2016 IEEE-RAS 16th International Conference on
Humanoid Robots (Humanoids)
Cancun, Mexico, Nov 15-17, 2016

Trajectory Learning from Demonstration with Canal Surfaces: A Parameter-free Approach

S. Reza Ahmadzadeh[1], Roshni Kaushik[2], and Sonia Chernova[1]

Abstract— **We present a novel geometric framework for intuitively encoding and learning a wide range of trajectory-based skills from human demonstrations. Our approach identifies and extracts the main characteristics of the demonstrated skill, which are spatial correlations across different demonstrations. Using the extracted characteristics, the proposed approach generates a continuous representation of the skill based on the concept of canal surfaces. Canal surfaces are Euclidean surfaces formed as the envelope of a family of regular surfaces (e.g. spheres) whose centers lie on a space curve. The learned skill can be reproduced, as a time-independent trajectory, and generalized to unforeseen situations inside the canal while its main characteristics are preserved. The main advantages of the proposed approach include: (a) requiring no parameter tuning, (b) maintaining the main characteristics and implicit boundaries of the skill, and (c) generalizing the learned skill over the initial condition of the movement, while exploiting the whole demonstration space to reproduce a variety of successful movements. Evaluations using simulated and real-world data exemplify the feasibility and robustness of our approach.**

I. INTRODUCTION

The main goal of Learning from Demonstration (LfD) approaches is to reduce the need for manual robot programming [1]. These approaches should ultimately enable even non-experts to teach new skills to robots interactively through demonstrations. Using a set of successful examples of a skill performed by a human teacher, LfD techniques encode the skill in a model that its representation varies among different approaches. The robot then should be able to employ the model to generalize the skill to novel situations autonomously. However, it has been shown that current robotic platforms are not good at being autonomous and need human assistance constantly [2]. The results from the recent DARPA Robotics Challenge (DRC) highlight that despite the existence of trajectory-based LfD techniques, the vast majority of the robot motions executed during the challenge were hand-coded or teleoperated. The main factors that make such approaches highly impractical (especially for non-experts) include the number of parameters to be tuned and their corresponding tuning methods. In addition, the tuning process requires that the user understands how the system reacts to such adjustments. So the level of complexity of the existing representations can be counted as an important factor that prompts a need for an efficient yet simpler approach.

To address these issues, we propose a novel parameter-free LfD approach based on Differential Geometry that enables robots to acquire novel trajectory-based skills through demonstrations. This approach encodes the demonstrated skill as a geometric model. The model is composed of a regular curve and a surface in Cartesian space called a *Canal Surface* [3]. The constructed canal surface extracts and represents the main characteristics of the skill, which are the spatial correlations across different demonstrations that are not specified by the user explicitly. The proposed representation is visually perceivable and can reproduce the learned skill as a time-independent trajectory using a simple geometric rule. The skill can be generalized to unforeseen situations inside the canal while the main characteristics of the motion are preserved. The proposed approach: *a*) requires no parameter tuning, *b*) maintains the important characteristics and implicit boundaries of the skill, *c*) generalizes the learned skill over the initial condition of the movement, while exploiting the whole demonstration space to reproduce a variety of successful movements.

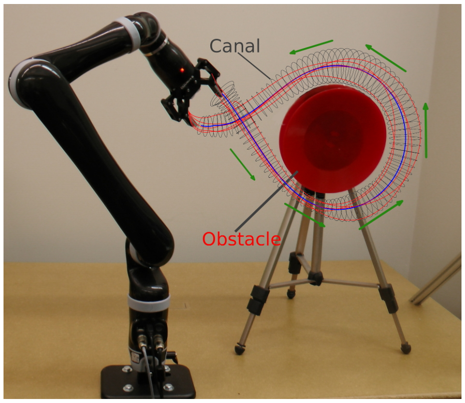


Fig. 1: A Jaco2 arm learns a circular movement in task-space while avoiding an obstacle. Five demonstrations (red curves) are captured through kinesthetic teaching and the skill is encoded as a canal surface (gray). Arrows show the direction of the movement.

II. RELATED WORK

In this work, we focus on learning, representing, and generalizing of trajectory-based skills. Therefore, we review related work on trajectory-based skill learning from demonstrations. Over the past two decades, several trajectory-based LfD approaches have been developed [1]. These approaches use a variety of techniques to encode a demonstrated skill and retrieve a generalized form of the trajectory. Many of them use regression-based techniques to represent the given set of demonstrations using a probabilistic representation [4], [5]. One of the most well-known works in this group is by Calinon *et al.,* [6], which builds a probabilistic representation of the demonstrations using Gaussian Mixture Model

*This work is supported in part by the ONR award N000141410795 and in part by NSF SURE Robotics REU site program award EEC-1263049.

[1] School of Interactive Computing, Georgia Institute of Technology, {reza.ahmadzadeh, chernova}@gatech.edu

[2] Rice University, roshni.s.kaushik@gmail.com

978-1-5090-4717-8/16/$31.00 ©2016 IEEE

(GMM) and retrieves a smooth trajectory using Gaussian Mixture Regression (GMR). Similarly, Gaussian Process (GP) Regression generalizes over a set of demonstrated trajectories [5], however, GP is computationally expensive. To address this issue local Gaussian process regression was proposed to extract the constraints of a demonstrated skill [7]. Another approach similar to GP, called *LfD by Averaging Trajectories* (LAT), uses only one-dimensional normal distributions [8]. Both GP and LAT cannot extract constraints from the demonstrations with objects aligned parallel to a Cartesian coordinate axis. GMM/GMR, GP, local GP, and LAT are time-dependent approaches that require an explicit time-indexing and are useful for encoding skills that are deemed to be performed in a fixed amount of time. The skills learned using time-independent approaches, on the other hand, can be temporally scaled [9]. Another disadvantage of these approaches is that they require parameter tuning (e.g. number of Gaussian components, scale, weight, kernel).

Ijspreet *et al.,* showed that dynamical systems can also be used to encode and reproduce trajectories [10]. Dynamic Movement Primitives (DMPs) represent demonstrated trajectories as movements of a particle subject to a set of spring-damper systems perturbed by an external force. The shape of the movement is approximated using Gaussian basis functions and the weights are calculated using locally weighted regression. DMPs are implicitly time-dependent and this makes the system sensitive to temporal perturbations. In addition, using DMPs, one has to tune parameters of the dynamical systems, such as time constants and scaling factors.

To tackle the problem of real-time motion planning, Majumdar and Tedrake extended the idea of elastic bands [11] and proposed an approach to approximate a boundary around a trajectory, which can be visualized as a funnel [12]. Their approach computes a library of funnels and their corresponding open-loop controllers off-line, and then uses a closed-loop system to generate trajectories from the library in real-time that can deal with obstacles. The computed funnels illustrate a similar representation to the proposed approach in this paper. However, our approach does not require extensive off-line computation. To reflect a human's intention, Dong and Williams proposed an approach called probabilistic flow tubes that represents continuous actions (i.e. trajectories) by extracting covariance data [13]. The learned flow tube consists of a spine trajectory and 2D covariance data at each corresponding time-step. Their representation can be seen as a special case of our approach in which the cross-section is formed using covariance data.

III. METHODOLOGY

The concept of *canal surfaces*, first introduced by Hilbert and Cohn-Vossen [3], plays a fundamental role in descriptive geometry. They are also known as Generalized Cylinders, which can be explained as a representation of an elongated object composed of a spine and a smoothly varying cross-section [14]. In the context of Computer Aided Graphic Design (CAGD), canal surfaces are used for shape reconstruction, construction of smooth blending surfaces, and transition surfaces between pipes [15]. In Robotics, canal surfaces have been used for finding flyable paths for unmanned aerial vehicles [16], but have not previously been learned.

A. Canal Surfaces

Let Φ_u be the one-parameter pencil[1] of regular implicit surfaces[2] with real-valued parameter u. Two surfaces corresponding to different values of u intersect in a common curve. As u varies, the generated surface is the envelope[3] of the given pencil of surfaces [17]. In 3D space, the envelope can be defined using the following equations:

$$\Phi_u : F(x_1, x_2, x_3, u) = 0, \tag{1}$$

$$\partial F(x_1, x_2, x_3, u)/\partial u = 0, \tag{2}$$

where Φ_u consists of implicit C^2-surfaces which are at least twice continuously differentiable.

Definition: A canal surface, $\mathscr{C}_u$, is an envelope of the one-parameter pencil of spheres and can be written as

$$\mathscr{C}_u : f(\mathbf{x};u) := \{(\mathbf{c}(u), r(u)) \in \mathbb{R}^{3,1} | u \in \mathbb{R}\}, \tag{3}$$

where the spheres are centered on a regular curve $\Gamma : \mathbf{x} = \mathbf{c}(u) \in \mathbb{R}^3$ in Cartesian space, known as the *spine curve* or *directrix*. The radius of the spheres are given by the function $r(u) \in \mathbb{R}$, called *radii*, which is a C^1-function. The non-degeneracy condition is satisfied by assuming $r > 0$ and $|\dot{r}| < \|\dot{\mathbf{c}}\|$ [15]. For the one-parameter pencil of **spheres**, Equation (3) can be written as

$$\mathscr{C}_u : f(\mathbf{x};u) := (\mathbf{x} - \mathbf{c}(u))^2 - r(u)^2 = 0. \tag{4}$$

An example of such surfaces is depicted in Fig. 2a that has a circular cross-section. Constructing a canal surface by the pencil of **ellipsoids** provides us with canal surfaces with elliptical cross-sections that can be written as follows:

$$\mathscr{C}_u : f(\mathbf{x};u) := \{(\mathbf{c}(u), \mathbf{r}(u)) \in \mathbb{R}^{3,2} | u \in \mathbb{R}\}, \tag{5}$$

where the radii function $\mathbf{r}(u) \in \mathbb{R}^2$ defines the major and minor axes lengths for the ellipses.

B. Parametric Representation

To form a parametric representation of (4), we use an orthonormal frame called Frenet-Serret or TNB. This frame is suitable for describing the kinematic properties of a particle moving along a continuous, differentiable curve in $\mathbb{R}^3$. TNB is composed of three unit vectors $\mathbf{e}_T$, $\mathbf{e}_N$, and $\mathbf{e}_B$, where $\mathbf{e}_T$ is the unit tangent vector, and $\mathbf{e}_N$ and $\mathbf{e}_B$ are the unit normal and unit binormal vectors, respectively. For a non-degenerate directrix curve $\Gamma : \mathbf{x}(s)$, parameterized by its arc-length parameter s, the TNB frame can be defined using the following equations:

$$\mathbf{e}_T = d\mathbf{x}(s)/ds, \tag{6}$$

$$\mathbf{e}_N = \frac{d\mathbf{e}_T}{ds} / \| \frac{d\mathbf{e}_T}{ds} \|, \tag{7}$$

$$\mathbf{e}_B = \mathbf{e}_T \times \mathbf{e}_N. \tag{8}$$

[1] A pencil is a family of geometric objects sharing a common property (e.g. spheres).

[2] An implicit surface is a surface in Euclidean space that can be represented in $F(x(u), y(u), z(u)) = 0$ form.

[3] An envelope is a curve/surface tangent to a family of curves/surfaces (2D or 3D).

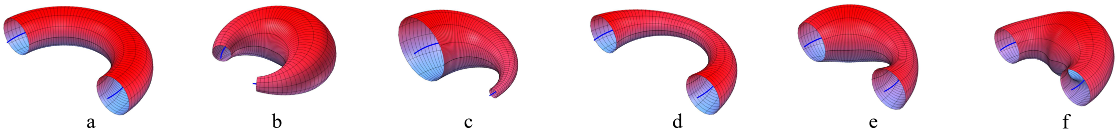


Fig. 2: Six canal surfaces sharing the same directrix with different radii functions. The cross-section of a canal surface can be a circle, an ellipse or generally a closed B-spline. It also can resize and/or reshape smoothly along the directrix.

Applying the second condition of envelopes from (2) to the formula of a canal surface defined by (4) gives

$$\frac{\partial \mathscr{C}_s}{\partial s} = 2(\mathbf{x} - \mathbf{c}(s))\mathbf{e}_T - 2r(s)\frac{dr}{ds} = 0. \tag{9}$$

For each value of s, (9) represents a circle orthogonal to the unit tangent vector of the directrix. Thus the canal surface can be represented as a set of circles formulated as

$$\mathscr{C}_s : f(s,v) = \mathbf{c}(s) + r(s)\left(-\mathbf{e}_T\frac{dr}{ds} + \sqrt{1-(\frac{dr}{ds})^2}\,(\mathbf{e}_B \sin(v) - \mathbf{e}_N \cos(v))\right), \tag{10}$$

where $v \in [0, 2\pi]$ varies along the circle.

Calculating Frenet-Serret frames for real data is prone to noise. The reason is that at some points the derivative vector $\frac{d\mathbf{e}_T}{ds}$ vanishes and the formulae cannot be applied anymore (i.e. $\mathbf{e}_N$ cannot be calculated). This problem can be addressed by calculating the unit normal vector $\mathbf{e}_N$ as the cross product of a random vector by the unit tangent vector $\mathbf{e}_T$.

C. Variations

Equation (10) denotes the parametric representation of a canal surface with a circular cross-section that can vary along the directrix (Fig. 2). As mentioned before, by replacing spheres with ellipsoids we can construct canal surfaces with elliptical cross-sections. In general, the cross-section may not only resize but also reshape from one frame to the next when the TNB frame translates along the directrix. Such variations can be represented using different shapes and techniques such as polygons, polynomials, parabolic blending, cone sections, and B-splines[14]. Such characteristics make canal surfaces a suitable candidate for modeling complicated constraints of trajectory-based skills captured through demonstrations.

IV. Skill Learning Using Canal Surfaces

In this section, we present how the above formulation can be used to represent and reproduce demonstrated robot trajectories. For a given skill, we assume that multiple examples of the motion are demonstrated and captured as a set of trajectories. A variety of demonstration techniques can be used to perform the demonstrations, such as teleoperation and shadowing; in our experiments, we use kinesthetic teaching (see Fig. 3). Given the set of demonstrations, the proposed approach first calculates the directrix (i.e. an average form of the movements) and then extracts the main characteristics of the set (i.e. spatial correlations across demonstrations) and forms the radii function by identifying its boundaries. When the canal surface is constructed, a geometric ratio rule is used for generating new trajectories starting from arbitrary initial poses. In this section, we explain both learning and reproduction phases of the proposed approach in detail. Algorithm 1 shows a pseudo code of the proposed approach.

A. Canal Surface Generation

Consider n different demonstrations of a task are performed and captured in task-space. For each demonstration the 3D Cartesian position of the robot's end-effector is recorded over time as $\hat{\xi}^j = \{\xi_1^j, \xi_2^j, \xi_3^j\} \in \mathbb{R}^{3\times T^j}$, where $j = 1\dots n$ denotes the j^{th} demonstration and T^j is the number of data-points within the trajectory. Since T^j can vary between demonstrations, we use interpolation and resampling in order to gain a frame-by-frame correspondence mapping among the recorded demonstrations and align them temporally. First, a set of piecewise polynomials is obtained using cubic spline interpolation for each demonstration. Then we generate a set of temporally aligned trajectories by resampling from the obtained polynomials. Another advantage of this technique is when the velocity and acceleration data are unavailable, the first and second derivatives of the obtained piecewise polynomials can be used instead. This process provides us with the set of n resampled demonstrations $\xi \in \mathbb{R}^{3\times N\times n}$, each of which consists of N data-points. An alternative widely used solution is employing Dynamic Time Warping [18].

Estimating the Directrix: To estimate the directrix, Γ, we calculate the directional mean value for the given set of demonstrations (Line 4 in Algorithm 1). Let $\mathbf{m} \in \mathbb{R}^{3\times N}$ be the arithmetic mean of ξ. Note that $\mathbf{m}$ is the space curve that all the spheres are centered on to form a canal surface. Alternatively, the directrix can be produced using GMR [6]. In that case, GMR generates the directrix by sampling from the statistical model learned by GMM. However, using GMR requires defining a time vector explicitly.

Estimating the Radii: The next step is to calculate the cross-section of the canal surface at each step s along the directrix. The circumference of a cross-section represents the implicit local constraints of the task (i.e. boundaries) imposed by the set of demonstrations. For instance, when all the trajectories pass through a narrow area, the user is

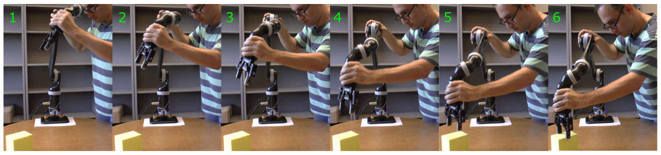


Fig. 3: Kinesthetic demonstration of reaching to an object (yellow box). Captured task-space pose of the end-effector is used as input.

Algorithm 1 Skill Learning using Canal Surfaces

```
1: procedure ENCODING DEMONSTRATIONS
2:     Input: set of $n$ demonstrations $\xi \in \mathbb{R}^{3\times N\times n}$
3:     Output: Canal surface $\mathscr{C}_s$, all TNB frames $\mathscr{F}$
4:     $\mathbf{m}(s) \leftarrow mean(\xi)$
5:     $r(s) \leftarrow boundary(\xi)$
6:     $\mathscr{C}_s, \mathscr{F} \leftarrow makeCanalSurface(\mathbf{m}(s), r(s))$
7: procedure REPRODUCING TRAJECTORY
8:     Input: initial point $\mathbf{p}_0 \in \mathbb{R}^3$, $\mathscr{C}_s$, $\mathscr{F}$
9:     Output: New trajectory $\rho \in \mathbb{R}^{3\times N}$
10:    $\eta \leftarrow \frac{\|\mathbf{p}_0-\mathbf{c}_0\|}{\|\mathbf{g}_0-\mathbf{c}_0\|}$
11:    $\mathbf{p}_i \leftarrow \mathbf{p}_0$   ,   $\rho \leftarrow \mathbf{p}_0$
12:    for each $frame \mathscr{F}_i$ do
13:        $\mathbf{p}_{i+1} \leftarrow project(\mathbf{p}_i, \eta, \mathscr{F}_{i+1}, \mathscr{F}_i)$
14:        $\rho \leftarrow \mathbf{p}_{i+1}$
15:        $i \leftarrow i+1$
```

Algorithm 2 Generating a canal surface (circular cross-section)

```
1: procedure makeCanalSurface
2:     Input: directrix $\mathbf{m}(s) \in \mathbb{R}^{3\times N}$, radii $r(s) \in \mathbb{R}^{1\times N}$
3:     Output: Canal surface $\mathscr{C}_s$, all TNB frames $\mathscr{F}$
4:     $v \leftarrow 0 : 2\pi$
5:     for each $\mathbf{m}_i \in \mathbf{m}(s)$ and $r_i \in r(s)$ do
6:         $\{\mathbf{e}^i_T, \mathbf{e}^i_N, \mathbf{e}^i_B\} \leftarrow estimateTNBframe(\mathbf{m}_i)$
7:         $\mathscr{F} \leftarrow \{\mathbf{e}^i_T, \mathbf{e}^i_N, \mathbf{e}^i_B\}$
8:         $\mathscr{C}_s \leftarrow \mathbf{m}_i + r_i\left(\mathbf{e}^i_N\cos(v) + \mathbf{e}^i_B\sin(v)\right)$
```

emphasizing that the movement in that specific area has to be constrained. Thus, the radii function represents both the boundaries of the demonstrated task and the constraints applied from surrounding objects and the environment.

For canal surfaces with circular cross-sections, the radius at a given point on the directrix can be calculated by measuring the distances from that point to the corresponding points on the demonstrated trajectories and using the maximum value. The estimated circle bounds other corresponding points as well (Fig. 4a). The calculated radii along the directrix form the radii function $r(s)$ (Line 5). We use (Line 6) the estimated mean and the radii function in Algorithm 2 to generate a canal surface with circular cross-section.

As mentioned previously, we can also use canal surfaces with non-circular cross-sections in order to cover a smaller yet more reasonable area while maintaining all the implicit local constraints of the task more efficiently. For example, to generate canal surfaces with elliptical cross-sections, (10) has to be modified to include the major and minor axes of the ellipse according to (5) (see Fig. 4b). As a more general form, B-splines (a generalized form of Bézier curves) provide a powerful tool for fitting a smooth curve to a set of key-points. Given the set of demonstrations, at each arc-length, a closed B-spline can be fitted to the data representing the cross-section of the canal surface at that point (Fig. 4c).

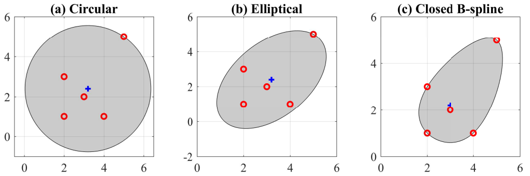


Fig. 4: Cross-sections with different shapes estimated on the same set of data. Blue dots represent the point on the directrix, and the red dots are the corresponding points on the demonstrations.

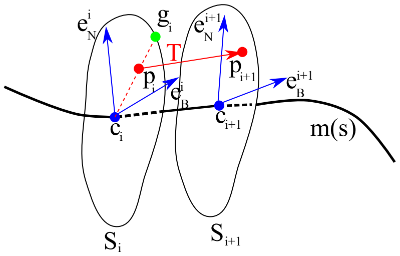


Fig. 5: Reproduction from a random initial pose p_i on the i^{th} cross-section S_i. Firstly the ratio of the initial point is calculated as $\eta = \frac{\overline{p_i c_i}}{\overline{g_i c_i}}$. Then the point is transferred to the next cross-section, S_{i+1} and scaled by $\overline{g_{i+1}c_{i+1}}$ such that $p_{i+1} = \eta.\overline{g_{i+1}c_{i+1}}.(T_{\mathscr{F}_{i+1},\mathscr{F}_i})$, where T is the transformation matrix between two frames.

B. Skill Reproduction

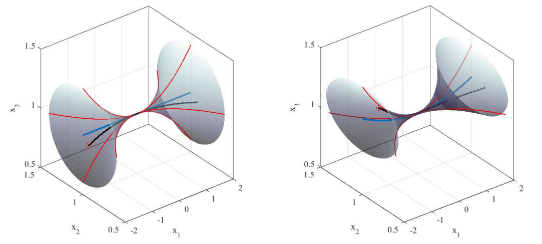
During the reproduction phase, the initial position of the end-effector p_0 in the cross-sectional plane S_0 (perpendicular to the directrix at c_0) is used as input. We measure the ratio η (Line 10) by calculating the distance $\overline{p_0 c_0}$ and the distance $\overline{g_0 c_0}$, where g_0 is located at the boundary of the corresponding cross-section (see Fig. 5). We use η to generate the next pose of the end-effector by transforming p_0 from the current TNB frame $\mathscr{F}_0$ to the next (Line 13). The pose on the second frame is then calculated as: $p_1 = \eta(\overline{g_1 c_1})(T_{\mathscr{F}_1,\mathscr{F}_0})$. Fig. 5 illustrates a single-step reproduction process using the *ratio rule*. The ratio rule can generate new trajectories from any point inside the canal. It also ensures that the essential characteristics of the demonstrated skill are applied to the reproduced time-independent trajectory.

V. EXPERIMENTS

To validate the capabilities and interpretability of the proposed approach, we performed a set of experiments using both noiseless simulated and real-world data. We also compared our approach with two well-known LfD approaches namely, DMPs and GMM/GMR.

A. Simulated demonstrations

In the first experiment, as shown in Fig. 6a, we manually simulate four symmetric demonstrations. The demonstrations imply that the movement can be started and ended in a wide task-space; in the middle, however, it is constrained to pass through a narrow area. This movement resembles threading a needle or picking an object placed in the middle of the movement. The obtained canal surface extracts and preserves the important characteristics of the demonstrated task. Given an arbitrary initial pose of the end-effector, our approach is capable of reproducing the learned skill using the ratio rule (Section IV-B). Although the directrix of the four symmetric trajectories is a straight line, we can reproduce a variety



(a) four demonstrations (b) three demonstrations

Fig. 6: Demonstrations (red), directrix (blue), canal surface (gray), and reproductions (dotted black) for the simulated experiments.

of trajectories that satisfy the constraint of the task in the middle. Using GMM/GMR in this case results in reproducing a straight trajectory similar to the directrix.

In our second experiment, we removed one of the demonstrations from the set and repeated the experiment. Although a canal surface with a circular cross-section still generates a valid encoding of the task, a more descriptive encoding of the skill can be represented using an elliptical cross-section. In fact, due to symmetry, a canal surface with an elliptical cross-section for the previous set would be identical to one with a circular cross-section. The result in Fig. 6b indicates the adaptation of the canal surface to the new situation while preserving the main characteristics of the task. In this experiment, the adaptation of the model and the reproduced trajectory after removing the demonstration are fairly predictable. That is, it can be seen that the canal surface and its directrix gravitate towards the area with more information. Such features make canal surfaces a promising tool for producing visually understandable models employable even by non-expert users.

B. Real-world demonstrations

In this section we present results for three skills recorded through kinesthetic teaching (see Fig. 3) using a 6DOF Kinova Jaco2 robotic arm. The data is recorded at a sampling rate of 30Hz. The captured demonstrations together with the obtained canal surfaces are illustrated in Fig. 7.

The first experiment shows a reaching skill where the object (green sphere) is placed on a table (Fig. 7a). The second experiment (Fig. 7b) shows a picking skill that resembles the simulated experiment in Section V-A. The third experiment shows a circular movement around an object (blue sphere) while avoiding collision (Fig. 7c). All the obtained canal surfaces, which are formed with circular cross-sections, represent the demonstrated skill continuously and enable the robot to reproduce the movement from any initial pose. Fig. 8 shows the demonstrations, directrix, reproductions, and the boundaries of the obtained canal surfaces plotted axis-wise for each task. Since we used circular canal surfaces, for each task the radii function at each step is estimated using the maximum distance from the directrix.

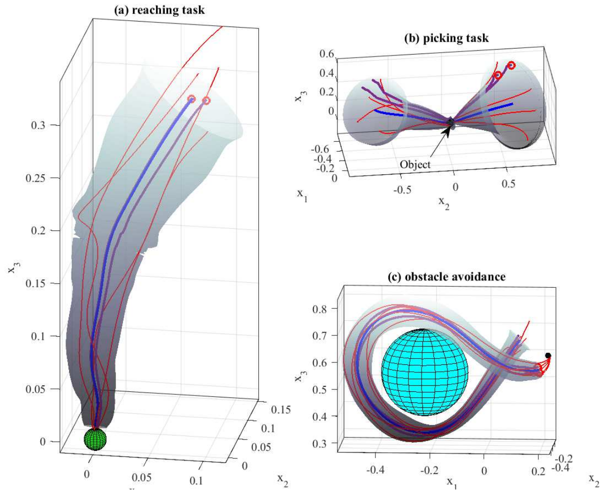


Fig. 7: Demonstrations (red), directrix (blue), reproductions (magenta), canal surface (gray) for three real-world experiments performed using our approach.

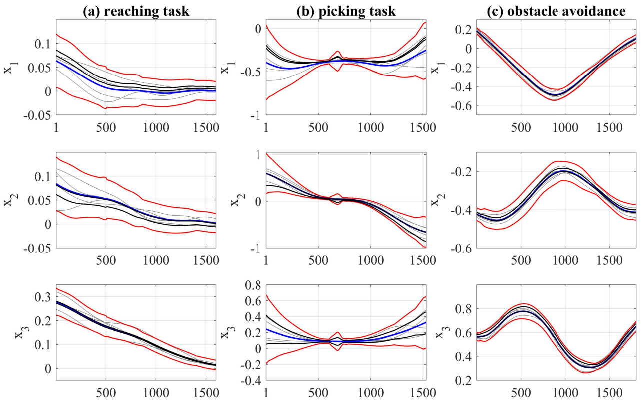


Fig. 8: Demonstrations (gray), directrix (blue), reproductions (black), and boundaries of the canal surface (red) for three real-world experiments performed using our approach. Task-space trajectories relative to base of the robot are plotted with respect to number of points in each task.

C. Comparing with other approaches

In this section we compare our approach to two well-known LfD approaches: DMPs [10] and GMM/GMR [6]. We employed these two algorithms to learn and generalize the three tasks explained in Section V-B. Due to lack of space, we only present results for the obstacle avoiding experiment and highlight the differences among the three approaches. The demonstrations and the obstacle, which is unknown to the robot, are depicted in Fig. 9a. We encoded the demonstrations using a canal surface with circular cross-section and reproduced five reproductions from various initial poses (Fig. 9b). As mentioned before, our approach requires no parameter tuning and enables the robot to learn the characteristics of the movement to avoid the obstacle. All the reproductions are guaranteed to remain inside the canal.

In our first attempt, we trained DMPs with five attractors and GMMs with five components. For DMPs several parameters, such as proportional and damping gains, also have to be

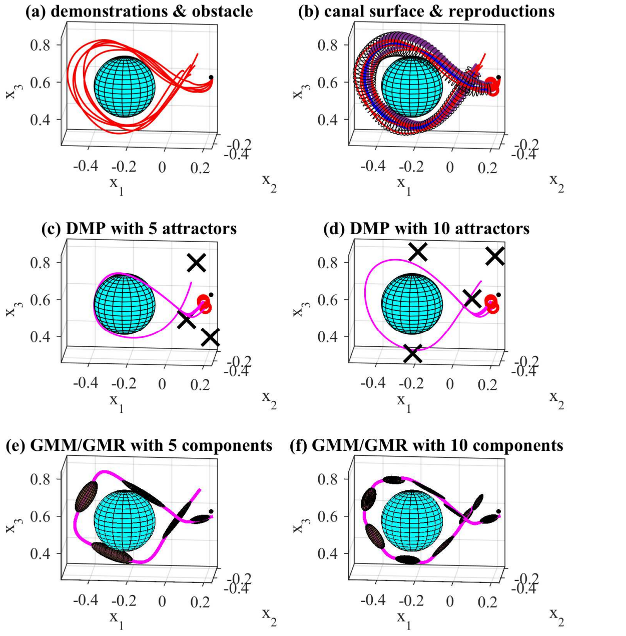


Fig. 9: From left to right, (a) five demonstrations and the obstacle, (b) obtained canal surface and five reproductions, (c) DMP with five attractors and five reproductions, and (d) GMM/GMR with five components and reproduction (see Section V-C).

tuned. Using the obtained model, we reproduced five trajectories using DMPs from the same initial poses. The results show that the reproduced trajectories could not accurately mimic the movement and avoid the obstacle (Fig. 9c). In addition, all of the reproduced trajectories converge toward the first attractor and exhibited a similar behavior afterwards.

Except for the number of components, GMM does not require additional tuning, but we have to include timing information to the dataset. As depicted in Fig. 9e since GMR uses time as input in order to reproduce a trajectory, it cannot generalize over the initial pose and all the reproductions would be identical. Although the reproduced trajectory manages to avoid the obstacle, the shape of the movement is not circular. It has to be noted that the reproductions by DMPs and GMR are similar to the directrix of the obtained canal surface (which by itself can be used as a reproduction). Whereas, the reproductions using the canal surface are not identical, they only maintain the important characteristics of the task while exploiting the whole space bounded by the canal surface.

In our next attempt, we double the number of DMP attractors and Gaussian components. The results show that both learned models are improved (Fig. 9d and 9f). This experiment shows the vital and sensitive effect of parameter tuning on both approaches especially on DMPs. Using GMM/GMR, we had to include timing information in our dataset. Also, with ten Gaussian components, the EM algorithm required more iterations to converge that had to be tuned manually.

VI. Discussion and Future Work

We proposed a parameter-free approach that not only simplifies the usage of the algorithm and makes the results consistent, but also can make our approach move convenient for non-expert users. Our approach preserves the main characteristics of the movement (including its boundaries), exploits the whole demonstrations space, and reproduces a various range of trajectories that achieve the goal of the task. Achieving similar results using other existing approaches such as DMPs and GMM/GMR requires trail and error.

Moreover, the proposed approach can be adapted to the corrections and constraints applied by a human teacher. Our future work includes activating the robot in compliant control mode and enabling the teacher to interact and refine the robot's movements during reproduction. The learned model can be actively updated, and the new reproductions will reflect the refinements.

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] H. A. Yanco, A. Norton, W. Ober, D. Shane, A. Skinner, and J. Vice, "Analysis of human-robot interaction at the darpa robotics challenge trials," *Journal of Field Robotics*, vol. 32, no. 3, pp. 420–444, 2015.

[3] D. Hilbert and S. Cohn-Vossen, "Geometry and the imagination," *Chelsea, New York*, 1952.

[4] S. Vijayakumar, A. D'souza, and S. Schaal, "Incremental online learning in high dimensions," *Neural computation*, vol. 17, no. 12, pp. 2602–2634, 2005.

[5] D. B. Grimes, R. Chalodhorn, and R. P. Rao, "Dynamic imitation in a humanoid robot through nonparametric probabilistic inference," in *Robotics: science and systems*. Cambridge, MA, 2006, pp. 199–206.

[6] 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.

[7] M. Schneider and W. Ertel, "Robot learning by demonstration with local gaussian process regression," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ Intl. Conf. on*. IEEE, 2010, pp. 255–260.

[8] B. Reiner, W. Ertel, H. Posenauer, and M. Schneider, "Lat: A simple learning from demonstration method," in *Intelligent Robots and Systems (IROS), IEEE/RSJ Intl. Conf. on*. IEEE, 2014, pp. 4436–4441.

[9] E. Gribovskaya, S. M. Khansari-Zadeh, and A. Billard, "Learning non-linear multivariate dynamics of motion in robotic manipulators," *The International Journal of Robotics Research*, pp. 1–37, 2010.

[10] 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.

[11] S. Quinlan and O. Khatib, "Elastic bands: Connecting path planning and control," in *Robotics and Automation. 1993 IEEE Intl. Conf. on*. IEEE, 1993, pp. 802–807.

[12] A. Majumdar and R. Tedrake, "Funnel libraries for real-time robust feedback motion planning," *arXiv preprint arXiv:1601.04037*, 2016.

[13] S. Dong and B. Williams, "Learning and recognition of hybrid manipulation motions in variable environments using probabilistic flow tubes," *International Journal of Social Robotics*, vol. 4, no. 4, pp. 357–368, 2012.

[14] U. Shani and D. H. Ballard, "Splines as embeddings for generalized cylinders," *Computer Vision, Graphics, and Image Processing*, vol. 27, no. 2, pp. 129–156, 1984.

[15] E. Hartmann, "Geometry and algorithms for computer aided design," *Darmstadt University of Technology*, 2003.

[16] M. Shanmugavel, A. Tsourdos, R. Zbikowski, and B. A. White, "3d path planning for multiple uavs using pythagorean hodograph curves," in *AIAA Guidance, Navigation, and Control Conference and Exhibit, Hilton Head, South Carolina*, 2007, pp. 20–23.

[17] E. Abbena, S. Salamon, and A. Gray, *Modern differential geometry of curves and surfaces with Mathematica*. CRC press, 2006.

[18] C. Myers, L. Rabiner, and A. Rosenberg, "Performance tradeoffs in dynamic time warping algorithms for isolated word recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 6, pp. 623–635, 1980.