Towards Generalizable Associative Skill Memories

Hakan Girgin and Emre Ugur

***Abstract*— Associative Skill Memories (ASMs) were formulated to encode stereotypical movements along with their stereotypical sensory events to increase robustness of underlying dynamic movement primitives (DMPs) against noisy perception and perturbations. In ASMs, the stored sensory trajectories, such as the haptic and tactile measurements, are used to compute how much a perturbed movement deviates from the desired one, and to correct the movement if possible. In our work, we extend ASMs: rather than using stored single sensory trajectory instances, our system generates sensory event models, and exploits those models to correct the perturbed movements during executions with the aim of generalizing to novel configurations. In particular, measured force and the torque trajectories are modeled using Hidden Markov Models, and then reproduced by Gaussian Mixture Regression. With Baxter robot, we demonstrate that our proposed force feedback model can be used to correct a non-linear trajectory while pushing an object, which otherwise slips away from the gripper because of noise. At the end, we discuss how far this skill can be generalized using the force model and possible future improvements.**

I. Introduction

Learning from Demonstration (LfD) [1] has been suggested as an efficient and intuitive way to teach new skills to the robots, where the robot observes, learns and imitates the actions demonstrated by the human tutors. LfD has been applied to various robotic learning problems including object grasping and manipulation [2]–[6]. Among others, learning methods that are based on dynamic systems [7] and statistical modeling have been popular in the recent years.

Dynamic Movement Primitives (DMPs) [7], for example, encode the demonstrated trajectory as a set of differential equations, and offers advantages such as one-shot learning of non-linear movements, real-time stability and robustness under perturbations with guarantees in reaching the goal state, generalization of the movement for different goals, and linear combination of parameters. The parameters of the system can be learned with different advanced algorithms such as Locally Weighted Regression [8] and Locally Weighted Projection Regression [9]. Statistical modeling, which can model the statistical regularities and important features of the demonstrated motions, has also been influential in learning the skills [2], [10].

After encoding the action, the robot is generally required to refine the parameters of the learned control policy [11].

This research has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 731761, IMAGINE; and was partially supported by TUBITAK BIDEB under agreement No 117C016 and by Bogazici Resarch Fund (BAP) Startup project no 12022.

Bogazici University, Department of Computer Engineering, Istanbul, Turkey firstname.lastname@boun.edu.tr

Fig. 1: Baxter robot pushing an object on table along a curved trajectory to a goal position.

Memorized force and tactile profiles can also be used to modulate learned Dynamic Movement Primitives (DMPs) [12], [13]. Memorized force and tactile profiles have already been successfully utilized in modulating learned movement primitives in difficult manipulation tasks that contain high degrees of noise in perception such as flipping a box using chopsticks However, we believe that rather than memorizing one single haptic profile for a skill, learning general multi-model sensory models might provide us with more generalizable and robust manipulation skills.

Recently, Chu et al. learned such multi-modal models based on Hidden Markov Models from temperature, pressure and fingertip information for exploratory object classification tasks [14], however the learned models were not used to adapt any further action execution. Latent Drichlet Allocation [15] and recently deep networks [16] were used to learn multi-modal models from different sensory information such as temperature, pressure, fingertip, contacts, proprioception, and speech; however these models were used only to categorize the sensory data without any effect on action execution.

In this paper, our system generates sensory event models, and exploits those models to correct the perturbed movements during executions with the aim of generalizing to the novel configurations. In particular, measured force and the torque trajectories are modeled using Hidden Markov Models, and then reproduced by Gaussian Mixture Regression. With Baxter robot, we demonstrate that our proposed force feedback model can be used to correct a non-linear trajectory while pushing an object, which otherwise slips away from the gripper because of noise.

II. Methods

The formulation of DMP allows the robot to learn a stereotypical skill from demonstration. Adding a sensory feedback to the system enhances the capabilities of the robot in the learned skill by sending corrective signals to low-level controllers [4]. In the skill of pushing a cup, for example, the dynamics of the environment can not be always known to robot and the cup can slide from its end-effector from time to time. The forces that the robot should feel during the execution, namely the desired forces, help the robot the orient and position its end-effector so that it prevents sliding of the cup.

However, storing and using force trajectory instance does not allow generalization to new situations in the long run. Pastor et al. called this storage of data coupling with movements as Associative Skill Memories [12]. Here, instead of memorizing how to feel during each execution, we propose to model the forces that the robot senses for each primitive movement by using Hidden Markov Models and reproduce them using Gaussian Mixture Regression. For a typical movement, we argue that at least two force models obtained can be tied linearly by means of their hidden states, and that desired forces for a new movement at the proximity of these demonstrations can be calculated from interpolation.

A. Dynamic Movement Primitives

A one dimensional DMP is represented by the following set of equations

$$\tau\dot{v} = K(g-x) - Dv - K(g-x_0)s + Kf(s) \quad (1)$$

$$\tau\dot{x} = v \quad (2)$$

where x and $\dot{x}$ are the position and the velocity, whereas v and $\dot{v}$ are the corresponding velocity and acceleration of the system scaled by the duration of the demonstration τ. K is a spring constant and D is a damping term. $f(s)$ is a non-linear function of the phase variable s, which is defined as

$$f(s) = \frac{\sum_i w_i \psi_i(s) s}{\sum_i \psi_i(s)} \quad (3)$$

where $\psi_i = exp(-h_i(s-c_i)^2)$ are Gaussian basis functions whose centers and widths are c_i and h_i, respectively. The parameters w_i for each basis function are to be learned by linear regression to render the shape specific to the trajectory.

The phase variable makes DMP temporal invariant by encoding time in its canonical system defined as

$$\tau\dot{s} = -\alpha s \quad (4)$$

where α is a constant representing the convergence rate of the phase variable from 1 to 0. Starting each DMP with the same phase variable and integrating with the same canonical system ensures their simultaneous evaluation.

B. Sensory Feedback Extension to DMPs

In ASMs, a coupling term is integrated into the original DMP formulation Eq. (1) to compensate for the generalized forces that the robot senses during the execution of a task, since each movement primitive should capture the entire dynamics of the skill. This coupling term is given by

$$\zeta = \mathbf{K_1 J_{sensor}^T K_2}(\mathbf{F} - \mathbf{F_{des}}) \quad (5)$$

where $\mathbf{K_1}$ and $\mathbf{K_2}$ are positive definite gain matrices, $\mathbf{J_{sensor}^T}$ is the transpose of the Jacobian with respect to sensors by which the forces are measured. $\mathbf{F}$ and $\mathbf{F_{des}}$ are the current and desired generalized forces which, in task-space, is the end-effector's 6D wrench.

Coupling term incorporated in DMP formulation Eq. (1) is then given by

$$\tau\dot{v} = K(g-x) - Dv - K(g-x_0)s + Kf(s) + \zeta \quad (6)$$

C. Encoding Force Feedback by Hidden Markov Models

In this paper, we propose to construct temporal probabilistic models to encode force feedback trajectories measured from the same movement primitive that is executed several times. For this we propose to use Hidden Markov Models (HMM).

An HMM of N component has the following set of parameters

$$\theta = \{\pi_i, a_{ij}, \mu_\mathbf{i}, \mathbf{\Sigma_i}\}$$

with $i, j = 1, 2, ..., N$ representing the states. Here, π_i is the initial distribution of the state i and a_{ij} is the transition probability from state i to j. Each hidden state i is represented by a Gaussian distribution whose mean vector is $\mu_\mathbf{i}$ and covariance matrix is $\mathbf{\Sigma_i}$. Baum-Welch algorithm is used to determine these parameters.

D. Force Feedback Model Reproduction using Gaussian Mixture Regression

The force feedback coupling term in Eq. (5) requires prediction of desired force trajectory, i.e. F_{des}. We propose to use Gaussian Mixture Regression to reproduce this desired trajectory. For each Gaussian representing a hidden state i in HMM, the mean vector $\boldsymbol{\mu_i}$ and the covariance matrix $\mathbf{\Sigma_i}$ can be expressed in terms of input $\boldsymbol{x}$ and output $\boldsymbol{y}$ dimensions as

$$\boldsymbol{\mu_i} = \begin{bmatrix} \boldsymbol{\mu_i}^{\boldsymbol{x}} \\ \boldsymbol{\mu_i}^{\boldsymbol{y}} \end{bmatrix} \quad \boldsymbol{\Sigma_i} = \begin{bmatrix} \boldsymbol{\Sigma_i}^{\boldsymbol{xx}} & \boldsymbol{\Sigma_i}^{\boldsymbol{xy}} \\ \boldsymbol{\Sigma_i}^{\boldsymbol{yx}} & \boldsymbol{\Sigma_i}^{\boldsymbol{yy}} \end{bmatrix} \quad (7)$$

Then, given the input vector $\boldsymbol{x}$, the output vector $\boldsymbol{y}$ can be calculated from the following linear regression equation

$$\boldsymbol{y} = \sum_{i=1}^{N} h_i [\boldsymbol{\mu_i}^{\boldsymbol{y}} + \boldsymbol{\Sigma_i}^{\boldsymbol{yx}} (\boldsymbol{\Sigma_i}^{\boldsymbol{xx}})^{-1} (\boldsymbol{x} - \boldsymbol{\mu_i}^{\boldsymbol{x}})] \tag{8}$$

where h_i is a weight factor based on the input and given by

$$h_i = \frac{\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu_i}^{\boldsymbol{x}}, \boldsymbol{\Sigma_i}^{\boldsymbol{xx}})}{\sum_{i=1}^{N} \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu_i}^{\boldsymbol{x}}, \boldsymbol{\Sigma_i}^{\boldsymbol{xx}})} \tag{9}$$

where $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu_i}^{\boldsymbol{x}}, \boldsymbol{\Sigma_i}^{\boldsymbol{xx}})$ is the multivariate Gaussian distribution of the input.

E. Interpolation From Force Models

We would like to utilize the constructed force feedback models to predict the force feedback in novel situations as well. Here, we propose to use at least 2 different force feedback models - and combine these models based on a similarity metric that computes the distance between known situations and the novel one. One idea would be to tie up the means of Hidden Markov Model of forces based on this metric. However, the means of Hidden Markov Model are not temporally aligned. To this end, we decided simply to interpolate between two force models reproduced by Gaussian Mixture Regression. For further discussion on generalization, please see Section IV.

III. Experiments

A. Robot Platform

Our experimental setup is composed of a Baxter robot which has two 7-DoF anthropomorphic arm, each actuated by a series elastic actuators enabling to measure torque output directly from the actuators (see Figure 1). The arm has a electric, parallel jaw gripper that is used in closed state with 4 cm wide open during the experiments.

The experiments are conducted using a 5 kg box with $9.5cm$ height and $8x8cm^2$ surface area, placed on a flat table. The task of the robot is to push the box from an initial position to a final position with a curved trajectory, which is taught by kinesthetic teaching.

B. Task

We selected the task of "pushing an object to a goal position along a trajectory" task in the experiments as this task requires exploitation of learned force feedback model when the object is not moving as expected during the execution in response to the learned and reproduced movement of the end-effector. Such unexpected behavior can be observed through introducing different types of noise and perturbations: by incorrect perception of the exact location of the object and initiating the push trajectory from a slightly shifted position; or by physically perturbing the object while being pushed. In this paper, we simulated noise in perception, initiated the push trajectory from a slightly different position (around $5cm$), and called this setup as 'misplaced object'.

The task for the robot arm is to push the object following a curved trajectory. The movement is demonstrated by kinesthetic teaching, using the gravity compensating mode of the Baxter arm. Because holding the end effector while kinesthetic teaching affects the force/torque measurements, the recorded trajectory is re-executed without human intervention, and modelled with a set of DMPs in joint space. Working in joint space allowed us to use the force feedback coupling term (Eq. 5) by computing transpose of the Jacobian at each joint value and to show the validity of the transformation from end-effector 6D wrench to 7D joint torques by multiplying it with the transpose of the Jacobian. Note that the torques generated by self-motion (upto 2N in our case) of the end effector are not automatically compensated by the robot, therefore we applied an additional compensation step. In this step, the measurements taken during push action are averaged with a moving window of 30 ms, and subtracted from force measurements created by the same trajectory without any external interaction, i.e. pushed object.

After the trajectory is encoded as DMP as described above, the trajectory is executed in four different conditions, three times each. See Table I for details of these conditions.

TABLE I: Experimental conditions.

Condition	**Explanation**
Default-No-Force	object is in its default position initially, control with no force feedback coupling term
Default-With-Force	object is in its default position initially, control with force feedback coupling term
Misplaced-No-Force	object is misplaced, control with no force feedback coupling term
Misplaced-With-Force	object is misplaced, control with force feedback coupling term

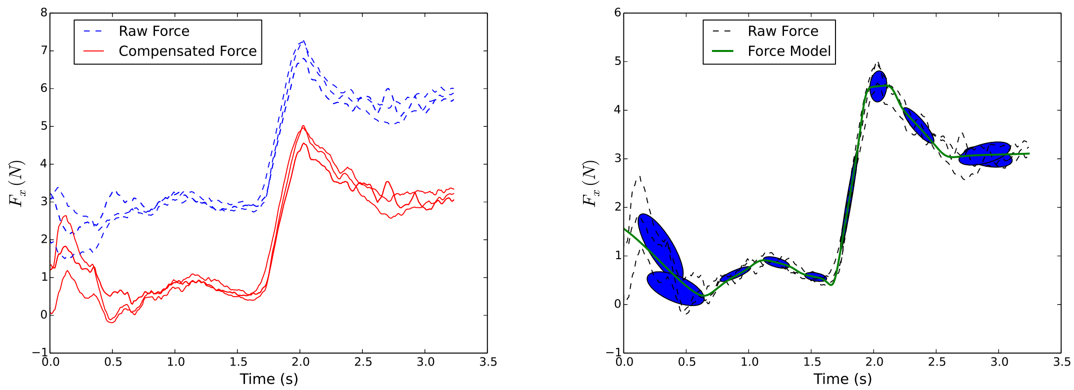
C. Force feedback model

Force/torque data obtained from the original end-effector trajectory was fitted into HMM model. Fig. 2a shows all the trajectories resulting from preprocessing of data described above. Since the task is to push a box on a table, we decided to neglect the effect of the forces orthogonal to the table, i.e. F_z, and the corresponding torques, T_x and T_y in the computations. Forces in x and y coordinates showed similar patterns, therefore we only showed the forces and position data in the x coordinate.

In our experiment, we set $\mathbf{K_1}$ as the diagonal matrix of 10's and $\mathbf{K_2}$ as identity matrix. Obtained force/torque and the time data are fitted in a Hidden Markov Model with 10 hidden states and with full covariance matrices. Taking time as input and force/torque as output, Gaussian Mixture Regression created a model from these 3 demonstrations. As shown in Fig. 2b, the resulting force feedback trajectory encodes forces taking into account the variation in the data. This feedback force model is set as F_{des} in Eq. (5).

D. Execution in default conditions

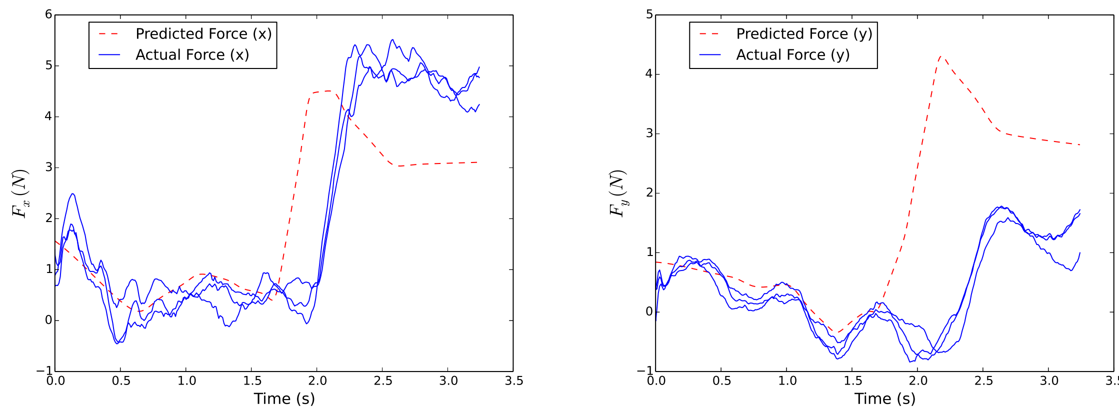
Due to inhomogeneities in the shape and the weight of the box (it includes 20 batteries inside), due to the noise in force readings, and additionally the slight differences in initial position of the box, executions even in *Default-With-Force* configuration show some variance. This causes discrepancies between the model and the original demonstration replication



(a) Force readings in default setup

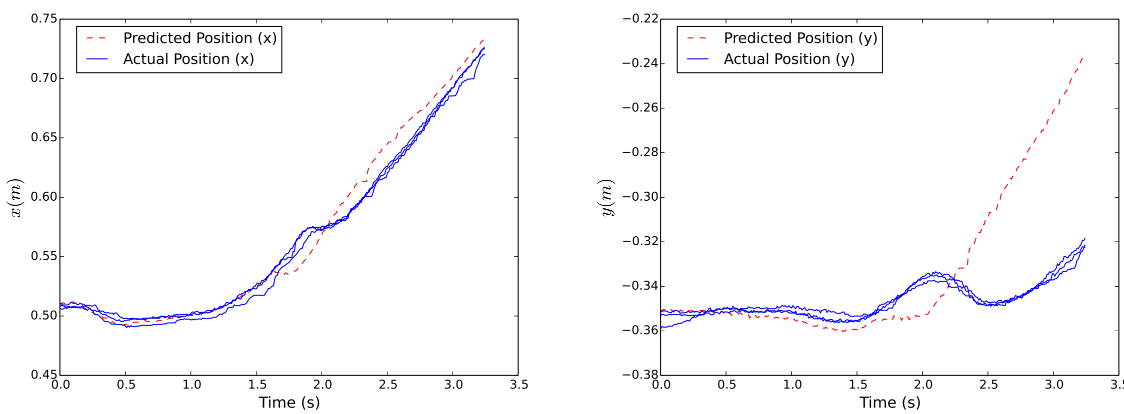
(b) HMM/GMR model

Fig. 2: (a) Raw and compensated force readings in x dimension. Three trajectories correspond to execution of the push action with object in default position initially. (b) The corresponding HMM model of the force trajectory with 10 hidden states. Mean and variances of hidden states represented by a univariate Gaussian distribution are shown by dark blue ellipses' center and radii.



(a) F_x

(b) F_y



(c) p_x

(d) p_y

Fig. 3: This figure shows how the actual force feedback deviates from the predicted one (at around $t = 1.7s$), and how the position of the gripper adapts in response to this change.
(a,b): Forces in *Misplaced-No-Force* configuration. Red dashed line represents GMR model of forces, namely the predicted force and the blue solid lines represent the actual force readings.
(c,d): Positions in *Misplaced-No-Force* configuration. Red dashed line represents the default predicted position and the blue solid lines represent the actual position readings.

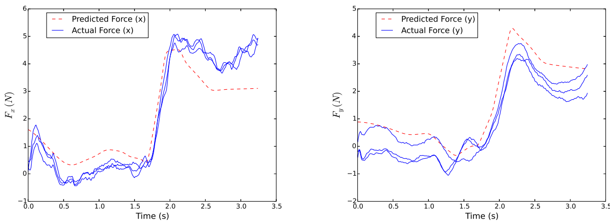


Fig. 4: Forces for robot execution with tactile feedback with object on its default position. Dashed lines correspond to the predicted force feedback obtained from GMR and solid lines correspond to the actual force feedbacks.

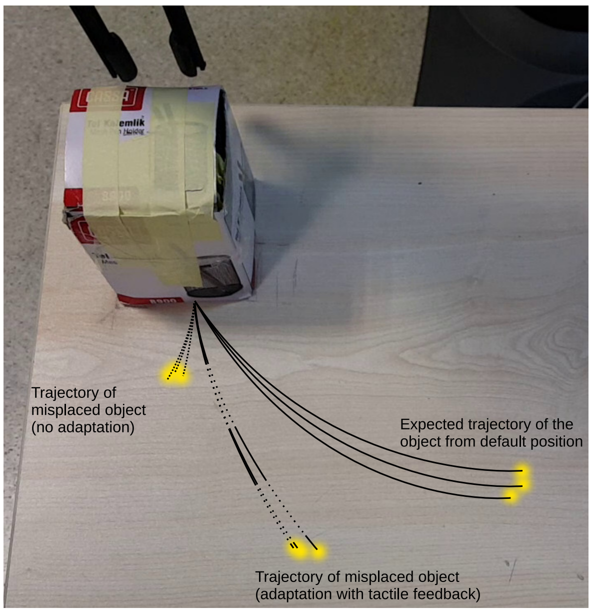


Fig. 5: The final positions of the object in different configurations, and the corresponding approximate trajectories. See link for the video : https://youtu.be/nxdP5QiyUY4

with tactile feedback as seen in Fig 4. Despite those, the end-effector follows the approximately the same x position and has little deviation from y position, therefore the object is pushed to the target position successfully (see Fig. 5).

E. Execution in noisy conditions

As described above, noisy conditions are generated by misplacing the object $5cm$ further away in the frontal direction from start position of the push trajectory. In all three executions, at around $t = 1.7s$, the actual force measured in x and in y directions starts deviating from the predicted force (see Fig. 3a and Fig. 3b). In response to this deviation, which is encoded in the force feedback component of the DMP, the movement of the gripper also starts deviating from the original demonstration trajectory. As seen in Fig. 3c and Fig. 3d, from $t = 1.7s$, while x position increases slightly following predicted trajectory, y position remains same for $0.4s$. Such behavior keeps the object in front of the gripper that starts pushing the object towards the goal position as desired. However, the effect of the initial perturbation cannot be compensated completely, as visible from the huge difference in both F_y and p_y (Fig. 3b and Fig. 3d), and the final position of the pushed object (Fig. 5). Still, without this force adaptation, the object slips away from the beginning as shown in Fig. 5.

F. Robot execution

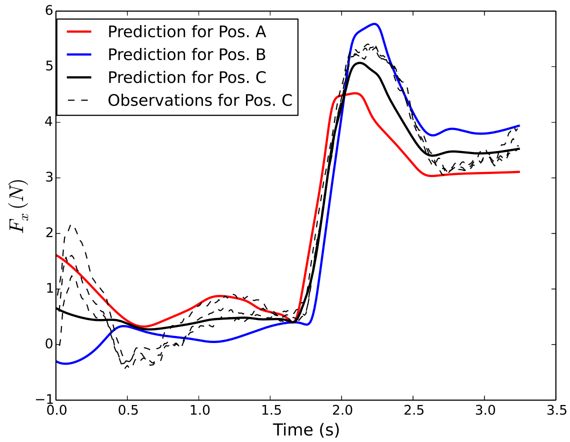
Figure 5 shows the trajectories of the object obtained in different configurations. When object is placed to its default position, the target object trajectory is observed with or without tactile feedback component. When object misplaced around 5 cm and without the tactile feedback component, the robot executes the taught trajectory and the object slips away in the beginning of this trajectory. When tactile feedback component is used and the movement is adapted based on this feedback, the object does not slip away from the beginning and the object is pushed towards the target position. As shown in this figure, the adaptation is not perfect, i.e. the object is not pushed to the target position exactly. We expect that through changing the effect of the tactile feedback component, i.e. by varying the magnitude of the coefficient K, the behavior of the system can be improved. Analyzing the reasons of this behavior and improving the trajectories with richer set of experiments are set as future work.

G. Generalization of the tactile feedback

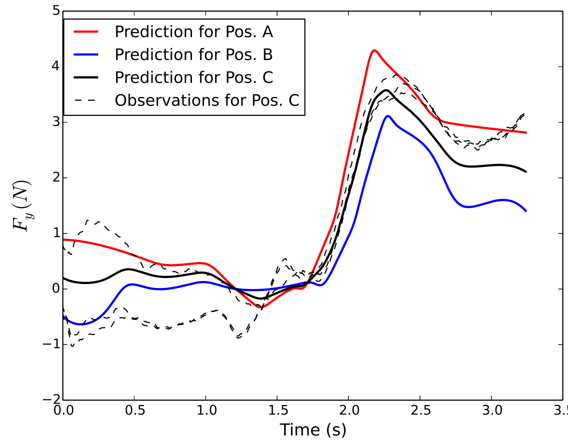
We developed our framework with the aim of generalizing the demonstrated actions to novel situations exploiting the learned sensory feedback models. While this aim is beyond the scope of the current workshop paper, in this section we will provide the initial results in this direction. For this, the learned DMP-based push action is executed with two different final positions. As the movements are performed on the table, and DMPs work in joint-space, we decided to set different wrist orientation angles for these two different final positions. Force feedback trajectory models for these two different final positions, denoted as Pos. A and Pos. B are constructed and shown by red and blue solid lines in Fig. 6. Then, force feedback model for a novel final position, Pos. C, is computed from those models, taking into account the pair-wise final position distances. The prediction of force feedback trajectory for this novel final position is plotted with solid black line in the same figure. Finally, to verify whether the predicted model, three more push actions are executed, this time with Pos. C, and the measured force feedback trajectories are shown with dashed black lines. As seen, generalized model gives a similar trajectory to the actual ones. While this holds roughly in this situation, we do not argue that such interpolation should always work. The dynamic relation between complex actions and interacted objects are difficult to model and the corresponding metrics can also be learned from further exploration [17].

IV. Conclusion

In this paper, we learned and exploited sensory event models to correct ongoing movements that are affected from noisy perception, with the future aim of generalizing learned



(a) F_x



(b) F_y

Fig. 6: Generalization of tactile feedback in (a) forces in x direction and (b) forces in y direction. Red line corresponds to forces when the wrist angle equals to 0.7 rad, Pos. A; blue line corresponds to forces when the wrist angle equals to 0.2 rad, Pos. B . Dashed lines represent forces of 3 demonstrations when the wrist angle equals to 0.45 rad, Pos. C. The green line corresponds to the GMR reproduction of these three demonstrations.

movements and sensory event models to the novel setups. Our system successfully exploited the learned force feedback models in order to adapt to noisy situations in a object pushing task with non-linear trajectory. We also showed that simple interpolation using learned force feedback models can be effective in predicting the force feedback in never experienced novel situations. However, we discussed that such interpolation idea would fail in complex settings. For this, in future, we plan to explore the idea of using Parametric HMMs in modeling sensory events [18].

REFERENCES

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

[2] S. Calinon, P. Evrard, E. Gribovskaya, A. Billard, and A. Kheddar, "Learning collaborative manipulation tasks by demonstration using a haptic interface," in *Advanced Robotics, 2009. ICAR 2009. International Conference on*. IEEE, 2009, pp. 1–6.

[3] T. Asfour, P. Azad, F. Gyarfas, and R. Dillmann, "Imitation learning of dual-arm manipulation tasks in humanoid robots," *International Journal of Humanoid Robotics*, vol. 5, no. 02, pp. 183–202, 2008.

[4] P. Pastor, L. Righetti, M. Kalakrishnan, and S. Schaal, "Online movement adaptation based on previous sensor experiences," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 365–371.

[5] H. Ben Amor, O. Kroemer, U. Hillenbrand, G. Neumann, and J. Peters, "Generalization of human grasping for multi-fingered robot hands," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 2043–2050.

[6] M. Mühlig, M. Gienger, and J. J. Steil, "Interactive imitation learning of object movement skills," *Autonomous Robots*, vol. 32, no. 2, pp. 97–114, 2012.

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

[8] C. G. Atkeson, A. W. Moore, and S. Schaal, "Locally weighted learning for control," in *Lazy learning*. Springer, 1997, pp. 75–113.

[9] S. Vijayakumar and S. Schaal, "Locally weighted projection regression: Incremental real time learning in high dimensional space," in *Proceedings of the Seventeenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc., 2000, pp. 1079–1086.

[10] S. Calinon, F. Guenter, and A. Billard, "On learning, representing, and generalizing a task in a humanoid robot," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 37, no. 2, pp. 286–298, 2007.

[11] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, p. 0278364913495721, 2013.

[12] P. Pastor, M. Kalakrishnan, L. Righetti, and S. Schaal, "Towards associative skill memories," in *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on*. IEEE, 2012, pp. 309–315.

[13] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, 2009, pp. 763–768.

[14] V. Chu, I. McMahon, L. Riano, C. G. McDonald, Q. He, J. Martinez Perez-Tejada, M. Arrigo, N. Fitter, J. C. Nappo, T. Darrell, *et al.*, "Using robotic exploratory procedures to learn the meaning of haptic adjectives," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3048–3055.

[15] T. Araki, T. Nakamura, T. Nagai, K. Funakoshi, M. Nakano, and N. Iwahashi, "Autonomous acquisition of multimodal information for online object concept formation by robots," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 1540–1547.

[16] A. Droniou, S. Ivaldi, and O. Sigaud, "Deep unsupervised network for multimodal perception, representation and classification," *Robotics and Autonomous Systems*, vol. 71, pp. 83–98, 2015.

[17] S. Hangl, E. Ugur, S. Szedmak, A. Ude, and J. Piater, "Reactive, Task-specific Object Manipulation by Metric Reinforcement Learning," in *17th International Conference on Advanced Robotics*. IEEE, 7 2015, pp. 557–564.

[18] L. Rozo, P. Jiménez, and C. Torras, "Force-based robot learning of pouring skills using parametric hidden markov models," in *Robot Motion and Control (RoMoCo), 2013 9th Workshop on*. IEEE, 2013, pp. 227–232.