An Incremental Approach to Learning Generalizable Robot Tasks from Human Demonstration

Conference Paper *in* Proceedings - IEEE International Conference on Robotics and Automation · August 2015

DOI: 10.1109/ICRA.2015.7139985

4 authors:

Amir Ghalamzan
University of Lincoln
23 PUBLICATIONS **95** CITATIONS
SEE PROFILE

Chris Paxton
Johns Hopkins University
18 PUBLICATIONS **111** CITATIONS
SEE PROFILE

Gregory D. Hager
Johns Hopkins University
463 PUBLICATIONS **15,338** CITATIONS
SEE PROFILE

Luca Bascetta
Politecnico di Milano
107 PUBLICATIONS **683** CITATIONS
SEE PROFILE

An Incremental Approach to Learning Generalizable Robot Tasks from Human Demonstration

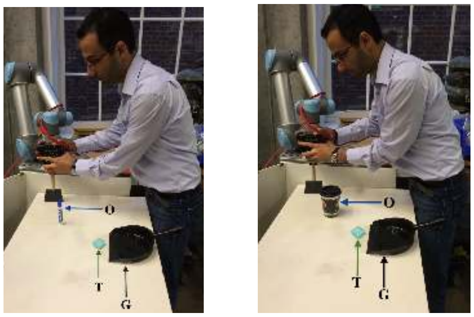
Amir M. Ghalamzan E., Chris Paxton, Gregory D. Hager, Luca Bascetta

***Abstract*— Dynamic Movement Primitives (DMPs) are a common method for learning a control policy for a task from demonstration. This control policy consists of differential equations that can create a smooth trajectory to a new goal point. However, DMPs only have a limited ability to generalize the demonstration to new environments and solve problems such as obstacle avoidance. Moreover, standard DMP learning does not cope with the noise inherent to human demonstrations. Here, we propose an approach for robot learning from demonstration that can generalize noisy task demonstrations to a new goal point and to an environment with obstacles. This strategy for robot learning from demonstration results in a control policy that incorporates different types of learning from demonstration, which correspond to different types of observational learning as outlined in developmental psychology.**

I. Introduction

Dynamic Movement Primitives (DMPs) have been proposed as a method for teaching a robot a skill from a task demonstration and reproducing the task with different start and end points [1]. While they have been successfully used to capture motion primitives, DMPs have a very limited ability to adapt to new environments. Further, ordinary DMPs will reproduce any noise present in a suboptimal human demonstration.

For example, assume that a person is teaching a household service robot how to sweep rubbish into a dustpan by moving its arm to provide a demonstration. This is a challenging problem that has been explored in prior work [2]. To perform the task, the robot should follow a noise-free average trajectory/path computed from the suboptimal human demonstrations, for example one learned via Gaussian Mixture Model/Gaussian Mixture Regression (GMM/GMR) [3]. Alternatively, we might acquire a DMP from the noisy demonstrated trajectory and use it to generate an appropriate path to different goal points: positions of the dustpan (Fig. 7). However, the generated path would not be applicable if a mouse or a coffee cup were in the robot's way. The robot should ideally be capable of adapting the learned skill to the new situation based on the provided demonstrations. In Figure 1, we show a version of this task where a human is sweeping a green block while avoiding a marker and a coffee cup. The human does not want to knock the marker over, so the robot should learn a different object avoidance policy for each object.



(a) (b)

Fig. 1. Teaching a robot to perform a sweeping task. (a) shows a marker being used as an obstacle o during demonstration of sweeping a green cube T into a dustpan G; (b) shows a cup as an obstacle.

A number of other methods have been developed to expand DMPs to cope with new environments in the presence of obstacles. In [4], variability of different demonstrations was used to estimate the stiffness matrices in a modified version of the DMP model. Then, a risk indicator modulating repulsive force was defined to enable a robot to safely avoid collision with a human. Guenter et al. [5] developed a model of a task using a dynamical system modulated by a Gaussian mixture model. The model was combined with reinforcement learning to enable the robot to learn a new way of accomplishing a task in a constrained environment.

In another work, Kormushev et al. [6] initialized a modified model of DMPs with imitation learning and used reinforcement learning to compute the optimal parameter values of the model for a new environment. Park et al. [7] added the gradient of a dynamic potential field to the acceleration term of the differential equation of the DMPs. The potential field depended on the relative distance and velocity between a robot's end effector and an obstacle. Hoffmann et al. [8] also added an acceleration term to the equations of motion in the DMP formulation to avoid colliding with a moving obstacle, relating the position of the end effector to the position of the obstacle.

In these works, parameters for obstacle avoidance were explicitly included in the problem. Therefore, the robot does not learn the desired responses to different objects, and a non-expert person cannot modify the behavior of the robot in response to the environment based on task requirements. Additionally, while methods using GMM/GMR [3] have been developed to compute an average trajectory from a set of suboptimal task demonstrations, sub-optimality will be

A. M. Ghalamzan E. and L. Bascetta are from the Department of Electronic, Information and Bioengineering, Politecnico di Milano, Piazza Leonardo Da Vinci 32, 20133, Milano, Italy (email: {amir.ghalamzan, luca.bascetta}@polimi.it).

C. Paxton and G. D. Hager are from the Department of Computer Science, Johns Hopkins University, 3400 N. Charles St. Baltimore, MD 21218-2686, USA (email: cpaxton3@jhu.edu, hager@cs.jhu.edu).

preserved in a DMP capturing that task through a suboptimal demonstration.

In this work, we propose a three-tiered approach for robot Learning from Demonstration (robot LfD). This approach deals with noisy demonstrations, generalizes the demonstrated task to different goal points, and learns how to encode the desired user response to different features of the environment, e.g. distance from an obstacle. With the proposed method, the user can teach the robot a desired behavior in the response to different classes of objects, e.g. to keep far from a mouse and not very far from a coffee cup.

In Section II, we describe our robot LfD approach inspired by observational learning. Section III and IV, contain the problem formulation and algorithm. Then, in Section V, we present two experiments to illustrate how the proposed method can be used to teach a robot to perform a pick-and-place task as well as how to sweep rubbish into a dustpan.

II. Incremental Learning from Demonstration

In this paper, we propose an alternative solution to the robot learning from demonstration problem inspired by human skill learning. According to studies of observational learning [9], [10], humans learn to perform tasks from demonstration at three different levels: mimicking, imitation and emulation. *Mimicking* is the copying of a model's bodily movements [10]. Thompson et al. [9] mentioned that mimicking must involve no conceptualization by the observer concerning the purpose of the action. Hence, it may not be possible to accomplish the task in a new environment through mimicking alone. Whiten et al. [11] defined *imitation learning* as a goal oriented copying the form of an observed action. In imitation learning, an observer is assumed to recognize what the form of the model's movements is bringing about and use that to carry out the task; it is therefore analogous to a traditional DMP. Lastly, in *emulation learning*, the observer replicates the expected results of the model's action [10]. We propose an incremental approach for robot learning from demonstration based on these three types of learning, outlined in Figure 2.

First, when mimicking a skill, we compute a noise-free average path from a set of suboptimal demonstrations, modeled as a Gaussian Mixture Model [3]. We refer to this model as the estimated nominal path ζ^N. The robot can replicate the task using the estimated nominal path if the environment is fixed and lacks any obstacles. However, this method alone cannot generalize to different environments.

Second, at the imitation learning level, the robot must be able to generalize a demonstration to a new start and goal point. We use DMPs to scale the nominal path ζ^N from a new start point $\mathbf{x}_{start}$ to a new goal point $\mathbf{x}_{goal}$ [1]. This results in an estimate of the underlying noise- and obstacle-free trajectory from any given $\mathbf{x}_{start}$ to $\mathbf{x}_{goal}$, denoted by $\zeta^N(\mathbf{x}_{start}, \mathbf{x}_{goal})$.

Finally, at the emulation learning level, based on the computed average path we use an Inverse Optimal Control (IOC) approach to compute a reward function whose optimal solution is as close as possible to the demonstrations. The IOC problem recovers a reward function R from a set of demonstrations, where the reward function is the sum of an imitation reward function R_I describing the tendency to follow the nominal path ζ^N and an emulation reward R_E describing the expected response to the environment. This strategy incorporates the noise-free skill obtained from human demonstrations with the desired user response to different environmental features in a single reward function, which can be used to generalize the task to new goal points and new environments with different obstacles.

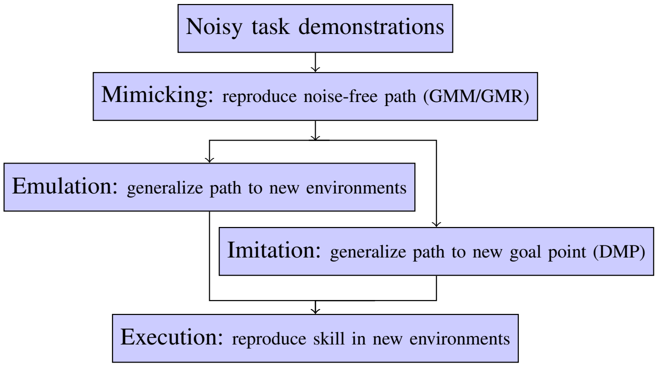

Noisy task demonstrations
Mimicking: reproduce noise-free path (GMM/GMR)
Emulation: generalize path to new environments
Imitation: generalize path to new goal point (DMP)
Execution: reproduce skill in new environments

Fig. 2. Overview of the proposed method inspired by human skill learning. In the mimicking step, we compute the average path from a set of demonstrations. We create a model that can be scaled to different start and goal positions (imitation) and learn a parameterized cost function modeling the appropriate response to different objects in the environment (emulation).

III. Reward Function Formulation

In order to integrate the different types of learning into a single model of a demonstrated task, we formalize the learning from demonstration problem posed above as an optimal control problem.

A. Optimal Control Formulation

We assume a set of demonstrated trajectories $\mathscr{D}$ such that each demonstration $\zeta_d \subset \mathscr{D}$, $\forall d = 1, ..., N_{demo}$, is an optimal solution to an unknown reward function in the corresponding environment E_d corrupted by noise, $E_d = \{\mathbf{O}_1, ..., \mathbf{O}_{N_{obs}}\}$, $\forall d = 1, ..., N_{demo}$, where N_{demo} is the number of demonstrations and N_{obs} is the number of obstacles O in d^{th} environment. We further assume that there exist N_{class} different object classes, $\mathscr{C} = \{C_1, \ldots, C_{N_{class}}\}$. Each class represents a set of objects with specific size and shape.

The optimal control problem is defined by a state space $\mathscr{S} \in \mathbb{R}^n$, an action space $\mathscr{A} \in \mathbb{R}^m$, a state transition function $T(\mathbf{s} \in \mathscr{S}, \mathbf{a} \in \mathscr{A}) : \mathbb{R}^{n+m} \to \mathbb{R}^n$, and a reward function $R(\mathbf{s} \in \mathscr{S}) : \mathbb{R}^n \to \mathbb{R}$. We assume the reward function is a function of the state of the actor $x \in \mathbb{R}^p$ and of the features of the environment $\mathbf{f} \in \mathbb{R}^q$, so that $R(\mathbf{s} = \{\mathbf{x}, \mathbf{f}\})$ and $n = p + q$. Our goal is to learn a reward function that allows us to compute the optimal action $\mathbf{a} \subset \mathscr{A}$ at each state $\mathbf{s} \subset \mathscr{S}$ for a new unobserved environment based on this reward function.

Assume a robot can optimally perform a task by following a nominal path ζ^N in an environment without any obstacle. The demonstrated paths ζ_d may not be identical to the nominal path because of the presence of obstacles in the corresponding environment and/or the noise. We estimate the nominal path as a Gaussian Mixture Model, and assume that the recovered reward function must be a function of the nominal model of the task and the corresponding features of environment. Features of the environment depend on x given a scene. However, for the sake of simplicity, we write $\mathbf{f}$ instead of $\mathbf{f}(\mathbf{x})$.

We consider the problem of performing a task to be an episodic, deterministic, optimal control problem with fixed time horizon T_e in discrete time, in a continuous state-space and action-space and with a known world model. As per [12], our goal is to learn the underlying reward function $R(s)$ from the demonstrations $\mathscr{D}$. We decompose the underlying reward function $R(\mathbf{x},\mathbf{f})$ into two components: an imitation component $R_I(\mathbf{x})$, whose optimal solution will be identical to the nominal path, and an emulation component $R_E(\mathbf{f})$, that encodes the response of the robot to the environmental features.

Given a reward function $R = R_I + R_E$, we maximize the expected return $\rho^{\pi} = \sum_{i=1}^{T_e-1} R(\mathbf{s}_{i+1})$, to find an optimal policy π^* defined as:

$$\begin{aligned} \pi^* = \underset{\pi}{\operatorname{argmax}} \sum_{i=1}^{T_e-1} & \left(R_I\left(\mathbf{x}_{i+1}\right) + R_E\left(\mathbf{f}_{i+1}\right)\right) \\ \text{subj. to} \quad & \mathbf{s}_{i+1} = T(\mathbf{s}_i, \mathbf{a}_i), \\ & \mathbf{a}_i \in \mathscr{A}, \end{aligned} \tag{1}$$

where $\pi = \{\mathbf{a}_1, ..., \mathbf{a}_{T_e-1}\}$ is the sequence of actions that a robot takes to accomplish the task and $\mathscr{A}$ is a polyhedral region that is a feasible subset of the set of all actions. By executing the optimal policy π^*, the robot follows a sequence of states $\bar{\zeta} = \{\mathbf{s}_1, \bar{\mathbf{s}}_2, ..., \bar{\mathbf{s}}_H\}$, where $\mathbf{s}_1$ is a given initial condition. We will now discuss the imitation component R_I and emulation component R_E of the reward function.

Imitation component of the reward function: We represent the model producing a generalized nominal path to a new goal point as a DMP. This DMP is produced by the imitation learning step described above: it generalizes the estimated nominal path ζ^N to a new start and goal point. The estimated nominal path is first learned as a Gaussian Mixture Model as described in [3], with $K = 4$ components initialized by k-means clustering. This model allows us to combine multiple demonstrations across slightly different environments with the same start and goal positions. The GMM removes noise from the human demonstrations; as such, we assume it is the noise-free optimal solution to performing a task with no obstacles in the environment.

We generate a trajectory from this GMM using Gaussian Mixture Regression (GMR). As per [3], we input a set of times and use GMR to recover the expected robot state x at each. This trajectory is used to learn the DMP model used to generalize to new start and goal positions. Note that it would be possible to replace this with the task-parameterized GMM approach described in [13] with very little modification; we use the two step approach to illustrate the parallels with human skill acquisition.

We can then model the imitation reward function in terms of deviation from the nominal path ζ^N generalized by the DMP:

$$\begin{aligned} R_I\left(\mathbf{x}_i : \mathbf{Q}\right) = -\left(\mathbf{x}_i - \mathbf{x}_i^{\mathrm{N}}\right)^T \mathbf{Q}\left(\mathbf{x}_i - \mathbf{x}_i^{\mathrm{N}}\right) \\ \mathbf{x}_i^N \subset \zeta^N \ , \ i = 1, ..., T_e \end{aligned} \tag{2}$$

where $\mathbf{x}_i^{\mathrm{N}} \in \mathbb{R}^n$ is a point on the nominal path ζ^N. We search along a line normal to ζ^N to find the optimal solution at each time step i, $i = 1, ..., T_e$, where the point of the nominal path corresponding to the time step i is $\mathbf{x}_i^{\mathrm{N}}$. The line segment between the point $\mathbf{x}_i^{\mathrm{N}}$ on the nominal path at time i and the next state $\mathbf{x}_i$ for the current environment, $\overline{\mathbf{x}_i^N \mathbf{x}_i}$, must be perpendicular to ζ^N. This allows us to restrict our search space and solve the problem in discrete time, because the state space normal to ζ^N at time i is continuous. For further details, see [14].

Emulation component of the reward function: The optimal solution for the emulation component problem is not invariant over different distributions of obstacles. Since deviation from the nominal model is local, a Gaussian function with covariance matrix $\mathbf{R}$ can be learned from features of the demonstration data. This gives us the emulation component of the reward function R_E:

$$R_E\left(\mathbf{f}_i : \mathbf{R}\right) = \sum_{j=1}^{N_{obs}} -e^{\left(-\mathbf{f}_{i,j}^T \, \mathbf{R}^{-1} \, \mathbf{f}_{i,j}\right)} \tag{3}$$

where $\mathbf{f}_{i,j} \in \mathbb{R}^q$ is a vector of the environmental features at $\mathbf{x}_i$ related to the j^{th} obstacle, captured during the d^{th} demonstration. While in theory any set of environmental features could be used, we describe response to different types of obstacles as a function of distance to those obstacles. We express f in terms of the given obstacles positions O in the d^{th} environment E_d. This lets us rewrite eq. (3) as:

$$R_E\left(\mathbf{x}_i, E_d : \mathbf{R}\right) = \sum_{j=1}^{N_{obs}} -e^{\left(-(\mathbf{x}_i - \mathbf{O}_j)^T \, \mathbf{R}^{-1} \, (\mathbf{x}_i - \mathbf{O}_j)\right)} \tag{4}$$

Accordingly, the general reward function, $R\left(x, E_d : \theta\right)$ characterizing the demonstrated behavior in a different environment with added obstacles is a combination of the emulation component given by eq. (3) and the imitation component given by eq. (2) as follows:

$$R(\mathbf{s}, E_d : \theta) = -\left(\mathbf{x}_i - \mathbf{x}_i^{\mathrm{N}}\right)^T \mathbf{Q}\left(\mathbf{x}_i - \mathbf{x}_i^{\mathrm{N}}\right) - \sum_{j=1}^{N_{obs}} e^{-\mathbf{f}_{i,j}^T \, \mathbf{R}^{-1} \, \mathbf{f}_{i,j}} \tag{5}$$

where $\theta = \{\mathbf{Q}, \mathbf{R}(\mathbf{O}_1), ..., \mathbf{R}(\mathbf{O}_{N_{obs}})\}$, $\mathbf{Q}$ and $\mathbf{R}(\mathbf{O}_l)$ being positive definite matrices. For the sake of simplicity we consider $\mathbf{Q}$ and $\mathbf{R}$ to be diagonal. In the following, we compute a set of parameters, $\theta = \{\mathbf{Q}, \mathbf{R}\}$ and $\mathbf{R} = \{\mathbf{R}(C_1), ..., \mathbf{R}(C_{N_{class}})\}$ where $\mathbf{R}(C_{N_{class}})$ represents the emulation parameters corresponding to the object class $C_{n_{class}} \subset \mathscr{C} \; \forall \, n_{class} = 1, ..., N_{class}$.

B. Inverse Optimal Control

Inverse optimal control aims at finding a reward function whose optimal solution is as close as possible to the demonstrations. Given an estimated reward function one can use existing methods, such as dynamic programming or reinforcement learning, to find a solution $\bar{\zeta}_{R(\theta,E_d)}$ to eq. (1). In our case, to learn the parameters of the reward function we minimize the cumulative distances between the optimal solution $\bar{\zeta}_{R(\theta,E_d)}$ and the demonstrations as follows:

$$\theta = \underset{\theta}{\operatorname{argmin}} \sum_{d=1}^{D} \sum_{i}^{T_e} \left\| \bar{\zeta}_{R(\theta,E_d)}(i) - \zeta_d(i) \right\|^2 \tag{6}$$

where $\zeta_d(i)$ and $\bar{\zeta}_{R(\theta,E_d)}(i)$ are the corresponding points on the demonstration and the solution to the estimated reward function. The parameters θ of the reward are iteratively computed by minimizing eq. (6) with *minConf*, using a quasi-Newton strategy and limited-memory BFGS updates [15].

IV. Solution to the Learned Reward Function

We maximize the expected return of eq. (1) in order to compute the optimal solution to the learned reward function for a new environment. In a finite-horizon problem, optimal control aims at finding the optimal policy by determining a sequence of actions $\bar{a}$ maximizing the expected return. In this paper, model predictive control is used to find an optimal solution to the learned reward function with continuous state and action spaces.

Consider a prediction time horizon T_p, the optimal action corresponding to the proposed problem in eq. (1) at i^{th} time step can be formulated as follows:

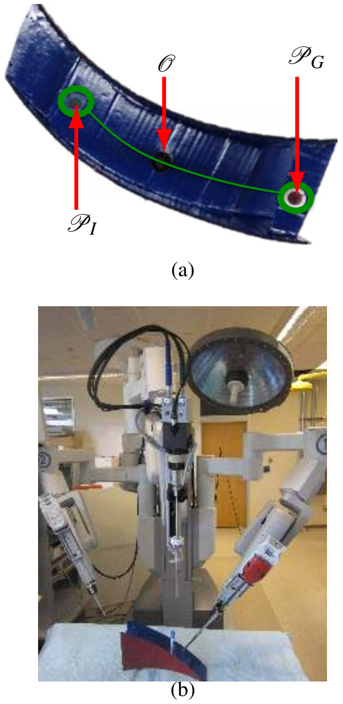
$$\begin{aligned} \bar{a}_i = \underset{\mathbf{a}_i}{\operatorname{argmax}} & \sum_{t=i}^{T_p+i} R(\mathbf{s}_t) \\ \text{subj. to} \quad \mathbf{z}_{t+1} &= \mathbf{A}\mathbf{z}_t + \mathbf{B}\mathbf{a}_t \\ \mathbf{x}_t &= \mathbf{C}\mathbf{z}_t \\ \mathbf{a}_t &\in \mathscr{A} \\ \mathbf{x}_t &\in \mathscr{X} \\ i = 1,2,&...,T_e-1, \end{aligned} \tag{7}$$

where $\mathscr{X}$ and $\mathscr{A}$ are the polyhedral feasible sets of actor states and actions respectively, and $\pi^* = \{\bar{a}_1,...,\bar{a}_{T_e-1}\}$ is a sequence of optimal actions where its corresponding sequence of optimal states is $\bar{\zeta}_{R(\theta,E_d)} = \{\mathbf{x}_1, \bar{\mathbf{x}}_2, ..., \bar{\mathbf{x}}_{T_e}\}$ with initial value $\mathbf{x}_1$.

In eq. (7) a linear dynamical system is considered as the transition function of the actor in eq. (1). We select A, B and C such that they represent a stable linear dynamical system in the receding horizon formulation in eq. (7); i.e. the magnitudes of the eigenvalues of A are all less than one. It is assumed that the actor moves with constant velocity along the nominal path. To find a solution to eq. (6) and (7), we use *minConf* with a quasi-Newton strategy and limited-memory BFGS updates [15]. It is worth mentioning that the asymptotic stability of the proposed receding horizon formulation for a path planning problem was discussed by Xu et al. [16].

V. Experiments

We present two experiments: picking and placing an object during surgical training and sweeping an object into a dustpan while avoiding various objects. These tasks demonstrate the usefulness of the proposed approach for solving real-world problems based on data from noisy expert demonstrations. We assume that all the demonstrations have the same number of sample points and the same duration.



(a)

(b)

Fig. 3. (a) The task model used for data collection with da Vinci surgical robot with a single obstacle (marker), the nominal path that expert follows in the absence of obstacles (green line); (b) da Vinci set up to collect expert demonstrations.

The task of picking and placing an object is a common task during surgeon training, and many surgical tasks are constrained by a small available space within a patient's body. To obtain a constrained environment we designed the structure shown in Fig. 3(a). The structure has two walls constraining the movement of the robotic tool during picking the object from $\mathbf{x}_{start}$ and placing at $\mathbf{x}_{goal}$. In the first experiment a da Vinci robot shown in Fig. 3(b) was used to collect a set of demonstrations of a simple task: moving an object from point $\mathbf{x}_{start}$ to point $\mathbf{x}_{goal}$ of this structure.

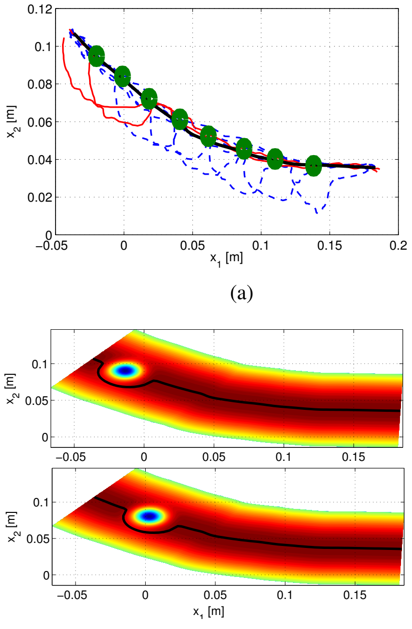
The operator was asked to maximize the distances from both walls while performing the task. A marker was fixed to the scene during execution of the task as an obstacle. This marker can be thought of as another instrument or a fragile tissue that needs to be avoided during an operation.

The operator performed the task several times with different positions of the marker; these demonstrations constitute our training data set

$$\zeta_d = \{\mathbf{x}_{d,1}, \ldots, \mathbf{x}_{d,T_e}\},\ d = 1, \ldots, N_{demo}$$

where N_{demo} was the number of demonstrations. Demonstrations can be seen in Fig. 4(a).

At the mimicking level, we use GMM/GMR to compute the nominal path $\zeta^N = \{\mathbf{x}_1^{\mathrm{N}}, \ldots, \mathbf{x}_{T_e}^{\mathrm{N}}\}$, from these demonstrations. Since the data points close to the obstacle do not represent the underlying nominal path, for average path computation by GMM/GMR, we exclude those data points



(a)

(b)

Fig. 4. (a) The data set collected with da Vinci robot for different positions of the marker (green shaded circles), training set (red sold line) and test set (blue dashed line), the average path computed by GMM/GMR (black thick line); (b) The contour of the learned reward function for the da Vinci experiment, and the computed optimal solution to the reward function. Areas with hotter colors represent positions with higher associated rewards.

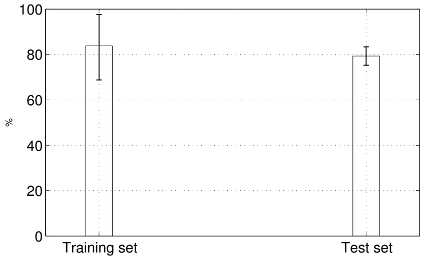


Fig. 5. The precision of the reproduced path with respect to use of the computed average path, ζ^N.

to get a better estimation of the underlying nominal path.

At the emulation level, the estimated nominal path ζ^N is used as a reference to build the emulation reward function $R_E(f)$ that encodes the response of the operator to the obstacles. In order to evaluate the obtained model of the task, the collected data set of task demonstrations is divided into a training set and a test set (Fig. 4(a)).

The parameters of the reward function are computed using eq. (6) from the training set, $Q = diag[200,\ 200]$ and $R^{-1} = diag[475.8,\ 8576.3]$. The learned model of the task (shown in Fig. 4(b)) is then used to generate the paths, $\zeta_R = \{\mathbf{x}_1^R, \ldots, \mathbf{x}_{T_e}^R\}$, corresponding to the paths within the test set. In order to validate the method and evaluate the obtained task model, the mean square error (MSE) of the generated paths is computed for both test set and training set, as follows:

$$MSE_R = \frac{1}{D_{.}T_e}\sum_{d=1}^{D_{.}}\sum_{i=1}^{T_e}\|\mathbf{x}_{d,i} - \mathbf{x}_i^R\|^2$$

where H is the number of sample points of the demonstrated path and $D_{.}$ is the number of paths within the training set (D_{tr}) or test set (D_{tst}). The MSE of the obtained optimal solutions for the test set is equal to $e = 0.0911[mm^2]$ with variance $var(e) = 0.0064$, illustrating that the obtained model results in small error values.

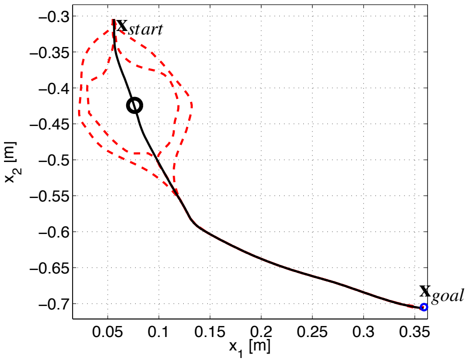


Fig. 6. The data set of sweeping task with initial point $\mathbf{x}_{start}$ and the goal point $\mathbf{x}_{goal}$. There are two demonstrations with marker and two with cup in the scene whose positions are shown by a black circle.

The computed MSE represents the error of the generated path with respect to the demonstrated one. Two main sources constitute this error, the distance between the computed average path and the demonstrated one and the adaptation component. In order to evaluate the obtained adaptation component of the model, a precision value of each generated path (Fig. 5) is defined and computed as follows

$$Pr = \frac{MSE_{av} - MSE_R}{MSE_{av}} \times 100$$

where

$$MSE_{av} = \frac{1}{DT_e}\sum_{d=1}^{D}\sum_{i=1}^{T_e}\|\mathbf{x}_{d,i} - \mathbf{x}_i^N\|^2$$

and $\mathbf{x}_i^N$ is a point of the computed average path by GMM/GMR. The computed Pr for the training set and the test set in Fig. 5 shows that the obtained model with the training set generates the paths corresponding to the ones in the test set with good precision values.

The second experiment concerns an UR5 robot that learns a sweeping task in a varying environment from a few demonstrations, which can then be applied to new environments with unseen obstable configurations. In the past few years, due to the development of robots that can safely perform different tasks out of the cage such as Roomba Vacuum Cleaning Robot and Erector Spykee, many studies have been conducted to allow robots to better adapt to unseen environments [17]. As an example, consider the task of sweeping a green cube into a dustpan while avoiding various obstacles, shown in Fig. 1. The task is demonstrated a few times in the presence of two objects, a marker and a cup, that we do not want to sweep (Fig. 6). The learning processes takes place as follows:

- Mimicking: an average path is computed from the set of task demonstrations;
- Imitation: a nominal path is computed for a new dustpan position $\mathbf{x}'_{goal}$, different from the demonstrated one $\mathbf{x}_{goal}$ (Fig. 7(c) and 7(d))

- Emulation: The nominal path is used to build the a reward function incorporating response to obstacles, which is used to generate the necessary path corresponding to new positions of the dustpan and the obstacles.

The obtained reward function is a model of the task that changes according to the position of the objects in the environment and of the dustpan. The obtained task model copes with noisy demonstrations, perturbed target positions and different locations of obstacles. The results, shown in Fig. 7, illustrate the effectiveness of the approach in capturing the responses to different object classes and in generalizing the learned skill to the new goal point.

When reproducing the task, we consider the orientation of the broom to be normal to the generated trajectory.

VI. Conclusion

Inspired by different types of observational learning, we discussed an incremental strategy for robot LfD employing learning from demonstration at three different levels. The proposed approach allows a robot to learn both how to execute a task with different start and goal positions and how to adapt the obtained task model to a new environment with an unseen obstacle configuration. This learning can be performed based on sub-optimal expert demonstrations, unlike typical DMP learning.

Although in prior work obstacle avoidance has been addressed by combining planning methods for obstacle avoidance with DMPs, these works do not allow a non-expert user to teach a robot the desired response to different objects. The proposed approach incorporates GMM/GMR, DMPs and IOC into a reward function used to generate the necessary path in a new situation. This integrated approach inspired by studies of observational learning in psychology is what allows the robot to learn a skill from noisy demonstrations and to learn the desired user responses to different classes of objects in the environment.

In this work, we considered only static obstacles. For future work, we plan to extend the approach to reproduce tasks in dynamic environments based on learned parameters.

This paper is accompanied by a video of the sweeping task, showing the ability of the proposed approach to adapt to unseen environments.

VII. Acknowledgments

We would like to acknowledge Dr. Simon Leonard for his assistance collecting data from the Da Vinci.

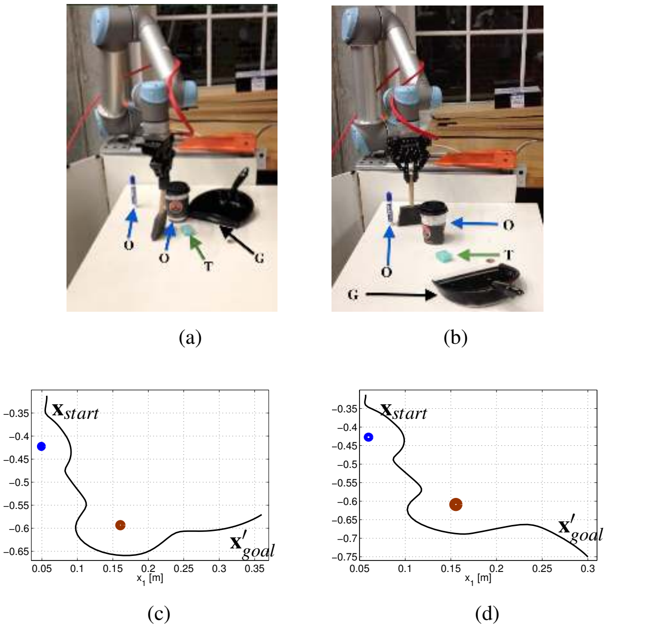
References

[1] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, vol. 2. IEEE, 2002, pp. 1398–1403.

[2] T. Alizadeh, S. Calinon, and D. Caldwell, "Learning from demonstrations with partially observable task parameters," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, 2014.

[3] S. Calinon, *Robot Programming by Demonstration*. EPFL Press, 2009.

[4] S. Calinon, I. Sardellitti, and D. G. Caldwell, "Learning-based control strategy for safe human-robot interaction exploiting task and robot redundancies," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. IEEE, 2010, pp. 249–254.



(a) (b)

(c) (d)

Fig. 7. (a) and (b) snapshots of the scenes used for task reproduction with different position of the dustpan and obstacles; (c) and (d) the generated paths corresponding to (a) and (b).

[5] F. Guenter, M. Hersch, S. Calinon, and A. Billard, "Reinforcement learning for imitating constrained reaching movements," *Advanced Robotics*, vol. 21, no. 13, pp. 1521–1544, 2007.

[6] P. Kormushev, S. Calinon, and D. G. Caldwell, "Robot motor skill coordination with em-based reinforcement learning," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. IEEE, 2010, pp. 3232–3237.

[7] D. Park, H. Hoffmann, P. Pastor, and S. Schaal, "Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields," in *8th IEEE-RAS Int. Conf. on Humanoid Robots*. IEEE, 2008, pp. 91–98.

[8] H. Hoffmann, P. Pastor, D. Park, and S. Schaal, "Biologically-inspired dynamical systems for movement generation: automatic real-time goal adaptation and obstacle avoidance," in *Proc. of IEEE Int. Conf. on Robotics and Automation*. IEEE, 2009, pp. 2587–2592.

[9] D. E. Thompson and J. Russell, "The ghost condition: imitation versus emulation in young children's observational learning." *Developmental Psychology*, vol. 40, no. 5, pp. 882 – 899, 2004.

[10] A. Whiten, N. McGuigan, S. Marshall-Pescini, and L. M. Hopper, "Emulation, imitation, over-imitation and the scope of culture for child and chimpanzee," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 364, no. 1528, pp. 2417 – 2428, 2009.

[11] A. Whiten and R. Ham, "On the nature and evolution of imitation in the animal kingdom: reappraisal of a century of research," *Advances in the Study of Behavior*, vol. 21, no. 1, pp. 239–283, 1992.

[12] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proc. of the 21st int. conf. on Machine learning*. ACM, 2004, pp. 1 – 9.

[13] S. Calinon, D. Bruno, and D. G. Caldwell, "A task-parameterized probabilistic model with minimal intervention control," in *Proc. of IEEE Int. Conf. on Robotics and Automation*. IEEE, 2014, pp. 3339–3344.

[14] A. M. Ghalamzan E., L. Bascetta, M. Restelli, and P. Rocco, "Estimating a mean-path from a set of 2-d curves," in *Proc. of IEEE Int. Conf. on Robotics and Automation*. IEEE, 2015, pp. –.

[15] M. Schmidt, "Graphical model structure learning with l1-regularization," Ph.D. dissertation, University of British Columbia, 2010.

[16] B. Xu, A. Kurdila, and D. Stilwell, "A hybrid receding horizon control method for path planning in uncertain environments," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. IEEE, 2009, pp. 4887–4892.

[17] T. Denning, C. Matuszek, K. Koscher, J. Smith, and T. Kohno, "A spotlight on security and privacy risks with future household robots: attacks and lessons," in *Proc. of the 11th int. conf. on Ubiquitous computing*. ACM, 2009, pp. 105–114.