Demonstration Based Trajectory Optimization for Generalizable Robot Motions

Dorothea Koert[1], Guilherme Maeda[1], Rudolf Lioutikov[1], Gerhard Neumann[1] and Jan Peters[1,2]

***Abstract*— Learning motions from human demonstrations provides an intuitive way for non-expert users to teach tasks to robots. In particular, intelligent robotic co-workers should not only mimic human demonstrations but should also be able to adapt them to varying application scenarios. As such, robots must have the ability to generalize motions to different workspaces, e.g. to avoid obstacles not present during original demonstrations. Towards this goal our work proposes a unified method to (1) generalize robot motions to different workspaces, using a novel formulation of trajectory optimization that explicitly incorporates human demonstrations, and (2) to locally adapt and reuse the optimized solution in the form of a distribution of trajectories. This optimized distribution can be used, online, to quickly satisfy via-points and goals of a specific task. We validate the method using a 7 degrees of freedom (DoF) lightweight arm that grasps and places a ball into different boxes while avoiding obstacles that were not present during the original human demonstrations.**

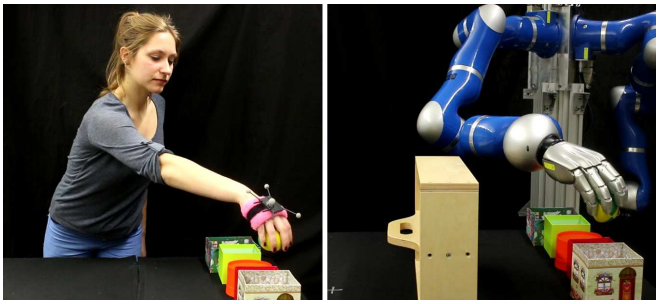

Fig. 1. (Left) As human movements are inherently stochastic, multiple demonstrations allow us to reveal the temporal and spatial correlations that govern the motion to achieve human-like robot motions. (Right) When the application workspace differs from the demonstrated scenario, e.g by obstacles not present during the demonstrations, the robot should be able to adapt the learned movements.

I. INTRODUCTION

Future generations of collaborative robots will not only change production paradigms in human-robot shared work spaces, but also address the unprecedented growth of the elderly population. Such robots will inevitable face a variety of unforeseen tasks; rendering manual pre-programming of motions unrealistic in practice. While methods such as kinesthetic teaching can be time-consuming or even infeasible for non-expert workers and non-backdriveable robots, a very natural and intuitive way to teach robots is to provide observations of human movements. However, complex tasks in dynamic environments require robots to go beyond simply mimicking the human. Robots must adapt their motions to unforeseen changes in the work space, different human partners or varying task constraints.

We propose a method which provides two main advantages compared to similar approaches from the fields of trajectory optimization and motion planning. First, our algorithm is able to optimize a distribution of trajectories (as opposed to a single trajectory), preserving the temporal and spatial correlation of human demonstrations. The use of demonstrations reduces the need for prior assumptions such as trajectory smoothness, actuator usage, and jerkiness, which are usually hand-coded in a cost function. Second, our algorithm is able to generalize the demonstrated motions not only for workspace changes such as obstacles, which were not present during the original demonstrations, but can also quickly adapt the robot motions to changing task specific constraints, such as different goals or via points. A probabilistic representation of the trajectories allows us to do the latter by encoding the optimized distribution as a movement primitive.

While path planning and trajectory optimization have been topics of research for many years, existing algorithms mainly focus on robot issues such as feasibility, joint constraints or energy minimization [1], [2]. Common to most standard trajectory optimization approaches is the fact that human demonstrations are not taken into account during the optimization process. Incorporating human demonstrations can provide means for the generation of more human-like motions [4], in particular in human-robot shared environments.

Moreover, trajectory optimization, even in stochastic settings, is mostly limited to generate a single optimal solution. Here, we optimize an entire distribution over trajectories, rather than a single optimal trajectory. As it will be clear in the next sections, this will be key to generalize human demonstrations to different robot workspaces and task specific constraints.

The contributions of this paper are twofold. First, we introduce an offline optimization algorithm for motion planning that optimizes trajectory distributions. Within this optimization the temporal and spatial correlations of the trajectories are extracted directly from human movements. Collision-free trajectories for obstacles not present during the human demonstrations can then be sampled from the optimized distribution over trajectories. Second, we show how to represent this optimized distribution as a probabilistic model such that the learned robot motions can be additionally general-

[1]Intelligent Autonomous Systems, Technische Universität Darmstadt, Hochschulstr. 10, 64289 Darmstadt, Germany
{koert,maeda,lioutikov,neumann,peters}@ias.tu-darmstadt.de
[2]Max Planck Institute Tuebingen

ized to satisfy different via points. To this end, statistical modeling methods offer temporal and spatial generalization and account for e.g. different desired via points during the robot movement [5], [6], [7], [8]. As our proposed trajectory optimization yields not only a single optimal trajectory but an optimized distribution of trajectories, it allows for a straightforward connection to those existing probabilistic movement modeling approaches.

II. Related Work

The idea of learning from demonstration has been a recent research topic [9], [10]. While methods such as kinesthetic teaching tend to constrain the demonstrators movement and might not even be feasible depending on the robot model or the task, a very natural and intuitive way of teaching is learning directly from unconstrained human demonstrations [11].

A core question is how to quickly adapt the learned motion to different and even unseen situations. Trajectory optimization and path planning and are long-standing research fields in robotics and have been studied extensively [12], [13], [14]. The basic problem is to find a solution for driving a robot from an initial configuration to a goal configuration while obeying certain constraints and avoiding obstacles in the robot's environment [15].

The adaptation and generalization of robot motions have been successfully achieved by local trajectory optimization methods [2], [1], [16]. Common to all these approaches is the fact that human demonstrations are not taken into account during the optimization process and smoothness is achieved by artificial prior assumptions such as penalizing accelerations [2], smooth kernels [1], [16] or temporally correlated noise [2]. Extracting information for motion planning and trajectory optimization out of few human trajectories has been used for different applications such as helicopter control [17], automatic constraint extraction [18] and early prediction of human motions [19]. However, these approaches do not use generalizable representation of motions and therefore the optimization needs to be run again whenever goals or via points change.

To obtain generalizable robot motions from human demonstrations a number of frameworks employ the concept of movement primitives [20], [21], [22]. Movement primitives enable the decomposition of complex movements to compact parametrization of robot policies [23]. The main idea is to encode a recorded trajectory in a way that can be used to generate different variations of the original movement in temporal as well as in spatial context.

Since human movements are inherently stochastic, multiple demonstrations allow us to reveal the temporal and spatial correlations that govern the motion. It is desirable to employ representations which are able to capture this variance in the motion such as extensions of Hidden Markov Models [5], Probabilistic Movement Primitives (ProMPs) [8] and Probabilistic Flow Tubes [7].

The particular aspect of motion planning and obstacle avoidance in combination with movement primitives and human demonstrations has been addressed for specific cases such as, Dynamic Movement Primitives [24], [25], a mixture of dynamical systems [26] or Task-Parameterized Gaussian Mixture Models [27]. However, in this work we propose a method for demonstration based trajectory optimization, where the trajectory optimization for obstacle avoidance is decoupled from the chosen movement representation. This allows for straightforward connection to any of the prior mentioned motion representations such as [5], [6], [7], [8].

To quickly generalize for different workspace settings, our proposed method does not optimize a single trajectory but rather a distribution over trajectories. Based on a probabilistic movement representation, we can leverage this distribution for online adaptation to given task constraints. The idea to consider variance and uncertainty in the motions is also considered in belief state planning approaches [28], but most of those do not take human demonstrations into account during the optimization. Another way to preserve a distribution during the optimization process is provided by using the Kullback-Leibler (KL) divergence as a measure to quantify deviations between the optimized solution and the original demonstrations [29]. In our approach we employ the KL divergence to measure deviations from the original demonstrated distribution.

III. Generalizable Robot Motions from Human Demonstrations

Our method comprises three main steps, which are depicted in Figure 2. First we collect human demonstrations of the desired behaviour in form of multiple trajectories (red). The second step processes these demonstrations during stochastic trajectory optimization (green). This optimization is performed off-line and produces a collision-free distribution that additionally is close to the empirical distribution inferred from the demonstrations. The online phase (blue) uses the optimized distribution, encoded as a probabilistic representation, to satisfy task-specific constraints. Such a probabilistic movement representation can be conditioned to quickly satisfy e.g. different desired via points. The connection of the demonstration based trajectory optimization with such a statistical movement model is feasible, since our proposed optimization outputs not only a single trajectory but an optimal, collision-free distribution.

This section introduces our method for demonstration based trajectory optimization and offers an overview of the probabilistic movement primitive structure which was used for the online adaptation to task constraints.

A. Demonstration Based Trajectory Optimization

We frame the trajectory optimization as a policy search problem, in which the policy defines a distribution from which trajectories can be sampled.

Given a robot trajectory $\boldsymbol{\tau} = \{\boldsymbol{x}^{[1]}, ..., \boldsymbol{x}^{[T]}\}$, where $\boldsymbol{x}^{[t]}$ denotes a state vector, of Cartesian or joint states, at time step t, the proposed optimization addresses two main objectives. First, it minimizes the deviation of the current policy p from the distribution of human demonstrations d measured as the

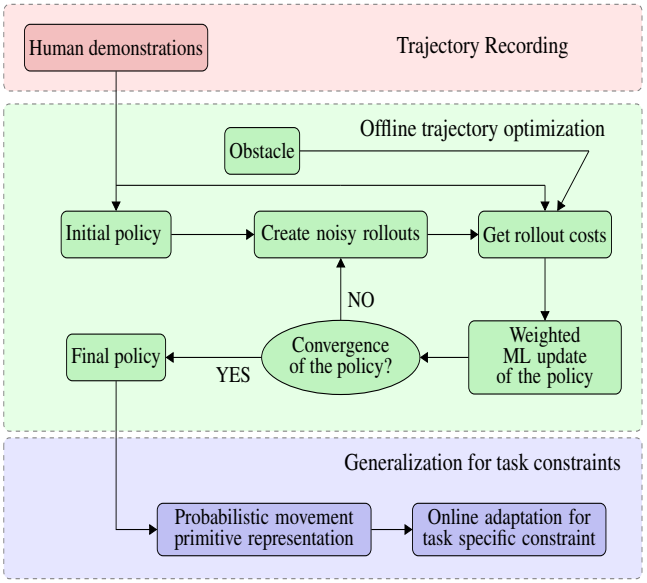

Fig. 2. The flowchart shows the structure of our framework. The proposed method builds on human demonstrations, collected as multiple trajectories (red). To process those demonstrations we introduce an off-line optimization algorithm for motion planning (green) that outputs a trajectory distribution whose temporal and spatial correlations are extracted directly from human movements. Collision-free trajectories for obstacles not present during the human demonstration can be sampled from this optimized distribution. Subsequently, we encode the result as a probabilistic representation. This allows to reuse the optimized distribution and quickly satisfy task constraints, such as via points, in an online fashion (blue).

KL-divergence between p and d

$$\text{cost}_{\text{KL}} = \int_{\boldsymbol{\tau}} p(\boldsymbol{\tau};\boldsymbol{\Theta}) \log\left(\frac{p(\boldsymbol{\tau};\boldsymbol{\Theta})}{d(\boldsymbol{\tau};\boldsymbol{\Theta_D})}\right) \mathrm{d}\boldsymbol{\tau}. \tag{1}$$

Here, $\boldsymbol{\Theta}$ denotes the policy parameters of the policy p and $\boldsymbol{\Theta_D}$ denotes the policy parameters of an empirical distribution, inferred from the demonstrated trajectories. Respectively $p(\boldsymbol{\tau};\boldsymbol{\Theta})$ and $d(\boldsymbol{\tau};\boldsymbol{\Theta_D})$ can be seen as the probability of $\boldsymbol{\tau}$ under the policy p and under the demonstrations. The formulation can be used for arbitrary distributions, but for our experiments we assumed the distribution over demonstrated trajectories, as well as the policy, to be Gaussian.

As a second objective we maximizes a reward function $R(\boldsymbol{\tau})$. For obstacle avoidance, a reward similar to the definition presented in STOMP [2] is used

$$R(\boldsymbol{\tau}) = -\sum_{t=1}^{T} \max\{\epsilon_{\text{dist}} - \Delta(\boldsymbol{x}^{[\text{t}]}), 0\}, \tag{2}$$

where ϵ_{dist} denotes a safety radius around the obstacle, that should not be touched by the trajectory, and $\Delta(x^{[\text{t}]})$ denotes the Euclidean distance of $x^{[\text{t}]}$ to the closest obstacle. Note that the policy p can start in an uninformed manner, that is, the initial guess on trajectories does not need to be smooth neither correlated. The optimization of (1) and (2) will naturally approximate d while trading off with the necessary deviations to satisfy the obstacle avoidance objective.

To optimize the policy, our method extends Relative Entropy Policy Search (REPS) [30] and incorporates (1) and (2) as

$$\begin{aligned} \underset{p}{\text{argmax}} \int_{\boldsymbol{\tau}} \left[p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]}) R(\boldsymbol{\tau}) - B p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]}) \log\left(\frac{p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})}{d(\boldsymbol{\tau};\boldsymbol{\Theta_D})}\right)\right] \mathrm{d}\boldsymbol{\tau} \\ \text{s.t.} \quad \int_{\boldsymbol{\tau}} p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]}) \log\left(\frac{p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})}{p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})}\right) \mathrm{d}\boldsymbol{\tau} \leq \varepsilon, \\ \int_{\boldsymbol{\tau}} p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]}) \mathrm{d}\boldsymbol{\tau} = 1, \end{aligned} \tag{3}$$

where ε denotes the upper bound for the deviation of the updated policy $p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})$ at the k-th iteration from the previous policy, $p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})$. As proposed in the original version of REPS, this constraint prevents huge jumps in the policy update step. The coefficient B is a scalar that trades off between minimal deviation from the demonstrated distribution and obstacle avoidance.

With the method of Lagrange multipliers, as explained in more detail in the Appendix, we obtain a closed form solution for the update rule of the policy

$$p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]}) \propto \exp\left(\frac{R(\boldsymbol{\tau})}{B+\eta}\right) \left(\frac{d(\boldsymbol{\tau};\boldsymbol{\Theta}_D)}{p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})}\right)^{\frac{B}{B+\eta}} p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]}), \tag{4}$$

where η is the Lagrange multiplier of the upper bound constraint. Note that (4) differs from the original solution in [30] as it explicitly incorporates the distribution of human demonstrations within the optimization.

In each iteration we use N samples $\boldsymbol{\tau}_1, ..., \boldsymbol{\tau_N}$ from the current policy $p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})$. In particular, if the policy is assumed to be Gaussian with $\boldsymbol{\Theta^{[k-1]}} = (\boldsymbol{\mu}^{[k-1]}, \boldsymbol{\Sigma}^{[k-1]})$, we obtain a new mean $\boldsymbol{\mu}^{[k]}$ and a new covariance matrix $\boldsymbol{\Sigma}^{[k]}$ with a weighted Maximum Likelihood update as follows

$$\boldsymbol{\mu}^{[k]} = \frac{\sum_{i=1}^{N} w_i \boldsymbol{\tau}_i}{\sum_{i=1}^{N} w_i}, \tag{5}$$

$$\boldsymbol{\Sigma}^{[k]} = \frac{\sum_{i=1}^{N} w_i (\boldsymbol{\tau}_i - \boldsymbol{\mu}^{[k+1]})(\boldsymbol{\tau}_i - \boldsymbol{\mu}^{k+1})^T}{z}, \tag{6}$$

$$z = \frac{\left(\sum_{i=1}^{N} w_i\right)^2 - \sum_{i=1}^{N} w_i^2}{\sum_{i=1}^{N} w_i}.$$

Here, the weightes w_i are defined by the update rule (4)

$$w_i = \exp\left(\frac{R(\boldsymbol{\tau}_i)}{B+\eta^*}\right) \left(\frac{d(\boldsymbol{\tau}_i;\boldsymbol{\Theta}_D)}{p(\boldsymbol{\tau}_i;\boldsymbol{\Theta}^{[k-1]})}\right)^{\frac{B}{B+\eta^*}} \qquad i = 1...N,$$

where η^* minimizes the dualfunction (13).

After the convergence of the optimization to a locally optimal policy p^*, we can connect p^* to any probabilistic movement primitive representation. This connection is straightforward as the proposed optimization outputs not only a single trajectory but a distribution over trajectories.

Algorithm 1 describes the proposed optimization of a trajectory distribution in pseudo code. In line 2 we initialize the mean $\boldsymbol{\mu}^{[0]}$ with the mean of the demonstrations and the

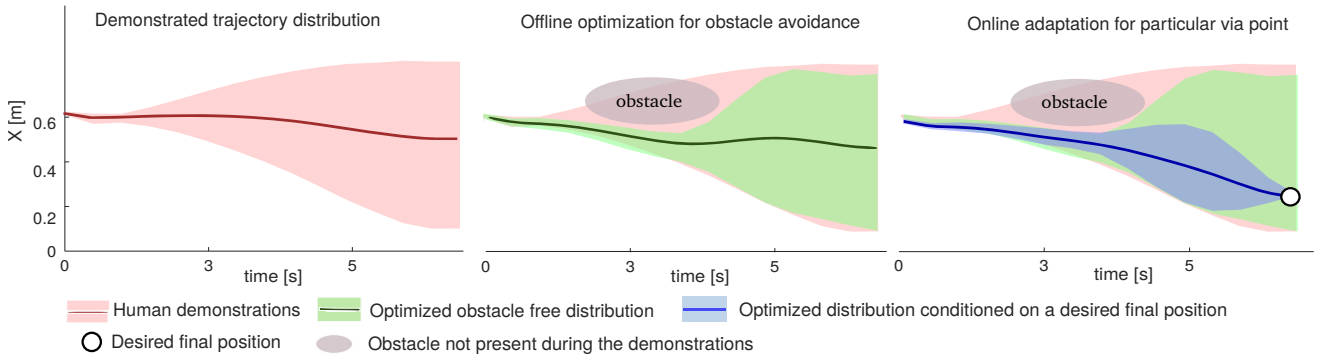

Fig. 3. The distribution over trajectories (illustrated as mean and two times standard deviation) is shown at different stages of the algorithm. The distribution of demonstrated trajectories (red) is optimized for obstacles not present during the demonstrations (green). Using a probabilistic representation the optimized distribution can be quickly adapted for task-specific constraints such as via points (blue).

covariance matrix $\boldsymbol{\Sigma}^{[0]}$ with the diagonal from the covariance matrix of the demonstrations, enlarged by a factor α. Using the diagonal of the covariance matrix of the demonstrations provides exploration noise that is on the same order of magnitude of the human variance and the independence among time steps gives the trajectory the flexibility to deviate from obstacles. While we noticed that this heuristic provides good convergence, an arbitrary initialization is also possible as the objective in (1) incorporates the full correlation of the demonstrations during the optimization in either case.

The proposed optimization can be applied for multiple trajectory dimensions, either independently parallelized or coupled over the obstacle cost (2).

B. Online Adaptation to Task Constraints

As our proposed optimization results in an optimized distribution of trajectories, the robot motion can be easily encoded by statistical modeling methods. For the experiments in this paper we used the ProMP method [8], but other representations such as [7], [6] or [5] can also be considered.

Only the basic concepts are mentioned here, details of the ProMP method can be found in [8].

Algorithm 1 Demonstration Based Trajectory Optimization

procedure DEBATO($\boldsymbol{\mu}_D$, $\boldsymbol{\Sigma}_D$, B)
  $\boldsymbol{\mu}$, $\boldsymbol{\Sigma} \leftarrow$ initialize (e.g as $\boldsymbol{\mu}_D$ and diag($\boldsymbol{\Sigma}_D$)+αI)
  repeat
    for i = 1 : N **do**
      $\boldsymbol{\tau}_i \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
      $\boldsymbol{l}_i$=log($\mathcal{N}(\boldsymbol{\tau}_i, \boldsymbol{\mu}, \boldsymbol{\Sigma})$)
      $\tilde{\boldsymbol{l}}_i$=log($\mathcal{N}(\boldsymbol{\tau}_i, \boldsymbol{\mu}_D, \boldsymbol{\Sigma}_D)$)
      $\boldsymbol{r}$(i) = R($\boldsymbol{\tau}_i$) Eq.(2)
    $\eta_{\text{opt}} \leftarrow$ minimize dualfunction Eq.(13)
    for i = 1:N **do**
      $\boldsymbol{w}$(i)=exp($\frac{1}{B+\eta_{\text{opt}}}$(($\boldsymbol{r}_i$)+B ($\tilde{\boldsymbol{l}}_i$-$\boldsymbol{l}_i$))
    $\boldsymbol{\mu} \leftarrow$ ML update($\boldsymbol{w}$,$\boldsymbol{\tau}$) Eq.(5)
    $\boldsymbol{\Sigma} \leftarrow$ ML update($\boldsymbol{\mu}$,$\boldsymbol{w}$,$\boldsymbol{\tau}$) Eq.(6)
  until sufficient convergence of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$
return $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$

A set of trajectories, obtained from a certain number of demonstration trials, is modeled using a set of basis functions $\boldsymbol{\Psi_t}$ and a weight vector $\boldsymbol{w}$

$$\boldsymbol{y}_t = \begin{bmatrix} \boldsymbol{x}_t \\ \dot{\boldsymbol{x}}_t \end{bmatrix} = \boldsymbol{\Psi}_t \boldsymbol{w} + \boldsymbol{\epsilon}_y; \qquad \text{with } \boldsymbol{\epsilon}_y \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_y). \quad (7)$$

In this equation $\boldsymbol{\Psi}_t = [\boldsymbol{\psi}_t, \dot{\boldsymbol{\psi}}_t]$ denotes the time-dependent basis matrix with positions $\boldsymbol{x}_t$ and velocities $\dot{\boldsymbol{x}}_t$ and ϵ_y defines zero-mean i.i.d. Gaussian noise.

To learn the weights of the basis functions linear ridge regression is used on multiple trajectories sampled from p^*

$$\boldsymbol{w} = (\boldsymbol{\Psi}^T \boldsymbol{\Psi} + \lambda \boldsymbol{I})^{-1} \boldsymbol{\Psi}_t^T \boldsymbol{y}, \quad (8)$$

where λ denotes a small regularization factor.

Encoding a movement with the learned weight vector $\boldsymbol{w}$, the probability distribution of a trajectory can be written as

$$p(\boldsymbol{y}|\boldsymbol{w}) = \prod_t \mathcal{N}(\boldsymbol{y}_t|\boldsymbol{\Psi}_t \boldsymbol{w}, \boldsymbol{\Sigma}_y). \quad (9)$$

To obtain a probability distribution over multiple trajectories the respective weight vectors are represented as a Gaussian distribution with mean $\boldsymbol{\mu}_w$ and covariance matrix $\boldsymbol{\Sigma}_w$.

The framework offers fast adaptation of the motion to changing via points by conditioning on a certain state $\boldsymbol{y}_t^*$ at time point t. Therefore, an observation $\boldsymbol{o}_t^* = \boldsymbol{y}_t^*, \boldsymbol{\Sigma}_y^*$, where $\boldsymbol{\Sigma}_y^*$ can be seen as the observation noise, is used to obtain a posterior distribution over the weights. This posterior is also Gaussian with mean $\boldsymbol{\mu}_w^{\text{new}}$ and covariance matrix $\boldsymbol{\Sigma}_w^{\text{new}}$

$$\boldsymbol{K} = \boldsymbol{\Sigma}_w \boldsymbol{\Psi}_t (\boldsymbol{\Sigma}_y^* + \boldsymbol{\Psi}_t^T \boldsymbol{\Sigma}_w \boldsymbol{\Psi}_t)^{-1} \quad (10)$$

$$\boldsymbol{\mu}_w^{\text{new}} = \boldsymbol{\mu}_w + \boldsymbol{K}(\boldsymbol{y}_t^* - \boldsymbol{\Psi}_t^T \boldsymbol{\mu}_w) \quad (11)$$

$$\boldsymbol{\Sigma}_w^{\text{new}} = \boldsymbol{\Sigma}_w - \boldsymbol{K} \boldsymbol{\Psi}_t^T \boldsymbol{\Sigma}_w. \quad (12)$$

Figure 3 summarizes the steps performed on the distribution of human demonstrations. It illustrates the original distribution (red), the optimized distribution after the demonstration based trajectory optimization (green), and the distribution of the probabilistic representation (blue), which can be reused e.g. for conditioning on a certain end point. The colors correspond to the steps, illustrated in Figure 2.

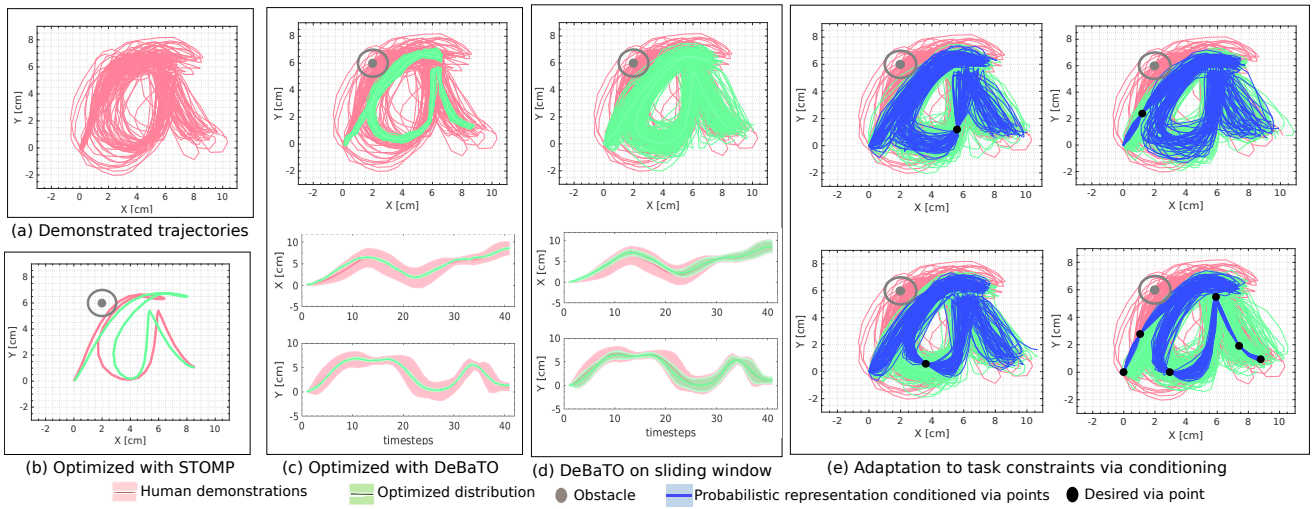

Fig. 4. The demonstrations are given as multiple drawings of a handwritten letter (a). Unlike other trajectory optimization methods for motion planning, such as STOMP (b), the proposed demonstration based trajectory optimization (c) preserves a distribution over trajectories, rather than a single optimized trajectory. Applying the optimization on segments of the trajectories (d) results in deformations of the trajectory parts in the vicinity of the obstacle, but preserves the variance of the demonstrations in the other parts. Statistical modeling methods for movement representation (e) offer fast adaptation by conditioning on different desired via points.

IV. EXPERIMENTS

This section presents the results of experiments, in which we evaluate a 2D example of redrawing a handwritten letter and a pick-and-place setup for a 7-DoF robot arm.

A. Drawing a Letter in a Constrained Workspace

In the first experiment the demonstrations are given as a set of multiple drawings of a handwritten letter "a". As illustrated by Figure 4(a), the demonstrations include variance and can be represented as a Gaussian distribution as explained in Section III. The algorithm attempts to reproduce this demonstrated distribution, however, with an obstacle placed at different locations.

Figure 4(c) presents a representative result of our method on this 2D example. We compared this result with the result of a standard trajectory optimization method, here based on STOMP, a stochastic trajectory optimization method for motion planning [2], depicted in Figure 4(b). The figures illustrate that our demonstration based trajectory optimization is able to operate with the full distribution as an input, and preserves a distribution during the optimization, whereas a standard trajectory optimization reduces the final solution to a single trajectory. In terms of further generalization of trajectories this means that the solution of STOMP is specific to the defined start and end point, whereas our solution provides variance.

In Figure 4 (c), note that the optimization achieves obstacle avoidance and captures the correlation of the demonstrations, but the variance of the original distribution is reduced even in areas not effected by the obstacle. This result stems from the fact that the demonstrated trajectories are strongly correlated in the temporal axis, and therefore, local deviations, for example to deviate from an obstacle at the beginning of the motion, propagates until the end of the trajectory. In certain applications, however, it may be desirable to preserve the full variance in areas not affected by the obstacle. One alternative is to assume that the demonstrated trajectories are not correlated in time, as it was done in [29]. However, this may generate excessively jerky trajectories. A compromise between full correlation and no correlation is to run (3) on a sliding window along the time axis, using only the corresponding segment of the demonstrations for each window. To improve efficiency, this process can be computed in parallel. The solution of each segment must be reconnected afterwards, to generate the whole trajectory distribution.

Figure 4(d) shows a result for this optimization of trajectory segments. The plots reveal that with this approach the demonstrated distribution (red) gets deformed only in regions in direct vicinity to the obstacle, and the optimized solution (green) still contains the full variance of the demonstrations in the other parts of the trajectories. For this experiment, segments with 1/5 of the whole trajectory length were used. Automating this procedure, either by adaptive tuning of the window size, or by decreasing the correlation of the demonstrations only at the areas that need to avoid the obstacle, is part of future research.

B. Adaptation For Different Via Points

As the proposed method results in a distribution over trajectories, it can be used in combination with statistical modeling methods for online adaptation to task constraints, such as via points.

We encode the optimized distribution as a probabilistic movement primitive and condition on different desired via points as explained in Section III-B. The results in Figure 4(e) illustrate the demonstrations (red), the optimized solution (green) and the remaining distribution after conditioning with the probabilistic movement representation (blue).

Fig. 5. We consider a pick-and-place scenario with a ball and five boxes as illustrated on the left. Human demonstrations, as shown on the right, are recorded using an OptiTrack motion capturing system for the wrist position. The proposed algorithm is evaluated on a 7DoF Kuka Lightweight arm to achieve robot motions that are similar to the demonstrated motions, while generalizing to different work space settings, such as obstacles, not present during the original demonstrations, and deliveries to different boxes.

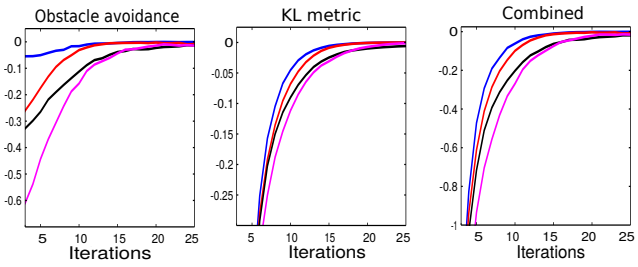

Fig. 6. The plots illustrate the average reward over 300 sampled trajectories during the demonstration based trajectory optimization. The two objectives of the optimization, obstacle avoidance and minimal deviation from the demonstrated distribution, are shown, as well as the combined reward. Each curve represents a different obstacle setting. The results reveal how the optimization converges to a distribution of trajectories that avoids obstacles and at the same time deviates as least as possible from the original demonstrations.

The plots reveal that a core advantage of optimizing a distribution over trajectories, as proposed in our demonstration based trajectory optimization, is given by the possibility to reuse the optimized obstacle-free distribution for adaptation to constraints that need not necessarily be known during the optimization process. This conditioning step is very fast and can be used for online variation of the written letters, while at the same time the prior optimization accounts for obstacle avoidance in the current workspace setting.

C. Pick-And-Place Scenario on a Humanoid

In this experiment, we report our initial evaluations of the method on a real robot task, using a 7-DoF KUKA lightweight arm as illustrated in Figure 5.For those experiments on a real robot we performed the optimization in task space and only considered the end effector position for obstacle avoidance. Evaluation of optimizing robot trajectories in joint space and approximation of the full arm geometry with a bounding-box approach, as well as consideration of self-collisions, are planned as a next step.

The experiment consisted of picking a ball, at a fixed location “6” on a table and placing it inside one of five possible boxes, shown in Figure 5 on the left. To collect demonstrations, a human performed multiple deliveries of the ball to the boxes “1”, “3”, and “5” (a sequence of snapshots is shown in Figure 5(right)). There are no demonstrations for box “2” and “4”.

Before creating a probabilistic primitive, however, we added a large box at different locations in the very middle of the table, as shown for one example in Figure 8. Using a Kinect camera to obtain a signed distance field with Euclidean distances to the obstacles, we optimized the distribution of human demonstrations with respect to the changed workspace. The experiment was repeated with the obstacle on a different position, leading to different contextualized trajectories, such as the one shown at the bottom of Figure 7.

Figure 6 shows the reward improvement during the optimization as a function of the iterations for four different obstacle positions. For both objectives the trajectory optimization converges steadily and achieves a trade-off between minimal deviation from the original demonstrations and obstacle avoidance. Even though the KL metric indicates that the optimized distribution stays close the demonstrations and is therefore ”human-like”, a more qualitative analysis of human-specific motion criteria remains an interesting aspect for future work.

For the reasons explained in Section IV-A we performed the optimization on a sliding window, with 1/5 of the full trajectory length, along the time axis. Subsequently we encode the optimized distributions over trajectories as probabilistic movement primitives. Conditioning on via points for those movement primitives allows to generalize the robot’s motions to deliver the ball to different desired box locations, even to the boxes “2” and “4”, that were not part of the demonstrations. Figure 8 shows resulting robot motions for one obstacle position, and illustrates deliveries to box “1” and box “4”.

Using the proposed algorithm the main planning time is needed in the offline optimization, whereas the average trajectory planning time for different box locations in the online phase is significantly faster, as illustrated in Table I for four different obstacle settings, computed on a core i7 computer.

TABLE I
ONLINE/OFFLINE PLANNING TIME

	Setting 1	Setting 2	Setting 3	Setting 4
Offline Optimization	71.97s	73.64s	72.09s	71.01s
Avg. Online Planning	0.20s	0.24s	0.18s	0.22s

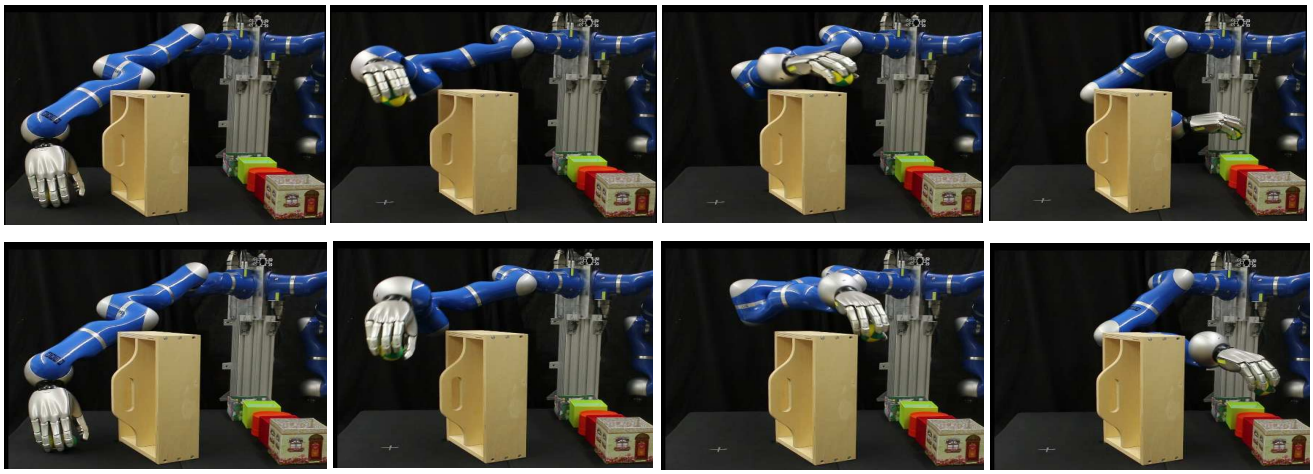

Fig. 8. The proposed algorithm optimizes the demonstrated trajectories with respect to obstacles, not present during the original demonstrations. Encoding the optimized distribution over trajectories as a probabilistic movement primitive allows for fast online adaptation to different box locations. Not only deliveries to boxes demonstrated by the human, such as box "1" (top row), can successfully be performed by the robot, but it is also possible to generalize for boxes, that were not included in the demonstrations, such as box "4" (bottom row).

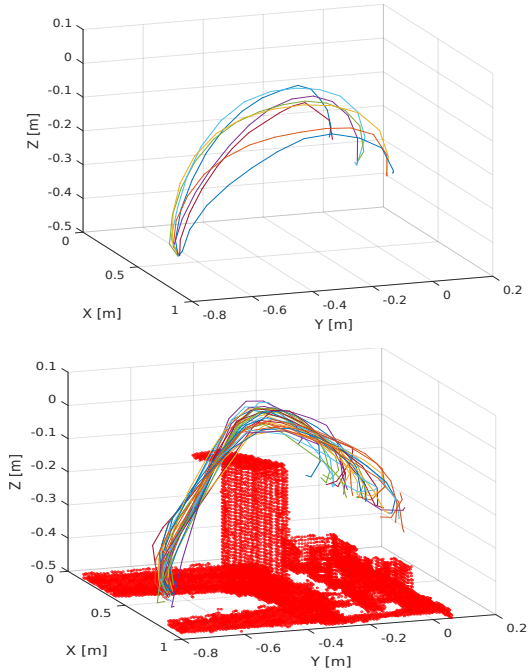

Fig. 7. (Top) Human demonstrations are recorded as multiple trajectories. The demonstrations include trajectories for boxes "1", "3" and "5". Each of those boxes was demonstrated three times to include variance. (Bottom) During the optimization our method preserves correlations of the original distribution such that solutions show small deviation to the demonstrated trajectories but at the same time avoid obstacles not present during the original demonstrations. To determine Euclidean distances to the obstacles we use a signed distance field (red), obtained from point clouds of a Kinect camera.

V. CONCLUSION AND FUTURE WORK

We introduced a method for demonstration based trajectory optimization that offers adaptation of a demonstrated distribution over human trajectories to changed robot workspaces. The proposed method accounts for both, minimal deviation from a demonstrated distribution and avoidance of static obstacles, not present during the original demonstrations. Straightforward connection to statistical modeling methods enables fast online adaptation of the optimized distribution to task specific constraints, such as changing via or goal points.

In a pick-and-place scenario on a 7-DoF robot arm we showed the algorithms suitability to create generalizable robot motions from human demonstrations. We believe our method is in particular suited for human-robot collaboration, since the motion of the robot naturally resembles the motions of the human demonstrator. In future work it could be beneficial to incorporate the obstacle position directly as task constraints to reduce offline optimization time.

An interesting extension of the proposed method could be to go beyond pure trajectory optimization and incorporate more information inferred out of the demonstrations. Extracting intentions out of human trajectories could provide a higher level for planning inside the method. In terms of generalization it would then be beneficial to transfer learned motions to tasks with similar intentions.

APPENDIX

We obtain a closed form solution of (3) with the method of Lagrange multipliers. The Lagrangian $L(p,\eta,\lambda)$ of the optimization problem in (3) is given by

$$L(p,\eta,\lambda)=\int_{\boldsymbol{\tau}}\left[p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})R(\boldsymbol{\tau})-Bp(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})\log\left(\frac{p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})}{d(\boldsymbol{\tau};\boldsymbol{\Theta}_D)}\right)\right]\mathrm{d}\boldsymbol{\tau}$$
$$+\eta\left[\varepsilon-\int_{\boldsymbol{\tau}}p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})\log\left(\frac{p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})}{p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})}\right)\mathrm{d}\boldsymbol{\tau}\right]+\lambda\left(1-\int_{\boldsymbol{\tau}}p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})\mathrm{d}\boldsymbol{\tau}\right)$$
$$\text{s.t. } \eta\geq 0$$

where η and λ denote the Lagrange multipliers.

In order to obtain the optimal new policy the Lagrangian is differentiated with respect to $p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})$. Setting the deriva-

tive to zero results in

$$p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]}) = \exp\left(\frac{R(\boldsymbol{\tau})-B-\eta-\lambda}{B+\eta}\right) d(\boldsymbol{\tau};\boldsymbol{\Theta}_D)^{\frac{B}{B+\eta}} p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})^{\frac{\eta}{B+\eta}}.$$

By constraining the probabilities $p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]})$ to still sum to one, we obtain

$$\exp\left(\frac{-B-\eta-\lambda}{B+\eta}\right) = \frac{1}{\int_{\boldsymbol{\tau}} \exp\left(\frac{R(\boldsymbol{\tau})}{B+\eta}\right) d(\boldsymbol{\tau};\boldsymbol{\Theta}_D)^{\frac{B}{B+\eta}} p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})^{\frac{\eta}{B+\eta}} \mathrm{d}\boldsymbol{\tau}}.$$

Inserting this in a rewritten form of the Lagrangian the dualfunction can be computed

$$D(\boldsymbol{\tau},\eta) = \eta\epsilon + (B+\eta)\log\left(\int_{\boldsymbol{\tau}} \exp\left(\frac{R(\boldsymbol{\tau})}{B+\eta}\right) d(\boldsymbol{\tau};\boldsymbol{\Theta}_D)^{\frac{B}{B+\eta}} p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})^{\frac{\eta}{B+\eta}} \mathrm{d}\boldsymbol{\tau}\right). \quad (13)$$

Rewriting the dualfunction results in

$$D(\boldsymbol{\tau},\eta) = \eta\epsilon + (B+\eta)\log\left(\int_{\boldsymbol{\tau}} p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]}) \exp\left(\frac{R(\boldsymbol{\tau})}{B+\eta}\right)\left(\frac{d(\boldsymbol{\tau};\boldsymbol{\Theta}_D)}{p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})}\right)^{\frac{B}{B+\eta}} \mathrm{d}\boldsymbol{\tau}\right).$$

We approximate the dualfunction using N samples $\boldsymbol{\tau}_1, ..., \boldsymbol{\tau}_{\boldsymbol{N}}$ obtained from the old policy $p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})$

$$D(\boldsymbol{\tau},\eta) = \eta\epsilon + (B+\eta)\log\left(\frac{1}{N}\sum_{i=1}^{\mathrm{N}} \exp\left(\frac{R(\boldsymbol{\tau}_i)}{B+\eta}\right)\left(\frac{d(\boldsymbol{\tau}_i;\boldsymbol{\Theta}_D)}{p(\boldsymbol{\tau}_i;\boldsymbol{\Theta}^{[k-1]})}\right)^{\frac{B}{B+\eta}}\right).$$

Finally, computing a minimal $\eta = \eta^*$ from the dualfunction results in the update rule for the new policy

$$p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k]}) = \frac{\exp\left(\frac{R(\boldsymbol{\tau})}{B+\eta^*}\right)\left(\frac{d(\boldsymbol{\tau};\boldsymbol{\Theta}_D)}{p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})}\right)^{\frac{B}{B+\eta^*}} p(\boldsymbol{\tau};\boldsymbol{\Theta}^{[k-1]})}{\frac{1}{N}\sum_{i=1}^{\mathrm{N}} \exp\left(\frac{R(\boldsymbol{\tau}_i)}{B+\eta^*}\right)\left(\frac{d(\boldsymbol{\tau}_i;\boldsymbol{\Theta}_D)}{p(\boldsymbol{\tau}_i;\boldsymbol{\Theta}^{[k-1]})}\right)^{\frac{B}{B+\eta^*}}}.$$

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community's Seventh Framework Programmes (FP7-ICT-2013-10) under grant agreement 610878 (3rdHand) and from the European Union's Horizon 2020 research and innovation programme under grant agreement 640554 (SKILLS4ROBOTS).

REFERENCES

[1] E. Theodorou, J. Buchli, and S. Schaal, "Reinforcement learning of motor skills in high dimensions: A path integral approach," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 2397–2403.

[2] M. Kalakrishnan, S. Chitta, E. Theodorou, P. Pastor, and S. Schaal, "Stomp: Stochastic trajectory optimization for motion planning," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 4569–4574.

[3] K. Darling, "'who's johnny?' anthropomorphic framing in human-robot interaction, integration, and policy," *Anthropomorphic Framing in Human-Robot Interaction, Integration, and Policy*, 2015.

[4] S. Schaal, "Is imitation learning the route to humanoid robots?" *Trends in cognitive sciences*, vol. 3, no. 6, pp. 233–242, 1999.

[5] T. Inamura, I. Toshima, H. Tanie, and Y. Nakamura, "Embodied symbol emergence based on mimesis theory," *The International Journal of Robotics Research*, vol. 23, no. 4-5, pp. 363–377, 2004.

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

[7] S. Dong and B. Williams, "Motion learning in variable environments using probabilistic flow tubes," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1976–1981.

[8] A. Paraschos, C. Daniel, J. R. Peters, and G. Neumann, "Probabilistic movement primitives," in *Advances in neural information processing systems*, 2013, pp. 2616–2624.

[9] 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.

[10] S. Schaal, A. Ijspeert, and A. Billard, "Computational approaches to motor learning by imitation," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 358, no. 1431, pp. 537–547, 2003.

[11] M. N. Nicolescu and M. J. Mataric, "Natural methods for robot task learning: Instructive demonstrations, generalization and practice," in *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*. ACM, 2003, pp. 241–248.

[12] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[13] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.

[14] S. M. LaValle and J. J. Kuffner Jr, "Rapidly-exploring random trees: Progress and prospects," in *Workshop on the Algorithmic Foundations of Robotics*, 2000.

[15] Y. K. Hwang and N. Ahuja, "Gross motion planning - a survey," *ACM Computing Surveys (CSUR)*, vol. 24, no. 3, pp. 219–291, 1992.

[16] Z. Marinho, A. Dragan, A. Byravan, B. Boots, S. Srinivasa, and G. Gordon, "Functional gradient motion planning in reproducing kernel hilbert spaces," *arXiv preprint arXiv:1601.03648*, 2016.

[17] A. Coates, P. Abbeel, and A. Y. Ng, "Learning for control from multiple demonstrations," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 144–151.

[18] G. Ye and R. Alterovitz, "Demonstration-guided motion planning," in *International symposium on robotics research (ISRR)*, vol. 5, 2011.

[19] J. Mainprice and D. Berenson, "Human-robot collaborative manipulation planning using early prediction of human motion," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 299–306.

[20] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Robotics and Automation (ICRA), 2002 IEEE International Conference on*, vol. 2. IEEE, 2002, pp. 1398–1403.

[21] J. Kober, K. Mülling, O. Krömer, C. H. Lampert, B. Schölkopf, and J. Peters, "Movement templates for learning of hitting and batting," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 853–858.

[22] D. Lee and Y. Nakamura, "Mimesis model from partial observations for a humanoid robot," *The International Journal of Robotics Research*, vol. 29, no. 1, pp. 60–80, 2010.

[23] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Learning attractor landscapes for learning motor primitives," Tech. Rep., 2002.

[24] F. Stulp, E. Oztop, P. Pastor, M. Beetz, and S. Schaal, "Compact models of motor primitive variations for predictable reaching and obstacle avoidance," in *2009 9th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2009, pp. 589–595.

[25] H. Hoffmann, P. Pastor, D.-H. Park, and S. Schaal, "Biologically-inspired dynamical systems for movement generation: automatic real-time goal adaptation and obstacle avoidance," in *Robotics and Automation (ICRA), 2009 IEEE International Conference on*. IEEE, 2009, pp. 2587–2592.

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

[27] S. Calinon, "A tutorial on task-parameterized movement learning and retrieval," *Intelligent Service Robotics*, vol. 9, no. 1, pp. 1–29, 2016.

[28] A. Bry and N. Roy, "Rapidly-exploring random belief trees for motion planning under uncertainty," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 723–730.

[29] P. Englert, A. Paraschos, M. P. Deisenroth, and J. Peters, "Probabilistic model-based imitation learning," *Adaptive Behavior*, vol. 21, no. 5, pp. 388–403, 2013.

[30] M. P. Deisenroth, G. Neumann, J. Peters, *et al.*, "A survey on policy search for robotics." *Foundations and Trends in Robotics*, vol. 2, no. 1-2, pp. 1–142, 2013.