cooperative behavior, as behaviors vary in duration, therefore, a delayed response would also be considered a cooperative behavior. Since the objective of our architecture is for a robot to learn behaviors from demonstration, we have not explicitly modeled non-cooperative behaviors of the user interacting with the robot.

In order to complete tasks such as object handovers, the robot needs to be capable of adapting its joint trajectory to the user's joint trajectory. The objective is to learn a response that consists of *the interaction partner*'s joint angles $\theta = \{\theta_1, \theta_2, \dots, \theta_N\}$ as well as the translation vector between the wrist joint positions of the two teachers $\mathrm{k} = \{\mathrm{k}_1, \mathrm{k}_2, \dots \mathrm{k}_N\}$.

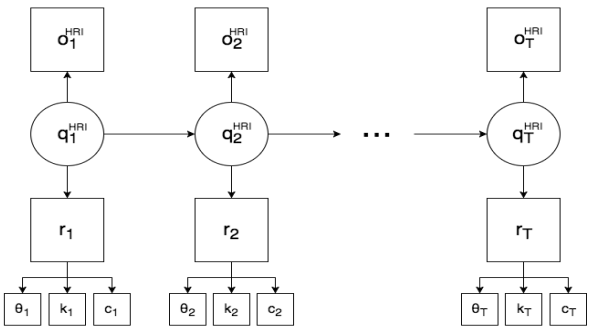



Fig. 2. Human-robot interaction model.

The robot behavior is represented by $\mathrm{R} = \{\mathrm{r}_1, \mathrm{r}_2, \dots, \mathrm{r}_N\}$, where $\mathrm{r}_n = \{\theta_n, \mathrm{k}_n, \mathrm{c}_n\}$. The overall human-robot interaction model is shown in Fig. 2. The model maps each hidden state $S_n^{HRI}$ to both the user's observed joint parameters $o_n^{HRI}$ and to a robot response behavior $r_n$. When learning the HMM, the robot also learns a mapping between each hidden state $\mathrm{S}_\mathrm{n}^{\mathrm{HRI}}$ to the corresponding robot response behavior $\mathrm{r}_\mathrm{n}$. When interacting with the user, the robot is then able to estimate the current interaction state $\mathrm{S}_\mathrm{t}^{\mathrm{HRI}}$ and retrieve the corresponding response $\mathrm{r}_\mathrm{t}$.

*2) Object Interaction Model*

A problem occurs in the *human-robot interaction* model when a task is composed of behaviors having similar joint trajectories. For example, a handover behavior and a reaching behavior can be identical in the representation of their joint trajectories but can be distinguishable based on the object. In order to capture the sequential nature of behaviors within an object interaction task, an *object interaction model* is learned using left-right HMMs that model the spatial relationship between a detected object and the wrists of the teachers over the course of the interaction.

Each demonstrated behavior, $\lambda_m^{OBJ}$, is represented by an HMM to form a library of $M$ learned behaviors $\{\lambda_1^{OBJ}, \lambda_2^{OBJ}, \dots, \lambda_M^{OBJ}\}$ that correspond to the same behaviors used in the *human-robot interaction model*. Each HMM has $P$ hidden object states $S = \{S_1^{OBJ}, S_2^{OBJ}, \dots, S_P^{OBJ}\}$, initial state probabilities $\Pi^{OBJ} = \{\pi_i^{OBJ}\}$, transition matrix $A^{OBJ} = \{a_{ij}^{OBJ}\}$, and observation probability distribution $B^{OBJ} = \{b^{OBJ}(o_t^{OBJ})\}$.

Observations $o_t^{OBJ}$ are extracted from the Euclidean distance between the center point of the detected object and the centroid of the torsos of the user and of the interaction partner and discretized to four object states: (no object, with *user*, with *interaction partner*, between *user* and *interaction partner*). The *object interaction model* emits an observation $o_t^{OBJ}$ for each state $q_t^{OBJ}$, Fig. 3.

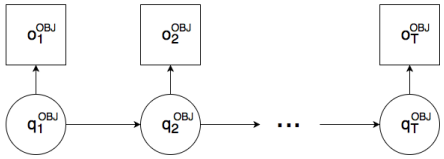



Fig. 3. Object Interaction model.

## B. Interaction Phase

The goal of the robot during the interaction phase is to collaboratively complete a task with the user. In recognition approaches using HMMs, observations of a demonstration are typically collected as a sequence of features which are used to train a single HMM model. When the detection of observed features is affected by noise, recognition accuracy can become degraded. To minimize the effect of noise and tracking errors and account for variations in demonstrated behaviors, we propose a PaHMM structure, where the observed features are split into separate feature sets, that are used to train independent HMM models, seen in the lower half of Fig. 1. The noise/errors present in one feature set will only affect the recognition accuracy of its corresponding HMM. PaHMMs were first successfully introduced for automatic speech recognition [22], [23]. Once we have the HMM models, a

decision layer combines the output of each model to produce a final recognition determination.

Our implementation uses two sets of HMMs: the Human-Robot Interaction Model, and the Object Interaction Model which receive human joint parameters $O_t^{HRI} = \{o_t^{HRI}, o_{t-1}^{HRI}, \dots, o_{t-F}^{HRI}\}$ and object features $O_t^{OBJ} = \{o_t^{OBJ}, o_{t-1}^{OBJ}, \dots, o_{t-F}^{OBJ}\}$, respectively. A sequence of observation of length $F$ is used in a sliding window to obtain the likelihoods of both $\lambda_m^{HRI}$ and $\lambda_m^{OBJ}$ for each learned behavior $m$, namely $P(O_t^{HRI}|\lambda_m^{HRI})$ and $P(O_t^{OBJ}|\lambda_m^{OBJ})$. The probability of the observations given each behavior HMM is evaluated simultaneously in both models and combined in a decision layer to make a final prediction of the observed user behavior. This final prediction is used to determine the appropriate robot behavior. We propose a simple recombination strategy where the two models are combined with a weighted sum according to:

$$m^* = \arg\max_m \left( P(O_t^{HRI}|\lambda_m^{HRI}) + \gamma \cdot P(O_t^{OBJ}|\lambda_m^{OBJ}) \right), \quad (5)$$

where $\gamma$ is a weighting factor for the *object interaction model.* We can obtain $\gamma$ empirically by creating a test dataset of repeated behaviors and selecting a value that maximizes accurate behavior predictions. In this work, $\gamma$ was set to 0.4. The value of $\gamma$ determines the contribution of the two models to the final behavior prediction. A $\gamma$ value of 0 will remove contributions from the *object interaction model*, producing just the HMM with the *HRI model*. A high $\gamma$ value will result in a model that primarily utilizes the object interaction model.

Once an interaction behavior is recognized, the current state of the behavior is estimated from the *human-robot interaction model* using the Viterbi algorithm [24] to find the sequence of interaction states $Q^{HRI} = \{q_1^{HRI}, q_2^{HRI}, \dots, q_t^{HRI}\}$ representing the sequence of user joint feature observations $O_t^{HRI}$, to maximize the probability $P(Q^{HRI}, O_t^{HRI}|\lambda^{HRI})$.

The robot behavior can then be determined from the robot response $r_t$ that corresponds to the state $q_t^{HRI}$. If the robot response is cooperative, this process will be repeated with new observations at time $t+1$. However, if the robot response is non-cooperative, the robot directly proceeds to the next behavior state in the interaction sequence $q_{t+1}^{HRI}$ and initiates the corresponding robot behavior $r_{t+1}$.

For both cooperative and non-cooperative behaviors, the robot must generate joint commands that correspond with the intended robot behavior. The *interaction partner* learned joint angles, $\theta_t$, can be mapped to the robot's joints taking into account the kinematics of both the human and robot arms. The robot arms have the same DOFs as human arms. When the DOFs are different, imitation techniques using optimization can be implemented.

If the robot was to rely on the demonstrated joints entirely, it would be unable to adapt to variations in user behavior, particularly ones that require the user's hand and robot end-effector to meet. As a solution, we project the learned wrist joint relationship $k_t$ onto the user's current wrist location to determine the target position of the robot's end-effector. We then determine the corresponding robot joint angles $\phi_t$ using an Inverse Kinematics (IK) solver. Since IK joint angles $\phi_t$ differ from the learned joints $\theta_t$, we incorporate $\theta_t$ and $\phi_t$ in a weighted sum based on the distance between wrists, $d_t$. When the robot and user wrists are far apart (i.e. a wave), $w(d_t)$ approaches 0 and the system utilizes the learned joint angles, whereas when wrists are close (i.e. a handover), $w(d_t)$ approaches 1 and the system utilizes the IK joints to obtain the robot joint angles $\theta_t^*$:

$$\theta_t^* = \phi_t \cdot w(d_t) + \theta_t \cdot \big(1 - w(d_t)\big). \quad (6)$$

The weight $w(d_t)$ is given by:

$$w(d_t) = e^{-\frac{d_t}{D}}, \quad (7)$$

where $d_t$ is the Euclidean distance between the nearest neighboring learned wrist joints of the teachers at state $q_t^{HRI}$, and $D$ is a constant that determines the weighting of $\theta_t$ and $\phi_t$ at distance $d_t$. We set the constant $D$ as the average distance between the centroid of the torsos of the robot and the user during an interaction task which was 0.3m. This value, when considered with the wrist-to-wrist distance $d_t$, determines the weighting of IK and learned joint angles to allow for a gradual transition of the system between movements that follow the demonstrated behavior at larger $d_t$ distances and movements that align to user variations at closer $d_t$ distances. Only $D$, the torso-to-torso distance, and $d_t$, the wrist-to-wrist distances, are represented in Cartesian space. The resultant robot joint angles, $\theta_t^*$, should be an accurate representation of learned behavior adapted to user behavior variations.

## IV. Interaction Experiments

Experiments were conducted to learn an interaction task demonstrated by two teachers. A baggage handling task was considered consisting of a human-robot and robot-human object handover scenario with the number of learned behaviors $M$ set to 4. During the interaction, the robot performs both cooperative and non-cooperative behaviors as shown in Table I. We compared the results of our PaHMM approach to both a single HMM approach from the literature [10], [14], and a single HMM approach we developed which also incorporated object feature states.

TABLE I
BAGGAGE HANDLING INTERACTION TASK SCENARIO

| Seq. # | Human user | Interaction partner | Behavior Type |
|---|---|---|---|
| 1 | Handover object | Take object | Cooperative |
| 2 | Wait | Scan object tag | Non-cooperative |
| 3 | Take object | Hand-over object | Cooperative |
| 4 | Wave | Wave back | Cooperative |

### A. Interactive Robotic System

The Baxter robot is a dual-arm robot with 7 DOFs in each arm, shown in Fig. 4. A 1-DOF parallel gripper was incorporated into the end-effector of each arm. An RGB camera is located at the wrist for each arm. A Kinect sensor was also mounted on top of the robot's head to provide both RGB and depth information of the interaction.

Teacher joint tracking is performed by the OpenNI framework [25] together with the NITE skeleton tracking package [26] which uses randomized decision trees trained on a large dataset of poses to estimate joint positions.

For each teacher, the position and orientation of the centroid of the torso, and the wrist, elbow and shoulder joints were tracked. The orientation of the shoulder joints, elbow angle and angular velocity of both joints were concatenated to produce a 16-dimensional vector used as observations in the *human-robot interaction model*.

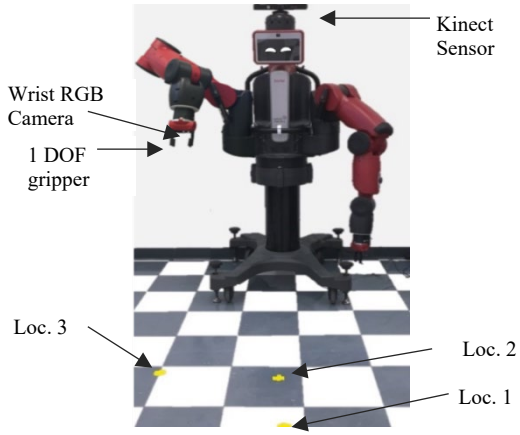



Fig. 4. Baxter Robot used in the experiments. Markers on the floor indicate different user locations used with respect to the robot.

The relative spatial position of the detected object with respect to the teachers was used to obtain the object state observations in the *object interaction model*. Object tracking was achieved using the *TensorBox* detector [27]. This detector uses a Convolutional Neural Network (CNN) trained to produce bounding boxes at image areas with a high probability of containing the object class. The detector was fine-tuned on a dataset of 1,000 examples of bags, backpacks, satchels and boxes. We developed a technique that used depth information from the Kinect sensor to locate potential grasp points by applying a convolution filter to the corresponding depth map of the object, allowing the robot to identify possible grasp locations. This is accomplished by using a Canny edge detector and a Hough transform on the RGB image to remove the background and extract prominent object edges. The depth map is then scaled based on distance to the object and convolutional filters are used to identify potential grasp regions.

The two layers of the *human-robot interaction model* and the *object interaction model* were combined for the PaHMM.

### B. Learning Phase

The collaborative baggage handling scenario was demonstrated to the robot in a single demonstration by two teachers. Each demonstrated behavior is shown in Fig. 5.

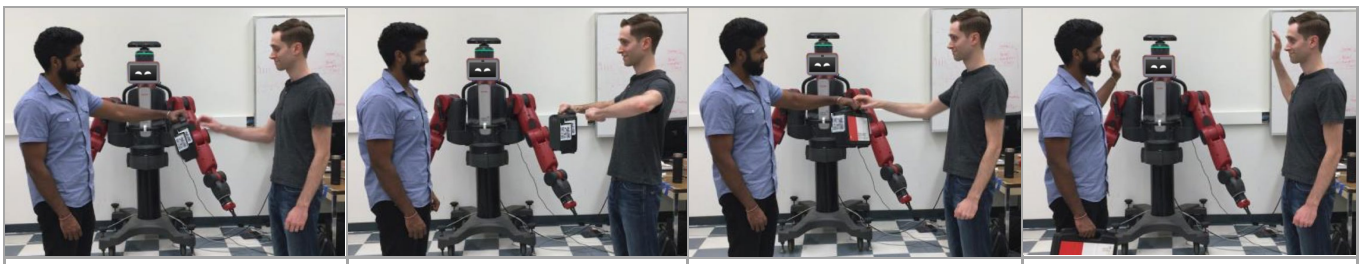
### C. Interaction Phase

The performance of our robot behavior learning system was investigated with respect to: 1) its ability to generate the appropriate robot behaviors during the chosen human-robot interaction scenario, 2) its robustness to variations among the users, and 3) the response time of a cooperative behavior, measured from the time a user behavior is initiated (based on prediction confidence level) up until the time a robot initiates its own response. Five participants (heights ranging from 162 cm to 188 cm) were asked to interact with the robot. The participants were told the sequence of behaviors of the interaction but were not given detailed instructions on how to perform each behavior. To provide additional variation, each participant was asked to perform the interaction at three different positions (Loc. 1, 2, and 3) in front of the robot, as shown in Fig. 4. For each of the behaviors within the interaction scenario, the robot's response time was recorded and, in instances where the robot performed the incorrect behavior or timed-out during the interaction (longer than 10 seconds), a behavior was considered a failure. In order to compare our approach with existing approaches, we conducted the experiment using both our PaHMM model and a single HMM model that did not directly incorporate object features, similar to [10],[14], which we define herein as HRIHMM. Each participant performed the interaction with the robot at the three positions for each of the two approaches on different days using a counterbalance procedure.

### D. Results

The detailed results are presented in Table II and examples of the interactions for two participants and their behaviors are presented in Fig. 6. The completion rates for all the behaviors for the PaHMM were between 87% and 100% ($\mu$=97%, $\sigma$=6%). Behavior response time for each of the cooperative behaviors varied between 0.1s and 4s ($\mu$=1.33s, $\sigma$=1.03s). The PaHMM model had two failures, both for the give behavior. In these instances, the skeletal tracking algorithm provided insufficient data for the PaHMM to make a confident prediction, and example of which is in Fig. 7.

The behavior completion rates for the HRIHMM model ranged between 13% to 100% with $\mu$=30% and $\sigma$=11%. There were numerous instances that the robot did not attempt to implement a behavior due to a previous behavior failing ('-' in Table II) or the robot failed during its attempt ('Fail'). Response time of those behaviors that were implemented varied from 0.1 s to 10 s, with an overall higher mean and standard deviation, i.e., $\mu$=2.72s and $\sigma$=2.57s. Even though the completion time was slightly faster for the take behavior using the HRIHMM model when compared to the PaHMM, the PaHMM model had a 100% success rate for this behavior versus only 40% for the HRIHMM model. Similar to the PaHMM approach, the HRIHMM's failures were due to skeleton tracking errors, however, failures were more frequent for the HRIHMM as it solely used joint data.



| Cooperative Human-Robot Hand-Over | Non-Cooperative Robot Scanning | Cooperative Robot-Human Hand-Over | Cooperative Wave |
|---|---|---|---|

Fig. 5. Teaching the robot an interaction scenario composed of three cooperative behaviors and one non-cooperative behavior.

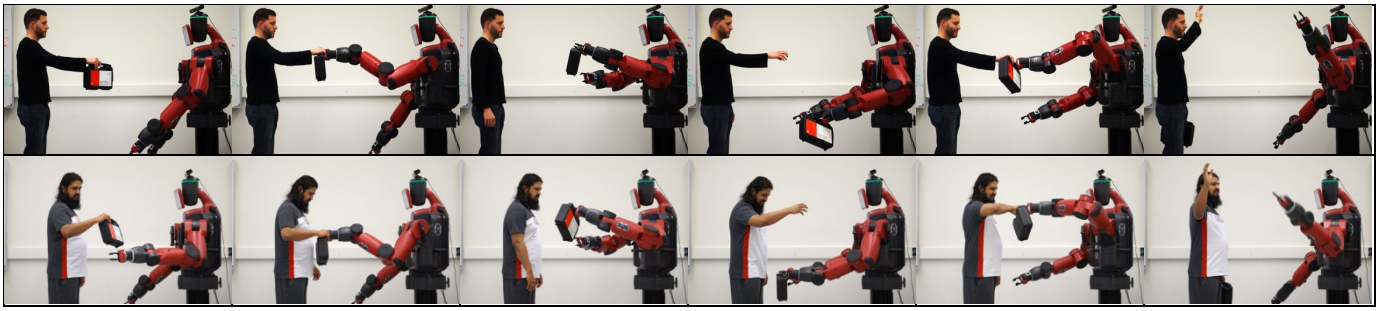


Fig. 6. User interaction examples with the robot generating behaviors during the task using the PaHMM. Variations in joint angles can be seen in interactions.

TABLE II
ROBOT BEHAVIOR RESULTS

| User | Loc. | PaHMM Response Time (s) | | | | HRIHMM Response Time (s) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | *Take* | *Scan* | *Give* | *Wave* | *Take* | *Scan* | *Give* | *Wave* |
| 1 | 1 | 1.6 | N/A | 0.7 | 2.1 | 0.1 | N/A | Fail | - |
| | 2 | 1.8 | N/A | 1.9 | 0.4 | Fail | - | - | - |
| | 3 | 0.2 | N/A | 1.1 | 1.6 | Fail | - | - | - |
| 2 | 1 | 3.9 | N/A | 0.1 | 0.3 | Fail | - | - | - |
| | 2 | 2.1 | N/A | 1.7 | 3 | 1.6 | N/A | 0.6 | Fail |
| | 3 | 0.5 | N/A | 0.2 | 0.2 | 1.2 | N/A | 0.1 | 0.1 |
| 3 | 1 | 0.9 | N/A | Fail | 1.1 | Fail | - | - | 10 |
| | 2 | 3.8 | N/A | 0.2 | 0.3 | Fail | - | - | - |
| | 3 | 3.2 | N/A | Fail | 1 | 0.1 | N/A | 4.5 | Fail |
| 4 | 1 | 4 | N/A | 1 | 0.1 | 1.6 | N/A | 2 | Fail |
| | 2 | 2.5 | N/A | 0.4 | 0.1 | 3.2 | N/A | Fail | - |
| | 3 | 2.2 | N/A | 0.4 | 1.1 | Fail | - | - | - |
| 5 | 1 | 1.1 | N/A | 0.1 | 0.1 | Fail | - | - | - |
| | 2 | 2.3 | N/A | 2 | 0.2 | Fail | N/A | - | - |
| | 3 | 0.5 | N/A | 3.6 | 2 | Fail | N/A | - | - |
| **Avg. Response Time (S.D.)** | | **2.0 (1.2)** | **N/A** | **1.0 (1.0)** | **0.9 (0.9)** | **1.3 (1.1)** | **N/A** | **1.8 (1.7)** | **5.1 (5.0)** |
| **Completion Rate %** | | **100%** | **100%** | **87%** | **100%** | **40%** | **100%** | **27%** | **13%** |

'N/A' is used for the non-cooperative scanning behavior when attempted by the robot as it is not a response to the user. '-' represents a sequential behavior that was not attempted by the robot due to failure in a previous behavior.


Fig. 7. Tracking Failure example on the right arm (yellow).

### *E. Human Behavior Interaction Prediction Comparison*

An additional comparison analysis was run for two reasons. First, the HRIHMM results from the previous experiment had many incomplete behaviors due to the failure of a prior behavior in the sequence. Second, though the PaHMM outperformed the HRIHMM, we wanted to investigate if the performance improvement was due to the design of the approach or simply the addition of object data.

To resolve the first issue, we analyzed the predictions of each model on an input dataset for the baggage handling interaction task using successfully tracked individual behaviors for all five participants. This allowed us to conduct a comparison of different models with the same dataset and the predictions of each model given the user's behavior.

For the second issue, we introduced a third model in this comparison in addition to the PaHMM and HRIHMM approach. We created a single layer HMM that combined object features with the user joint parameters into a single feature vector, that we define as the Human-Object HMM (HOHMM). This approach leverages the same data as the PaHMM, however, uses a single model structure to form its predictions. Results of this analysis are presented in Table III.

Overall, our PaHMM model had high prediction rates for the take (83%), give (100%), and wave (100%) behaviors. The HOHMM model was less successful at predicting wave (94%), give (84%) and take (19%), often misinterpreting take as a give behavior (65%). The HRIHMM model had the lowest prediction rates for wave (71%), give (78%), and frequently mispredicted take (14%) as a give (85%).

While the object features in the HOHMM produced an improvement in prediction of all the cooperative behaviors with respect to the HRIHMM model, the prediction rates were still lower with respect to our PaHMM model, especially for the take behavior. Since both PaHMM and HOHMM used the same inputs, the improved performance of the PaHMM is due to the parallel structure utilizing multiple HMMs combined in the decision layer. Namely, the HOHMM approach determines the probability of a behavior based on how well the entire feature vector matches with the demonstration, and then applies an equal weighting to both the
joint parameters and object features. In contrast, by using two models, the *object interaction model* and the *human-robot interaction model*, the PaHMM is less sensitive to variations in observations that occur due to tracking errors or user behavior variations. Therefore, it can still predict a behavior with high probability even if one of the models outputs low probabilities for all learned behaviors, such as when joints are occluded by the object. Although the HOHMM and PaHMM used the same input parameters, the HOHMM more frequently mistook the take and give behaviors. The lower performance of the HOHMM was due to it using a single feature vector (joints and object features together) for prediction probabilities. Therefore, any reduced information (i.e., occluded joints) or noisy data directly affected the overall prediction. However, the PaHMM uses two separate models which provide independent predictions and are not as affected by the aforementioned issues. For example, occluded joints do not influence the object interaction model prediction as it is only interested in object observations.

TABLE III
CONFUSION MATRIX COMPARISON OF COOPERATIVE BEHAVIOR PREDICTIONS

| | | Actual | | |
|---|---|---|---|---|
| **Predicted** | | **Take** | **Give** | **Wave** |
| **PaHMM** | Take | 83% | 0% | 0% |
| | Give | 0% | 100% | 0% |
| | Wave | 17% | 0% | 100% |
| | None | 0% | 0% | 0% |
| **HOHMM** | Take | 19% | 16% | 0% |
| | Give | 65% | 84% | 6% |
| | Wave | 16% | 0% | 94% |
| | None | 0% | 0% | 0% |
| **HRIHMM** | Take | 14% | 22% | 3% |
| | Give | 85% | 78% | 26% |
| | Wave | 1% | 0% | 71% |
| | None | 0% | 0% | 0% |

## V. CONCLUSIONS

In this paper, we present a novel *LfD* architecture that utilizes a PaHMM structure composed of a *human-robot interaction model* and an *object interaction model* to learn cooperative and non-cooperative behaviors. The parallel use of the two models provides robustness to variations in user behaviors and allows the robot to perform both cooperative behaviors and non-cooperative behaviors during a task. Experiments were conducted with the Baxter robot implementing a baggage handling task. The results showed that the robot learned to interpret user behaviors and respond appropriately while generalizing to user variations.

Performance comparisons with a traditional single layer HMM showed that our proposed approach had, on average, lower robot behavior response times and significantly higher completion rates. Furthermore, our PaHMM model was able to have significantly higher behavior prediction accuracy than both the single layer HMM model (HRIHMM) and a single layer HMM model with object features (HOHMM). Though studied with a baggage handling task, the PaHMM framework can handle other tasks comprised of behaviors that can be defined through a combination of joint parameters/end-effector positions and object features. Depending on the task, tuning the torso-to-torso distance, $D$, and the relative weighting, $\gamma$, may improve performance. Tasks involving additional objects are also possible through the incorporation of new object features into the object interaction model. Therefore, the use of a PaHMM approach can be effective for combining different feature types for a variety of robot collaborative tasks.

## REFERENCES


[1] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robot. Auton. Syst.*, vol. 57, no. 5, pp. 469–483, May 2009.

[2] M. Vincze *et al.*, "Towards a robot for supporting older people to stay longer independent at home," in *Int. Symp. on Robotics*, 2014.

[3] W. Y. G. Louie and G. Nejat, "A learning from demonstration system architecture for robots learning social group recreational activities," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2016, pp. 808–814.

[4] E. Carpintero, C. Pérez, R. Morales, N. García, A. Candela, and J. Azorín, "Development of a robotic scrub nurse for the operating theatre," in *IEEE Int. Conf. on Biomedical Robotics and Biomechatronics*, 2010, pp. 504–509.

[5] J. T. C. Tan, F. Duan, Y. Zhang, K. Watanabe, R. Kato, and T. Arai, "Human-robot collaboration in cellular manufacturing: Design and development," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 29–34.

[6] F. Duan, J. T. C. Tan, J. G. Tong, R. Kato, and T. Arai, "Application of the Assembly Skill Transfer System in an Actual Cellular Manufacturing System," *IEEE Trans. Autom. Sci. Eng.*, vol. 9, no. 1, pp. 31–41, Jan. 2012.

[7] M. Ito and J. Tani, "On-line Imitative Interaction with a Humanoid Robot Using a Dynamic Neural Network Model of a Mirror System," *Adapt. Behav.*, vol. 12, no. 2, pp. 93–115, Jun. 2004.

[8] D. Lee, C. Ott, and Y. Nakamura, "Mimetic Communication Model with Compliant Physical Contact in Human-Humanoid Interaction," *Int. J. Robot. Res.*, vol. 29, no. 13, pp. 1684–1704, 2010.

[9] W. Takano, K. Yamane, T. Sugihara, K. Yamamoto, and Y. Nakamura, "Primitive communication based on motion recognition and generation with hierarchical mimesis model," in *IEEE Int. Conf. on Robotics and Automation*, 2006, pp. 3602–3609.

[10] H. B. Amor, D. Vogt, M. Ewerton, E. Berger, B. Jung, and J. Peters, "Learning responsive robot behavior by imitation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2013, pp. 3257–3264.

[11] M. Prada, A. Remazeilles, A. Koene, and S. Endo, "Dynamic Movement Primitives for Human-Robot interaction: Comparison with human behavioral observation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2013, pp. 1168–1175.

[12] M. Prada, A. Remazeilles, A. Koene, and S. Endo, "Implementation and experimental validation of Dynamic Movement Primitives for object handover," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2014, pp. 2146–2153.

[13] A. Koene, S. Endo, A. Remazeilles, M. Prada, and A. M. Wing, "Experimental testing of the CogLaboration prototype system for fluent Human-Robot object handover interactions," in *IEEE Int. Symp. on Robot and Human Interactive Communication*, 2014, pp. 249–254.

[14] D. Vogt, S. Stepputtis, S. Grehl, B. Jung, and H. B. Amor, "A system for learning continuous human-robot interactions from human-human demonstrations," in *IEEE Int. Conf. on Robotics and Automation*, 2017, pp. 2882–2889.

[15] S. Calinon, "A tutorial on task-parameterized movement learning and retrieval," *Intell. Serv. Robot.*, vol. 9, no. 1, pp. 1–29, Jan. 2016.

[16] J. Yang, Y. Xu, and C. S. Chen, "Human action learning via hidden Markov model," *IEEE Trans. Syst. Man Cybern. - Part Syst. Hum.*, vol. 27, no. 1, pp. 34–44, Jan. 1997.

[17] B. Akgun and A. Thomaz, "Simultaneously learning actions and goals from demonstration," *Auton. Robots*, vol. 40, no. 2, pp.211-227, 2016.

[18] L. Rozo, P. Jiménez, and C. Torras, "A robot learning from demonstration framework to perform force-based manipulation tasks," *Intell. Serv. Robot.*, vol. 6, no. 1, pp. 33–51, Jan. 2013.

[19] Z. Wang *et al.*, "Probabilistic movement modeling for intention inference in human–robot interaction," *Int. J. Robot. Res.*, vol. 32, no. 7, pp. 841–858, Jun. 2013.

[20] G. J. Maeda, G. Neumann, M. Ewerton, R. Lioutikov, O. Kroemer, and J. Peters, "Probabilistic movement primitives for coordination of multiple human–robot collaborative tasks," *Auton. Robots*, vol. 41, no. 3, pp. 593–612, Mar. 2017.

[21] J. A. Bilmes and others, "A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," *Int. Comput. Sci. Inst.*, vol. 4, no. 510, p. 126, 1998.

[22] H. Bourlard, S. Dupont, H. Hermansky, and N. Morgan, "Towards subband-based speech recognition," in *European Signal Processing Conf.*, 1996, pp. 1–4.

[23] H. Hennansky, S. Tibrewala, and M. Pavel, "Towards ASR on partially corrupted speech," in *Int. Conf. on Spoken Language Processing*, 1996, vol. 1, pp. 462–465 vol.1.

[24] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[25] *OpenNI*. OpenNI organization.

[26] *NITE*. PrimeSense Inc., 2010.

[27] R. Stewart, *TensorBox*. 2017.