Robot Learning from Demonstration of Force-based Tasks with Multiple Solution Trajectories

Leonel Rozo, Pablo Jiménez and Carme Torras

Abstract— A learning framework with a bidirectional communication channel is proposed, where a human performs several demonstrations of a task using a haptic device (providing him/her with force-torque feedback) while a robot captures these executions using only its force-based perceptive system. Our work departs from the usual approaches to learning by demonstration in that the robot has to execute the task *blindly*, relying only on force-torque perceptions, and, more essential, we address *goal-driven* manipulation tasks with multiple solution trajectories, whereas most works tackle tasks that can be learned by just finding a generalization at the trajectory level. To cope with these multiple-solution tasks, in our framework demonstrations are represented by means of a Hidden Markov Model (HMM) and the robot reproduction of the task is performed using a modified version of Gaussian Mixture Regression that incorporates temporal information (GMRa) through the forward variable of the HMM. Also, we exploit the haptic device as a teaching and communication tool in a human-robot interaction context, as an alternative to kinesthetic-based teaching systems. Results show that the robot is able to learn a container-emptying task relying only on force-based perceptions and to achieve the goal from several non-trained initial conditions.

Fig. 1. Learning and reproduction phases

I. INTRODUCTION

During the last two decades robotics science has focused on creating robots able to carry out complex tasks in human environments under very dynamic conditions [1]. For achieving this ambitious goal, it is necessary to provide robots with learning capabilities allowing them to acquire new knowledge, to correct or refine their actions and to reason about their perception-action relationships [2]. Learning from demonstration (**LbD**) (also known as robot programming by demonstration or imitation learning) stands out as a promising way of providing a robot with learning skills. Such framework is based on an interactive student-teacher structure that allows humans to teach robots as they would teach other humans, thus exploiting skills that humans already possess [3], [4].

Several learning settings have been proposed that may be classified according to either the perception system used (e.g. vision or motionsensors) or the level at which the task is learned (i.e. at symbolic or trajectory level). Researchers work at the symbolic level when it is necessary to learn a very complex task that can be carried out through a
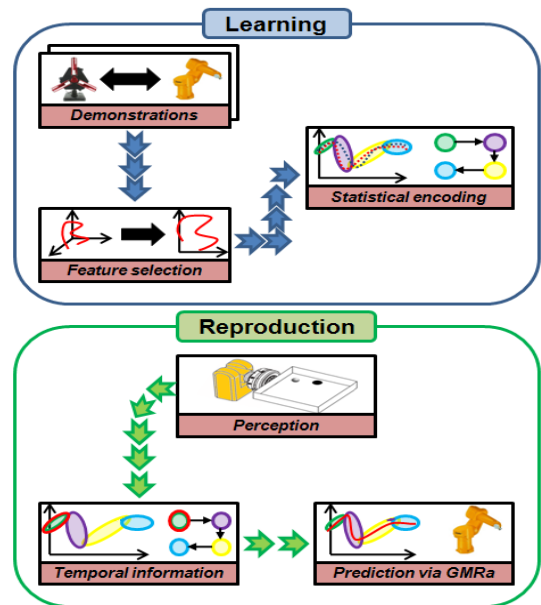
sequence of actions-states, where a state can represent the accomplishment of a given previous subgoal [5], [6]. Another research branch deals with learning at trajectory level where the main idea is to extract relevant information from given demonstrations (movements) with the aim of reproducing a generalized motion under new conditions [7], [8]. These two levels complement each other for creating structures able to learn very complex tasks. More specifically, learning at trajectory level can be used to encode task primitives that constitute the basis for a framework working at symbolic level [3].

On the other hand, most of the state of the art has used vision as perception system arguing that this is the most natural and effective way in which humans capture information from demonstrations given by an expert [9], [10], [11], while other works have used motion sensors attached to the teacher's body [12], [13]. Although vision and/or motion sensors provide information which is essential and enough for many tasks, there are also whole families of scenarios where these data are clearly insufficient, like opening doors, pulling drawers, cutting slices of bread, or emptying deformable bags. Such type of tasks demands to use other information sources allowing to get more knowledge about the task, like force/torque data.

Forces and torques can be considered either as a com-

plementary information for speeding up the learning stage by providing additional information missed by vision or motion sensors, or as main data for being able to learn a task successfully. Based on the former considerations, recent works have focused their efforts on exploiting this new data source in LbD. [14] proposed a learning framework based on Gaussian Mixture Model (**GMM**) and Gaussian Mixture Regression (**GMR**) for endowing a humanoid robot with the ability to perform a collaborative manipulation task with a human operator using a haptic device and working at trajectory level (just a simple vertical movement was learned). An extension of this research [15], combines LbD and adaptive control for teaching the task, which endows the robot with variable inertia and an adaptive algorithm to generate different reference kinematic profiles depending on the perceived force. Finally, other recent work [16] presents a framework that allows a robotic manipulator to learn to perform tasks which require exerting forces on external objects by interacting with a human operator. We contribute with a complete F/T data-based learning framework that includes filtering processes, high fidelity haptic feedback, and handling of perceptual aliases (multi-valued function). This structure can be used for applications where F/T information is necessary to learn/carry out a task, like opening a door with a key, where F/T patterns corresponding to unlocking effects have to be felt. Our task is characterized by similar features, as the fact of being based on F/T information, and the need of using temporal coherence to discriminate between different states associated to the same perceptions.

In the proposed learning framework, a human shows several demonstrations of the task using a haptic device (providing him/her force-torque feedback) while a robot observes these executions just using its force-based perceptive system (refer to Fig. 1). The learning process of the task is not based merely on finding a generalization at trajectory level, since the task can be carried out by executing different trajectories that accomplish the goal in a successful way. Thus, demonstrations are represented by a HMM through a sequence of states, and the reproduction stage is implemented through a modified version of GMR that uses temporal information of the task given by the forward variable of the HMM [17] that relies on transitions between states.

The obtained results show that this encapsulated temporal information is relevant for learning multi-solution tasks based on force data and without including time as an input variable, which improves approaches followed by [18] and [7], respectively. Moreover, the proposed framework shows how AI learning techniques embedded in a human-robot interaction setting can rapidly increase the manipulation competences of a robot beyond teleoperation, thus widening their range of application.

The remainder of this paper is organized as follows: Section II describes the experimental setting and the task to be learned, while Sections III and IV explain how demonstrations are codified by a HMM and how the robot reproduces the task using the modified version of GMR. The obtained results are discussed in Section V and finally Section VI is
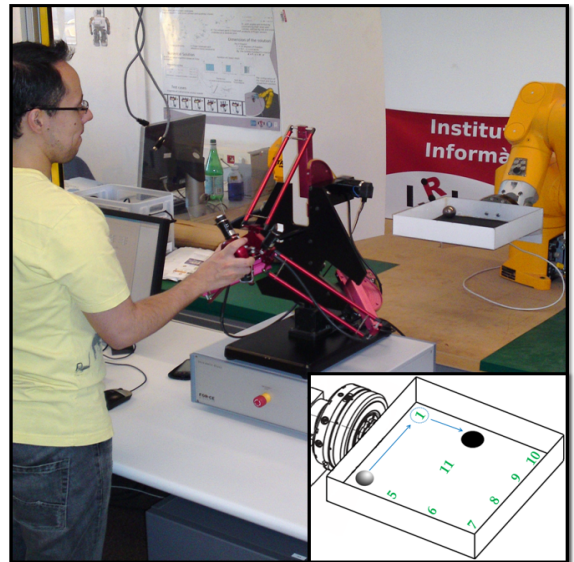

Fig. 2. Experimental setting and initial positions of the ball for the training phase (*at the bottom right corner*)

devoted to conclusions and future work.

II. EXPERIMENTAL SETTING

One of the main goals of this work is to show how it is possible to learn a given task based on force/torque data successfully. For achieving this objective we have proposed an experimental setting with a bidirectional communication channel between the human teacher and the robotic student constructed from a teleoperated robotic arm through a 6-DOF haptic device which allows to reflect forces/torques (**F/T**) on the user's hand (see Figure 2). In our experimental setup, we aim at teaching a robot to extract a metallic ball from inside a box-like container, which has a hole on its base. The robot has to orient the box in such a way that the ball is forced to roll towards the hole. The task is exclusively based on force/torque feedback, which means that the robot does only sense the forces and torques exerted on its wrist by the box and the ball, which change with the orientation of the box and the position of the ball. The teacher has an additional source of information by watching the scene. In addition, the user holds the end-effector of the haptic device which allows him to feel the forces and torques sensed at the robotic wrist.

Despite the task to be learned seems simple at first sight, the entire process implies to solve several technical and research issues. Regarding to the acquisition of suitable training data from teacher demonstrations, first it is necessary to take into account that the box is not a rigid structure, it vibrates when the robot moves. Here, the solution is to implement a digital low-pass filter to reduce the effects on the sensor readings caused by vibrations. Second, it is important that the teacher can feel F/T generated in the box as real as possible without distracting him/her from task executions. For solving this, we propose to feedback only the ball's dynamics inside the structure without reflecting F/T generated

by the box's mass, this is achieved by compensating this mass dynamically while executions are realized.

On the other hand, once the teacher has carried out several different samples of the task, it is necessary to determine which learning framework is appropriate for this kind of settings. As it is shown in Figure 1, all demonstrations composing the training data are analyzed with the aim of selecting the most relevant variables of the task at the same time as the *What to Imitate?* problem is solved based on the approach given in [18] where Mutual Information(MI)-based feature selection was applied. After that, demonstrations are encoded statistically by a HMM which also captures their temporal information (more details in Section III). The reproduction phase is carried out once the HMM has been trained, where the first step is to acquire a suitable perception, which means to filter and compensate sensor readings. Next, the resulting data are provided to the trained HMM as an observation of the system with the aim of finding the weights of the GMR, which rely on the temporal information contained in the HMM and used in the prediction step through GMR (details in Section IV).

Although the teacher can watch the scene directly, we propose to get the human more involved in the teaching process by making the robots perceptions during demonstrations available to him/her. This bidirectional communication channel has been very little exploited as a human-robot interaction tool in LbD in contrast to kinesthetic-based teaching. Filtering processes and dynamic compensation are necessary, as confirmed by testing our framework with several people. They were fed back with: a) box and ball dynamics or b) just ball dynamics. Every participant argued that the presence of the container's mass while demonstrating the strategy was a very distracting factor making the task harder.

III. ROBOT LEARNING BASED ON HIDDEN MARKOV MODELS

The teacher's demonstrations given as training data start with the ball placed at different positions inside the box, which means that the goal can be achieved from several initial conditions relying on teacher executions[1]. This implies that the learning framework's goal is not to learn merely a trajectory [14] or a task with predefined states as in assembly processes [19] that can be represented at a symbolic level. For endowing the robot with a suitable learning structure for this kind of tasks and avoiding to assume some aspects about the task to be learned, we propose to use a HMM to encode the teacher demonstrations using an ergodic topology, similar to the approach followed in [17].

Given our experimental setting described in Section II and following the notation of [20], let us denote a training datapoint as $\mathbf{d}_p^m \in \Re^D$, with $m = 1, 2, \ldots, M$ and $p = 1, 2, \ldots, P$, where M is the number of demonstrations, P is the number of datapoints collected along demonstration m, and D is the total number of input and output variables.

[1]Demonstrations were carried out executing a predefined motion strategy that consisted in taking the ball to the wall adjacent to the hole, and then rolling the ball along this wall to the hole.

For our current task, inputs correspond to F/T sensed at the robotic wrist and outputs are the velocity commands ω_l at each robot joint q_l with $l = 1, \ldots, 6$. However, thanks to the MI process we concluded that just torques along x and y axes (i.e. T_x and T_y) are necessary as inputs to learn the task successfully because these describe the position of the ball inside the box entirely. Thus, each training datapoint is defined as $\mathbf{d}_p^m = (T_x, T_y, \omega_1, \omega_2, \ldots, \omega_6)$.

With all demonstrations we can encode the joint distribution $P(\mathbf{T}, \omega)$ through an ergodic HMM defined as $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$, where:

- $\mathbf{A} = \{a_{ij}\}$ is the state transition probability matrix, with $1 \leq i, j \leq N$.
- $\mathbf{B} = \{b_j(k)\}$ is the observation symbol probability matrix, with $1 \leq k \leq (M*P)$ and assuming continuous observation densities defined as normal distributions $\eta(O; \mu_j, \Sigma_j)$.
- $\pi = \{\pi_i\}$ is the initial state probability vector, with $1 \leq i \leq N$.
- N is the number of states S_i of the model.

The main idea is to adjust the model λ to maximize $P(O|\lambda)$ where O is an observation sequence $O = O_1 O_2 \ldots O_T$ with each O_t corresponding to a training datapoint $\mathbf{d}_p^m$. To achieve this objective, an iterative procedure such as the *Baum-Welch* method is used (more details in [20]). In order to describe the procedure for re-estimation of HMM parameters, it is necessary to define the following variables:

$$\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)} \quad (1)$$

$$\gamma_t(i) = \sum_{j=1}^{N}\xi_t(i,j) \quad (2)$$

where α and β are called *forward* and *backward* variables, respectively, and defined as:

$$\alpha_1(i) = \pi_i b_i(O_1) \quad (3)$$

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N}\alpha_t(i)a_{ij}\right]b_j(O_{t+1}) \quad (4)$$

$$\beta_T(i) = 1 \quad (5)$$

$$\beta_t(i) = \sum_{j=1}^{N}a_{ij}b_j(O_{t+1})\beta_{t+1}(j) \quad (6)$$

From equations 1 and 2, it is possible to estimate the HMM parameters iteratively as follows:

$$\overline{\pi}_i = \gamma_1(i)$$

$$\overline{a}_{ij} = \frac{\sum_{t=1}^{T-1}\xi_t(i,j)}{\sum_{t=1}^{T-1}\gamma_t(i)}$$

$$\overline{\mu}_{jk} = \frac{\sum_{t=1}^{T}\gamma_t(j,k)O_t}{\sum_{t=1}^{T}\gamma_t(j,k)}$$

$$\overline{\sum}_{jk} = \frac{\sum_{t=1}^{T}\gamma_t(j,k)(O_t - \mu_{jk})(O_t - \mu_{jk})'}{\sum_{t=1}^{T}\gamma_t(j,k)}$$
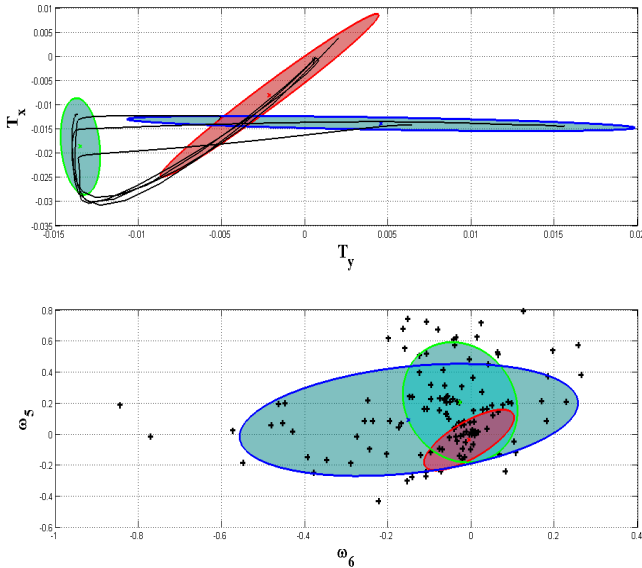
Fig. 3. Resulting 3-states HMM trained with demonstrations starting at positions {1,2,3,4}. *Top*: Input space composed of the most relevant inputs {T_x, T_y}. *Bottom*: Output space composed of robot joint velocities playing the most important role for the given task.

These equations permit obtaining a suitable trained HMM that represents the teacher demonstrations statistically through a states model capturing the velocity commands for given sensed torques and taking temporal coherence into account from the resulting matrix **A**. This may be better understood by observing Figure 3, where the resulting HMM is displayed after the training phase with demonstrations sharing similar trajectories. This model was initialized with an ergodic topology through the initial values in matrix **A**. However, the final values of **A** show that the *hidden* structure in training data is a *left-to-right* model, where the resulting vector π gives as initial state the blue Gaussian. It is also important to highlight that blue and red states intersect each other in input space, covering the same segments of trajectories. In this case, the temporal information is essential to determine which velocity command has to be provided, which is not clear using a GMM/GMR approach.

On the other hand, since the task is neither strictly learned as a sequence of discrete actions nor as a simple trajectory, it is necessary to find a suitable way to reconstruct the velocity commands, given a perception and the resulting trained HMM. To achieve this goal, a modified version of GMR (here named GMRa) is used for computing the velocity commands to be sent to the robot controller as desired joint velocities, as described next.

IV. REPRODUCTION OF THE TASK USING TEMPORAL INFORMATION

Once the learning stage has been carried out and the HMM has been trained, it is necessary to devise a suitable tool for synthesizing the desired velocity commands when the robot is executing the task under known and unknown conditions (e.g. non-trained initial positions of the ball inside the box). Using the statistical representation given by the resulting

HMM, a modified version of Gaussian Mixture Regression (GMRa) that uses the temporal information captured by the HMM [17] is implemented for obtaining the velocity commands to be sent to each robot joint. The idea of using this temporal information (from HMM variable α) is to predict the desired velocity command as a function of the given perception (i.e. torques sensed at the robotic wrist) and sequential information probabilistically encapsulated in the HMM.

Recent works propose to use a framework based on GMM/GMR for learning tasks at trajectory level, where the main idea is to model data from a mixture of Gaussians and compute predictions for a given set of queries through regression by applying the original version of GMR. In this approach, standard GMR averages the different observations, even if they have been observed at different parts of the skill. Formally, for each Gaussian component i, both input and output data are separated by expressing the mean and covariance matrix as:

$$\mu_i = \{\mu_{T,i}, \mu_{\omega,i}\} \quad , \quad \Sigma_i = \left(\begin{array}{cc} \Sigma_{TT,i} & \Sigma_{T\omega,i} \\ \Sigma_{\omega T,i} & \Sigma_{\omega\omega,i} \end{array} \right)$$

Then, the conditional expectation $\hat{\omega}$ given T, for a mixture of N Gaussians is:

$$\hat{\omega} = \sum_{i=1}^{N} \beta_i \left[\mu_{\omega,i} + \Sigma_{\omega T,i}(\Sigma_{TT,i})^{-1}(T - \mu_{T,i}) \right] \quad (7)$$

where $\beta_i = \frac{p(i)p(T|i)}{\sum_{l=1}^{N} p(l)p(T|l)}$ is a weight based on given input variables (i.e. force/torque data).

HMM/GMRa can provide a better estimation by using a weight that takes into consideration both F/T and sequential information instead of the GMM/GMR weights, that would take only F/T into account. Now, the process of estimating the weights is not only based on the actual values of the inputs, but takes also into account the previous values of them, which is done implicitly through transition probabilities using the HMM forward variable. Thus, the new definition for GMR based on temporal information is given by:

$$\hat{\omega} = \sum_{i=1}^{N} \alpha(i) \left[\mu_{\omega,i} + \Sigma_{\omega T,i}(\Sigma_{TT,i})^{-1}(T - \mu_{T,i}) \right] \quad (8)$$

where $\alpha(i)$ is the forward variable for the i-th Gaussian[2] in the HMM. This variable expresses the probability of observing the partial sequence, $O = O_1 O_2 \ldots O_t$ and of being in state S_i at time t. Now, for a given F/T perception, the predicted velocity command is based on current and past observations, which makes sense for those tasks where more than one output exists for a given input pattern[3].

---

[2]Assuming continuous observation densities defined as normal distributions, in other words, each HMM state is represented by a Gaussian.

[3]For our task, torques equal to zero mean that the ball is close to the sensor's frame origin or that the ball is out from the box. In the first case, the robot has to orient the box in such a way the ball moves closer to the goal, while the robot does not have to execute any movement in the second case.
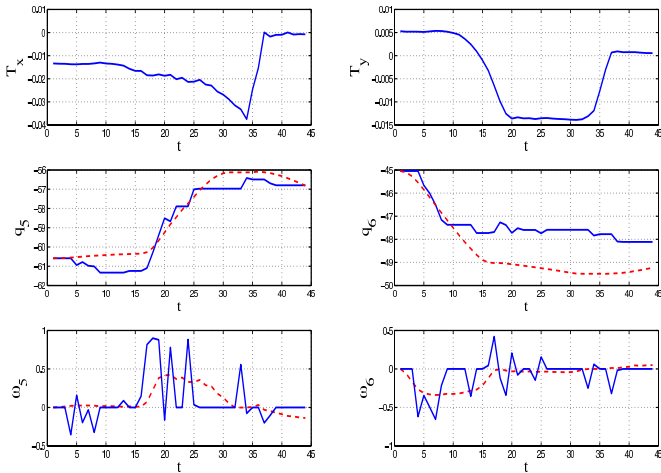
Fig. 4. *Top*: Input torques $\{T_x, T_y\}$ during a demonstration starting from position number 3. *Middle*: Robot joint trajectories corresponding to the teacher's demonstration and the robot's execution using the HMM/GMRa approach. *Bottom*: Robot joint velocity profiles obtained from teacher and robot executions, these later computed through GMRa
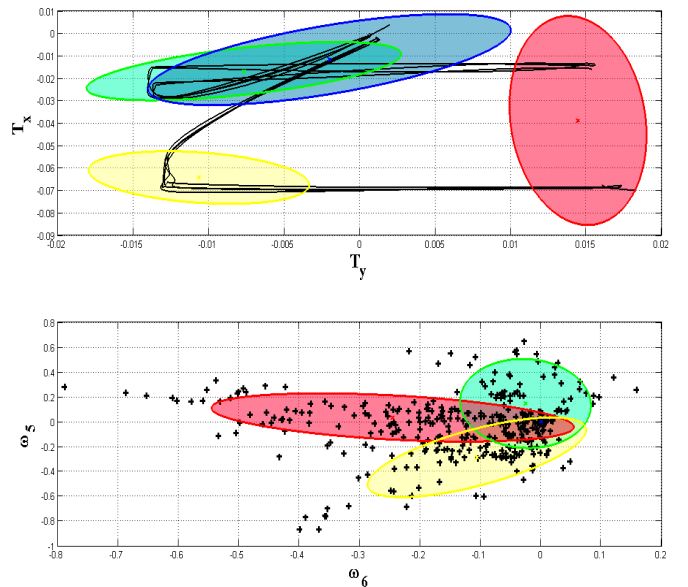

Fig. 5. Resulting 4-states HMM trained with demonstrations starting at every position inside the box. *Top*: Input space composed of the most relevant inputs $\{T_x, T_y\}$. *Bottom*: Output space composed of robot joint velocities playing the most important role for the given task.

We used the HMM/GMRa approach proposed by Calinon et al. because this allows us to encode the demonstrations taking the implicit temporal information into account. Our work differs from theirs and other approaches in: a) only F/T are considered, b) we do not use time as an additional input variable, c) our task behaves as a multi-valued function where taking temporal coherence does really make sense, and d) our task can be solved by executing two different sets of movements. For instance, in Figure 4, the robot joint trajectories for q_5 and q_6 are shown. These trajectories are obtained from the velocity commands predicted through GMRa using the resulting HMM shown in Figure 3. Here, it is possible to observe how the robot's execution is similar to the teacher's one for a given input pattern. Moreover, the learning framework performs successfully when input data lie simultaneously on two HMM states, because it uses the temporal information encapsulated in variable α_k for deciding in which state the system is in.

V. RESULTS

For evaluating the performance of the learning framework proposed, the teacher carried out three demonstrations for ten initial positions placed along the box edges. Every demonstration was executed by teleoperating the robotic arm through the 6-DOF haptic device (as shown in Figure 2) and following the motion strategy explained in Section IV. From these demonstrations, the resulting training dataset consisted of all datapoints $\mathbf{d}_p^m = (T_x, T_y, \omega_1, \omega_2, \ldots, \omega_6)$, with which a 4-states HMM was trained by applying the *Baum-Welch* method until convergence (Figure 5 shows the resulting HMM).

Observing the HMM obtained, it is interesting to highlight how HMM states are distributed in both input and output spaces. For instance, the red state in input space covers the beginning of all demonstrations whose initial positions are placed on the wall opposite to where the hole is. At these

starting positions, a larger velocity command is required to draw the ball out of its resting configuration (Figure 5, bottom). On the other hand, the blue state covers the trajectory segments corresponding to the end of the task (i.e. when the ball is getting out of the box) in input space. For these torque data, the robot should not carry out any movement (small blue ellipse at (0,0) in output space).

As for the prediction phase, one teacher's demonstration for each initial position was removed from the training examples and used as "query data" for evaluating the learning framework performance by comparing its results with teacher executions. All robot joint trajectories obtained from velocity commands synthesized by our HMM/GMRa approach are smoother than the teacher's samples (as shown for initial position 3 in Figure 4). In addition, all synthesized trajectories follow the same motion pattern as that of the teacher's executions, which indicates that the strategy applied by the human user was learned successfully.

Finally, once computational results were sufficiently good, the prediction phase was validated experimentally. In a first set of experiments, the ball was placed at the same initial positions shown in Figure 2. For all different positions of the ball, the robot was able to take the ball out of the box. Afterwards we placed the ball at random positions, and the robot executions achieved the final goal as well. In these cases, the robot executes the motions learned for the closest initial position, by identifying the corresponding HMM state. It was observed that in some executions the ball reached and surpassed the hole, without falling through it. However, the robot was always able to take the ball out of the box after some more executions, as it correctly identified the HMM state corresponding to the current and past input patterns (taking into account the temporal information). This

means that the robot predicts its actions as a function of its current and past perceptions, following the teached motion strategy. If the robot fails to reach the goal, the ball goes to another position inside the box, providing new perceptions from which the robot can predict new movements [21].

VI. CONCLUSIONS AND FUTURE WORK

This work presents a framework suitable to learn tasks where force/torque data are essential. AI techniques to encode perception-action sequences (HMM) and to generalize them to deal with new situations (GMRa) were embedded in a learning-by-demonstration setting allowing a robot to accomplish a taught task from a variety of initial conditions. Unlike common approaches (i.e. symbolic or trajectory learning), our work proposes a more general structure able to encode force-based tasks with multiple solution trajectories. Moreover, our framework performs efficiently when the teacher's demonstrations exhibit a multi-valued function behavior, which means there may be more than one action – velocity command – for the same perception (i.e. force/torque input pattern). This was achieved by means of a GMRa tool using temporal information encapsulated by a HMM without considering time as another input variable, which can be viewed as redundant information already present implicitly along the teacher's demonstrations.

In the past we had addressed our task as learning at trajectory level using GMM/GMR, F/T as inputs and robot joint values as outputs [18]. We measured the robot performance through the MSE between the desired trajectory and the predicted one. When executing the predictions in the real setting, we realized that failures could arise when the same F/T pattern corresponds to two (or more) possible outputs, and therefore, temporal coherence should be taken into account. This is why we changed from GMM/GMR to HMM/GMRa. Low MSEs do not guarantee successful completion of the task because the relevant issue is not to follow exactly a trained trajectory but to orient the container in the correct direction. Thus we changed outputs from joint values to velocity commands. Moreover, we evaluated the robot success directly in the execution phase as described.

Note that both computational and robot execution results yielded smooth trajectories, that eventually took the ball out of the box even if the robot missed the hole at previous attempts. This was possible because our learning framework identifies which state the ball is in, and then predicts a suitable velocity command by taking into account the current state and previous ones through the learned transition probabilities. This allows the robot to try to solve the task by changing the strategy (i.e., with different movements or trajectories) if it failed before, which is a natural behavior observed in humans trying to achieve a given goal.

As future work, we plan to take the learning framework to more realistic settings where force/torque feedback is relevant: opening doors, pulling drawers or emptying deformable bags. Moreover, we would like to apply force-based skill learning to compliant robots in an active learning environment as a refinement or correction phase. In addition, this type of robots would allow us to extend our approach to human-robot collaborative tasks by taking advantage of their compliance features, from a impedance control-based perspective.

REFERENCES

[1] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, *Springer Handbook of Robotics*, 2008, ch. 59. Robot Programming by Demonstration, pp. 1371–1394.

[2] B. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning by demonstration," *RAS*, no. 57, pp. 469–483, 2009.

[3] S. Schaal, A. Ijspeert, and A. Billard, *The Neuroscience of Social Interaction*. Oxford University Press, 2004, no. 1431, ch. Computational Approaches to Motor Learning by Imitation, pp. 199–218.

[4] M. Lopes, F. Melo, L. Montesano, and J. Santos-Victor, *From Motor to Interaction Learning in Robots*. Springer, 2010, ch. Abstraction Levels for Robotic Imitation: Overview and computational approaches, pp. 313–355.

[5] D. Bentivegna, C. Atkeson, and G. Cheng, "Learning tasks from observation and practice," *RAS*, vol. 47, no. 2-3, pp. 163–169, 2004.

[6] D. Grollman and O. Jenkins, *From Motor to Interaction Learning in Robots*. Springer, 2010, ch. Can We Learn Finite State Machine Robot Controllers from Interactive Demonstration?, pp. 407–430.

[7] S. Calinon and A. Billard, *Imitation and Social Learning in Robots, Humans and Animals*. Cambridge, 2007, ch. Learning of Gestures by Imitation in a Humanoid Robot, pp. 153–177.

[8] M. Schneider and W. Ertel, "Robot learning by demonstration with local gaussian process regression," in *IROS*, 2010, pp. 255–260.

[9] R. Chalodhorn, D. Grimes, K. Grochow, and R. Rao, "Learning to walk through imitation," in *IJCAI*, 2007, pp. 2084–2090.

[10] D. Bentivegna, A. Ude, C. Atkeson, and G. Cheng, "Humanoid robot learning and game playing using pc-based vision," in *IROS*, 2002, pp. 2449–2454.

[11] D. Grollman and O. Jenkins, "Dogged learning for robots," in *ICRA*, 2007, pp. 2483–2488.

[12] T. Inamura, N. Kojo, T. Sonoda, K. Sakamoto, and K. Okada, "Intent imitation using wearable motion capturing system with on-line teaching of task attention," in *Humanoids*, 2005, pp. 469–474.

[13] S. Calinon and A. Billard, "Incremental learning of gestures by imitation in a humanoid robot," in *Humanoids*, 2007, pp. 255–262.

[14] P. Evrard, E. Gribovskaya, S. Calinon, A. Billard, and A. Khedda, "Teaching physical collaborative tasks: Object-lifting case study with a humanoid," in *Humanoids*, 2009, pp. 399–404.

[15] E. Gribovskaya, A. Kheddar, and A. Billard, "Motion learning and adaptive impedance for robot control during physical interaction with humans," in *ICRA*, 2011.

[16] P. Kormushev, S. Calinon, and D. Caldwell, "Imitation learning of positional and force skills demonstrated via kinesthetic teaching and haptic input," *Advanced Robotics*, vol. 25, no. 5, pp. 581–603, 2011.

[17] S. Calinon, F. D'halluin, E. Sauser, D. Caldwell, and A. Billard, "Learning and reproduction of gestures by imitation," *IEEE Robotics and Automation Magazine*, vol. 17, no. 2, pp. 44–54, 2010.

[18] L. Rozo, P. Jiménez, and C. Torras, "Sharpening haptic inputs for teaching a manipulation skill to a robot," in *IEEE Intl. Conf. on Applied Bionics and Biomechanics*, 2010.

[19] S. Dong and F. Naghdy, "Application of hidden markov model to acquisition of manipulation skills from haptic rendered virtual environment," *Robotics and Computer-Integrated Manufacturing*, pp. 351–360, 2007.

[20] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," in *Proceedings of the IEEE*, 1989, pp. 257–286.

[21] Videos accompanying this paper. Available in: http://www.iri.upc.edu/people/lrozo/Publications/ICAR11.html.