Encoding Demonstrations and Learning New Trajectories using Canal Surfaces *

S. Reza Ahmadzadeh and **Sonia Chernova**
School of Interactive Computing, Georgia Institute of Technology
{reza.ahmadzadeh , chernova}@gatech.edu

Abstract

We propose a novel learning approach based on differential geometry to extract and encode important characteristics of a set of trajectories captured through demonstrations. The proposed approach represents the trajectories using a surface in Euclidean space. The surface, which is called *Canal Surface*, is formed as the envelope of a family of regular implicit surfaces (e.g. spheres) whose centers lie on a space curve. Canal surfaces extract the essential aspects of the demonstrations and retrieve a generalized form of the trajectories while maintaining the extracted constraints. Given an initial pose in task space, a new trajectory is reproduced by considering the relative ratio of the initial point with respect to the corresponding cross-section of the obtained canal surface. Our approach produces a continuous representation of the set of demonstrated trajectories which is visually perceivable and easily understandable even by non-expert users. Preliminary experimental results using simulated and real-world data are presented.

1 Introduction

The main goal of Learning from Demonstration (LfD) approaches is to enable even non-experts to teach new skills to robots interactively through demonstrations. LfD approaches eliminate the need for manual programming of the desired behavior to the robot and instead, they use a set of successful examples of the behavior to learn a model. The robot then should be able to learn and 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[1]. Therefore, ignoring the presence of the human teacher after the demonstration phase, which is a common case, becomes one of the main drawbacks of most LfD approaches. In fact, the existing representations are so complicated that make it almost impossible for non-experts to diagnose a failure or an undesirable behavior and find a way to resolve it. A possible solution, however, is to employ an encoding process that is both powerful enough to learn and generalize the task robustly and is also readily comprehensible for non-experts. Such learning approach should take advantage of teacher's feedback not only for providing new demonstrations but also for updating and adjusting the learned model in the loop.

In this paper, we propose a novel LfD approach based on Differential Geometry that enables robots to acquire novel trajectory-based skills through demonstrations. This approach, which builds on our previous work [Ahmadzadeh and Chernova, 2016], encodes the demonstrated skill as a geometric model. The model is composed of a regular curve and a surface in three-dimensional Cartesian space called a *Canal Surface*. The constructed canal surface represents the main features of the skill and its constraints. Since the encoded skills using canal surfaces are visually perceivable and easily understandable, they enable even non-expert users to provide feedback to improve the quality of the learned skill. Canal surfaces and their parametric formulation are described in Section 3. The proposed learning approach is explained in Section 4 and preliminary experimental results using artificial and real data gathered from a Jaco^2 robotic arm are presented in Section 5. Then in Section 5.4, the proposed approach is compared with the Gaussian Mixture Model [Calinon *et al.*, 2007]. Potential extension of the proposed approach and its limitations are discussed in Section 6.

2 Related Work

During the two past decades, several trajectory-based LfD approaches have been developed to encode a demonstrated skill and retrieve a generalized form of the trajectory [Argall *et al.*, 2009]. Many of them use regression-based techniques to represent the given set of demonstrations using a probabilistic representation [Vijayakumar *et al.*, 2005; Grimes *et al.*, 2006]. One of the most well-known works in this category is by Calinon *et al.,* [2007], which builds a probabilistic representation of the demonstrations using Gaussian Mixture Model (GMM) and uses Gaussian Mixture Regression (GMR) to retrieve a smooth trajectory. The encoded model is learned through the Expectation-Maximization (EM) algorithm. In GMM/GMR the learning and reproduction phases are fast and can be used in an online manner, but the reproduce trajectory is time-dependent. similarly, Gaussian Pro-

*This work is supported in part by the Office of Naval Research award N000141410795.

[1]For instance, see results from the recent DARPA Robotic Challenge at http://www.theroboticschallenge.org/

cess (GP) Regression was proposed to generalize over a set of demonstrated trajectories [Grimes *et al.*, 2006]. Later local Gaussian process regression was employed to extract the constraints of a demonstrated skill [Schneider and Ertel, 2010]. Another approach similar to Gaussian Process called *LfD by Averaging Trajectories* (LAT), was introduced in [Reiner *et al.*, 2014]. LAT is based on the product of normal distributions estimated from trajectories relative to the objects observed during the demonstrations. Each demonstration is transformed with respect to the pose of the objects in the scene. Both LAT and GP cannot extract constraints of the demonstrated skills.

Ijspreet *et al.,* showed that dynamical systems can also be used to encode and reproduce trajectories [Ijspeert *et al.*, 2002; Ijspeert *et al.*, 2013]. Dynamic Movement Primitives (DMPs) represent the demonstrated trajectories as movements of a particle subject to a set of damped linear spring 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. However, the boundaries and the state-space formed by the DMP representation are not visually perceivable.

In order to reflect a human's intention more accurately, Dong and Williams [2012] proposed a representation of continuous actions (i.e. trajectories) by extracting covariance data which is called probabilistic flow tubes. They utilized binary contact information among objects and state of the environment during the demonstrations in order to match all the sequences temporally. The learned flow tube consists of a spine trajectory and a set of 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 the covariance data.

Another similar idea has been proposed in [Majumdar and Tedrake, 2016] to address the problem of real-time motion planning. Firstly, a finite library of open-loop trajectories and their corresponding designed controllers are computed off-line. This library approximates a boundary which can be visualized as a funnel around the trajectory. A closed-loop system, which is guaranteed to remain within the computed funnel, is then used to generate trajectories in real-time that can deal with obstacles in the environments. The computed funnel illustrates a similar representation to the proposed approach in this paper. However, we do not use controllers and off-line computation.

3 Canal Surface

Canal surfaces, which play a fundamental role in descriptive geometry, were first introduced by Hilbert *et al.,* [1952]. In the context of Computer Aided Graphic Design (CAGD), canal surfaces are used for the construction of smooth blending surfaces, shape reconstruction, and transition surfaces between pipes [Farouki and Sverrisson, 1996; Hartmann, 2003]. They are also known as Generalized Cylinders [Shani and Ballard, 1984] which can be explained as a representation of an elongated object composed of a spine and a smoothly varying cross-section. In robotics, canal surfaces have been used for path planning [Shanmugavel *et al.*, 2007]. In this paper, we use canal surfaces to encode a set of trajectories captured through demonstrations, extract their important features, and retrieve a generalized form of the movement. This section provides an overview of the mathematical formulation of canal surfaces.

Envelope of a Pencil of implicit Surfaces

Let Φ_u be the one-parameter pencil[2] of regular implicit surfaces[3] with real-valued parameter u [Abbena *et al.*, 2006]. Two surfaces corresponding to different values of u intersect in a common curve. As u varies, the generated surface is the envelope[4] of the given pencil of surfaces. In 3D space, the envelope can be defined using the following equations:

$$\Phi_u : F(x, y, z, u) = 0, \tag{1}$$

$$\frac{\partial F}{\partial u}(x, y, z, u) = 0, \tag{2}$$

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

Canal Surface

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

$$\mathcal{C}_u : f(\mathbf{x}; u) := \{(\mathbf{c}(u), r(u)) \in \mathbb{R}^4 | u \in \mathbb{R}\}, \tag{3}$$

where the spheres are centered on a regular curve $\Gamma : \mathbf{x} = \mathbf{c}(u) \in \mathbb{R}^3$ in Cartesian space. The radius of the spheres are given by the function $r(u) \in \mathbb{R}$, which is a C^1-function. The non-degeneracy condition is satisfied by assuming $r > 0$ and $|\dot{r}| < \|\dot{\mathbf{c}}\|$ [Hartmann, 2003]. In Differential Geometry, Γ is known as the *spine curve* or *directrix* and $r(u)$ is called the *radii* function. For the one-parameter pencil of **spheres**, Equation 3 can be written as

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

Parametric Representation

To form a parametric representation of Equation 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 $\{\mathbf{e}_T, \mathbf{e}_N, \mathbf{e}_B\}$, 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:

$$\mathbf{e}_T = \frac{d\mathbf{x}(s)}{ds}, \tag{5}$$

$$\mathbf{e}_N = \frac{d\mathbf{e}_T}{ds} / \|\frac{d\mathbf{e}_T}{ds}\|, \tag{6}$$

$$\mathbf{e}_B = \mathbf{e}_T \times \mathbf{e}_N. \tag{7}$$

[2] A pencil is a family of geometric objects sharing a common property.

[3] An implicit surface is a surface in Euclidean space that can be defined by an equation such as $F(x(u), y(u), z(u)) = 0$ [Abbena *et al.*, 2006]. For instance, a sphere is an implicit surface.

[4] A surface tangent to each member of a family of surfaces in 3D space is called an envelop.

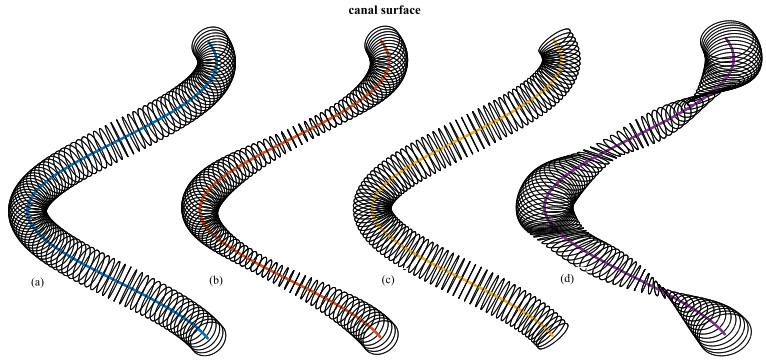


Figure 1: Four canal surfaces sharing the same directrix with different radii function. (a) a constant circular cross-section, (b) time-varying circular cross-section, (c) constant elliptical cross-section, and (d) time-varying elliptical cross-section.

Applying the second condition of envelopes from Equation 3 to the formula of a canal surface defined by Equation 4 gives

$$\frac{\partial \mathcal{C}_s}{\partial s} = 2(\mathbf{x} - \mathbf{c}(s))\mathbf{e}_T - 2r(s)\frac{dr}{ds} = 0. \tag{8}$$

For each value of s, Equation 8 represents a circle orthogonal to the unit tangent vector of the directrix. Thus the canal surface can be represented as a set of circles

$$\mathcal{C}_s : f(s,v) = \mathbf{c}(s) + \tag{9}$$
$$r(s)\left(-\mathbf{e}_T\frac{dr}{ds} + \sqrt{1-(\frac{dr}{ds})^2}\,(\mathbf{e}_B\sin(v) - \mathbf{e}_N\cos(v))\right),$$

where v represents a parameter that varies from 0 to 2π along the circle. Equation 9 denotes the parametric representation of a canal surface with circular cross-section. In some cases, the cross-section may vary in shape and size when the TNB frame translates along the directrix. For instance, it can start with a circular shape and then gradually vary to an ellipse. In general, cross-sections can be represented using different shapes and techniques such as polygons, polynomials, parabolic blending, cone sections, and B-splines[Shani and Ballard, 1984]. Such characteristics makes canal surfaces a suitable candidate for modeling complicated constraints of trajectory-based skills captured through demonstrations. Figure 1 illustrates a few examples of canal surfaces with circular and elliptical cross-sections.

4 Skill Learning using Canal Surfaces

In this section, we explain how canal surfaces can be employed to encode, learn and reproduce new skills from demonstrations. We assume that multiple examples of the same skill are demonstrated. Although any demonstration technique such as teleoperation, and shadowing can be employed, we use kinesthetic teaching. Our approach consists of two phases: learning and reproduction. In the learning phase, we build a canal surface by calculating the directrix and the radii function from the given demonstrations. In the reproduction phase, we devise a geometric method for generating new trajectories starting from arbitrary initial positions. Note that Algorithm 1 shows a pseudo code of the proposed approach in its simplest form.

4.1 Learning Phase

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

Algorithm 1 Skill Learning using Canal Surfaces

procedure ENCODING DEMONSTRATIONS
  Input:set of demonstrations $\boldsymbol{\xi} \in \mathbb{R}^{3\times N\times n}$
  Output: Canal surface $\mathcal{C}_s$, all TNB frames $\mathcal{F}$
  $\mathbf{m}(s) \leftarrow$ *mean*$(\boldsymbol{\xi})$
  $r(s) \leftarrow$ *boundary*$(\boldsymbol{\xi})$
  $\mathcal{C}_s, \mathcal{F} \leftarrow$ *makeCanalSurface*$(\mathbf{m}(s), r(s))$
procedure REPRODUCING TRAJECTORY
  Input:initial point $\mathbf{p}_0 \in \mathbb{R}^3$
  Output: New trajectory $\boldsymbol{\rho} \in \mathbb{R}^{3\times N}$
  $\eta \leftarrow \frac{\|\mathbf{p}_0 - \mathbf{c}_0\|}{\|\mathbf{g}_0 - \mathbf{c}_0\|}$
  $\mathbf{p}_i \leftarrow \mathbf{p}_0 \quad , \quad \boldsymbol{\rho} \leftarrow \mathbf{p}_0$
  for each *frame*$\mathcal{F}_i$ **do**
    $\mathbf{p}_{i+1} \leftarrow$ *project*$(\mathbf{p}_i, \eta, \mathcal{F}_{i+1}, \mathcal{F}_i)$
    $\boldsymbol{\rho} \leftarrow \mathbf{p}_{i+1}$
    $i \leftarrow i+1$

Algorithm 2 Generating a canal surface with 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 $\mathcal{C}_s$, all TNB frames $\mathcal{F}$
4: $v \leftarrow 0 : 2\pi$
5: **for each** $\mathbf{m_i} \in \mathbf{m}(s)$ and $r_i \in r(s)$ **do**
6: $\{\mathbf{e}_T^i, \mathbf{e}_N^i, \mathbf{e}_B^i\} \leftarrow$ *estimateTNBframe*$(\mathbf{m_i})$
7: $\mathcal{F} \leftarrow \{\mathbf{e}_T^i, \mathbf{e}_N^i, \mathbf{e}_B^i\}$
8: $\mathcal{C}_s \leftarrow \mathbf{m_i} + r_i \left(\mathbf{e}_N^i \cos(2\pi v) + \mathbf{e}_B^i \sin(2\pi v)\right)$

Estimating the Directrix

In order to estimate the backbone curve or the directrix, we can simply calculate the directional mean value for the given set of demonstrations. Let $\mathbf{m} \in \mathbb{R}^{3\times N}$ be the arithmetic mean of $\boldsymbol{\xi}$. Note that $\mathbf{m}$ is the space curve that all the spheres are centered on to form a canal surface. Line 4 in Algorithm 1 calculates the directrix. Alternatively, the directrix can be produced using GMR [Calinon *et al.*, 2007]. In that case, GMR generates the directrix by sampling from the learned statistical model using GMM.

Estimating the Radii

In this step, we explain methods for calculating the cross-section of the canal surface at each time step. The circumference of a cross-section represents the implicit local constraints of the task imposed by the set of demonstrations. For instance, when all the trajectories pass through a very narrow area, the tutor is 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.

In its simplest form, the radii function of a canal surface with a circular cross-section can be calculated by measuring the distance from each point on the directrix to the corresponding points on the demonstrated trajectories. If these points are assumed as vertices of a polygon, a circumscribed circle can be found that passes through all the vertices. Another method that we employ in our experiments is to use the vertex with a maximum distance from the center to make a circle that bounds other points as well. Figure 2 depicts the estimated boundaries for a real-world experiment. Line 5 in Algorithm 1 calculates the boundaries and forms the radii function which is later used to form the canal surface (Line 6). Algorithm 2 shows a procedure for generating a canal surface with a circular cross-section.

Note that to generate canal surfaces with elliptical cross-sections, Equation 9 has to be modified to include the major and minor axes of the ellipse. The advantage of such modification is that the obtained cross-section can cover a smaller yet more reasonable area while maintaining all the implicit local constraints of the task more efficiently. Some examples can be seen in Figure 1 where four canal surfaces with identical directrix and different radii functions are illustrated.

In a more general form, as mentioned before, cross-sections can be represented by different shapes and techniques. Among others, B-splines, which are 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. A set of B-spline cross-sections are depicted in Figure 3 where two, three, four, and more demonstrations are given.

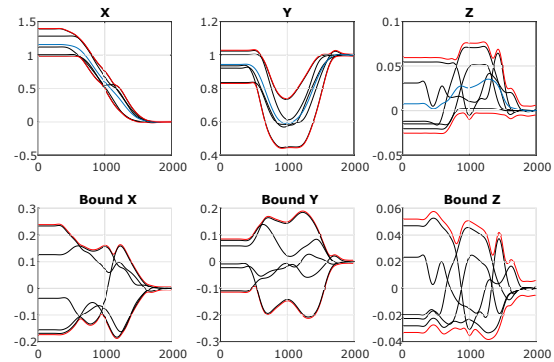


Figure 2: Extracted constraints of the task used for calculating the radii function. The demonstrations are shown in black and the extracted boundaries are shown in red.

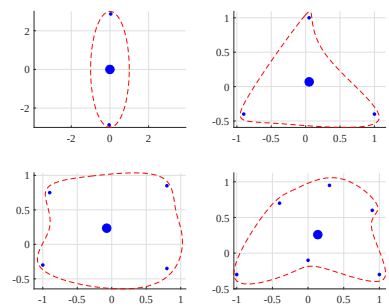


Figure 3: Cross-section generated using closed B-splines with two, three, four, and more demonstrations. Small dots (blue) represent the points sampled from the demonstrations and the big (blue) dot shows the corresponding point on the directrix.

4.2 Reproduction Phase

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 between p_0 and c_0 ($\overline{p_0 c_0}$) and the distance between c_0 and g_0 placed at the boundary of the corresponding cross-section ($\overline{g_0 c_0}$). We use the measured ratio to generate the next pose of the end-effector by transforming it from the current TNB frame to the next (Line 13). An illustration of a single-step reproduction process using the *ratio rule* can be seen in Figure 4. The ratio rule ensures that the essential characteristics of the demonstrated skill are applied to the reproduced time-independent trajectory.

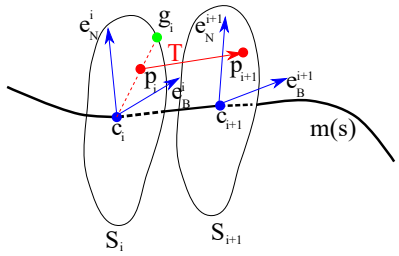


Figure 4: Reproduction from a random initial pose p_i on the ith cross-section S_i. Firstly the ratio of the initial point is calculated as $\eta = \frac{\overline{p_i c_i}}{\overline{g_i c_i}}$. Then the point is transferred to the next cross-section, S_{i+1} and scaled by $\overline{g_{i+1} c_{i+1}}$ such that $p_{i+1} = \eta.\overline{g_{i+1} c_{i+1}}.(T p_i)$, where T is the transformation matrix between two frames.

Dealing with Noise

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). Under such circumstance, a possible solution is to produce the unit normal vector $\mathbf{e}_N$ as the cross product of a random vector by the unit tangent vector $\mathbf{e}_T$.

5 Experiments

To validate the capabilities and interpretability of the proposed approach, we performed a set of experiments using both artificially made and real demonstrations. The simulated experiments show the capabilities of the proposed approach using noiseless data while the real-world experiment shows its feasibility. The real demonstrations were collected through kinesthetic teaching using a 6DOF Jaco^2 robotic arm from Kinova. The data is recorded at a sampling rate of 30Hz.

5.1 Task I

Consider four simulated demonstrations as illustrated in Figure 5. The demonstrations imply that the motion can begin and end in a wide task-space but it is constrained in the middle and has to pass through a very narrow area. The obtained directrix and the canal surface with a circular cross-section are estimated using Algorithm 1 and are shown in Figure 5. Although the directrix of the four identical trajectories becomes a straight line, the canal surface extracts and preserves the important features and constraints of the demonstrated task. Given a new initial pose of the end-effector, the reproduced trajectory using the ratio rule (explained in Section 4.2) starts and finishes inside the canal surface while satisfying the constraint of the task in the middle.

5.2 Task II

In this task, one of the demonstrations from the previous set is removed and the experiment is repeated. The result is depicted in Figure 6. Although the canal surface calculated in the previous experiment is still valid for this set as well, a better description of the skill can be represented using a canal surface with an elliptical cross-section. The reproduced

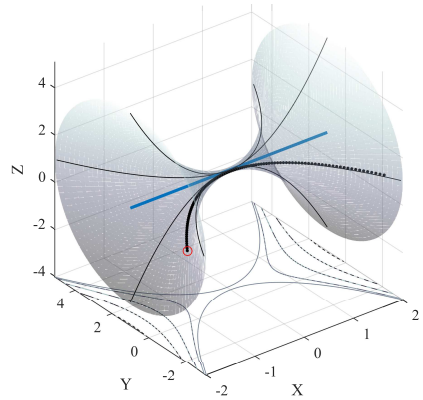


Figure 5: Four demonstrations (solid black), directrix (blue) and the encoded canal surface (gray) for task I. The projection of the surface on the $X - Y$ plane shows the boundaries of the obtained canal surface. Reproduction of the motion (dotted black) from a random initial pose (red marker) is depicted.

trajectory generated using the ratio rule (explained in Section 4.2) is also illustrated in Figure 6. This experiment indicates that after removing a demonstration from the set, the canal surface adapts to the new situation while maintaining the constraints of the task. And also, the adaptation of the model and the reproduced trajectory are fairly predictable. Such features make canal surfaces a promising approach for producing visually understandable models employable even by non-expert users.

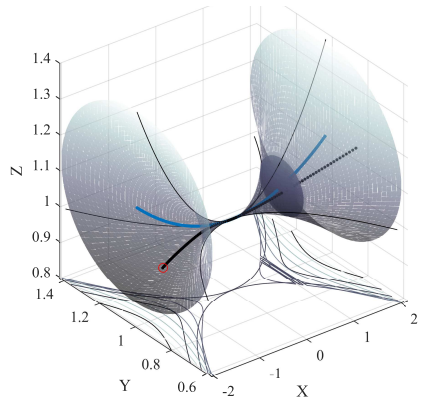


Figure 6: One of the demonstrations is removed from the previous set and the canal surface is re-generated with an elliptical cross-section (task II). The directrix (blue) is inclined towards the demonstration on top. Reproduction of the motion (dotted black) from a random initial pose (red marker) is depicted.

5.3 Task III

In this experiments, we collected five demonstrations in task-space using the Jaco^2 robotic arm through kinesthetic teaching. Figure 2 shows the captured demonstrations and the extracted constraints of the task. The obtained canal surface using Algorithm 1 with an elliptical cross-section is illustrated in Figure 7. The reproduced time-independent trajectory is

generated using the ratio rule as explained in Section 4.2. This experiment shows the feasibility of the proposed approach in real-world experiments.

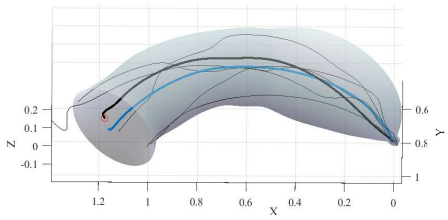


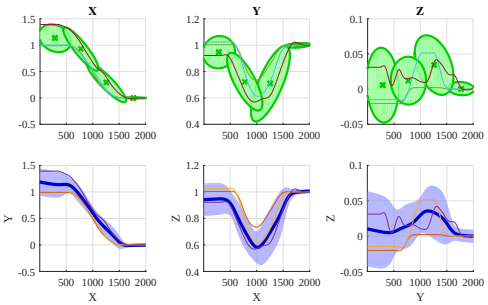
Figure 7: Captured demonstrations (solid black lines) using the Jaco^2 robotic arm and the corresponding encoded canal surface. The directrix is shown in blue and the reproduction of the motion from a random initial pose is plotted with dotted black line.

5.4 Comparing with GMM/GMR

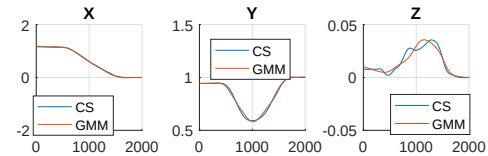
We applied GMM/GMR approach as presented in [Calinon *et al.*, 2007] to the set of real demonstrations from Section 5.3. As shown in Figure 8a, a statistical model is trained using GMM with 4 Gaussian components, and a generalized version of the dataset with associated constraints is retrieved through GMR. We found that 4 Gaussian components efficiently encode the skill. As it can be seen in Figure 8b, comparing the outcome of GMM/GMR with the results from Section 5.3 reveals that the reproduced trajectory by GMM/GMR is similar to the directrix of the modeled canal surface. The obtained representation by using our approach is visually descriptive and easily understandable even for non-experts. This feature enables end-users to evaluate the given set of demonstrations and improve the learned model by providing proper feedback (e.g. verbal, physical). Another advantage is that our approach reproduces time-independent trajectories from any random initial pose inside the canal. However, GMR reproduces time-based trajectories and requires an additional component to generate trajectories from an arbitrary initial pose. Several reproduced trajectories by our approach from various initial poses are depicted in figure 9.

6 Discussion and Future Work

One of the advantages of employing geometrical approaches such as canal surfaces is that the obtained representation is visually descriptive and easily understandable even for non-expert users. This feature enables end-users to evaluate the given set of demonstrations and improve the learned model by providing proper feedback (e.g. verbal, physical). Unlike many existing LfD approaches in which the human operator is only at the beginning of the process, our approach is capable of keeping the user in the loop. For instance, if by observing the obtained model the end-user realizes that the canal needs to be more constrained at some specific area, the model can be updated by removing some of the trajectories and adding new ones. Making such decisions to adjust the model would be very complicated for non-experts while dealing with representations such as DMPs and GPs.



(a) Result from GMM/GMR.



(b) Directrix vs. GMR reproduction.

Figure 8: Comparing the proposed approach with GMM/GMR.

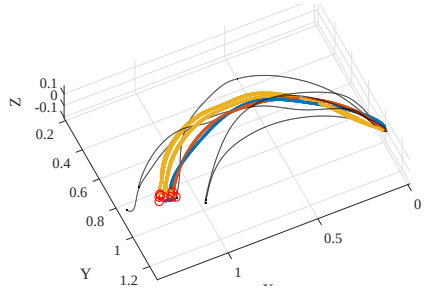


Figure 9: Demonstrations (black), directrix (blue), GMR reproduction (orange), and multiple reproductions from arbitrary initial points (yellow) using our approach, for task III.

Another merit of our approach is that unlike many existing approaches the reproduced trajectories are time-independent. This feature enables our approach to reproduce trajectories reactively from any point inside the canal surface.

The proposed approach can be adapted to the adjustments and constraints applied even during the reproduction phase. Our future work includes activating the robot in compliant control mode to enable the user to interact and refine the robot's movements during reproduction. The learned model can be actively updated based on the physical corrections from the user. The new reproduced trajectories based on the updated model will reflect the given feedback in the form of new constraints on the surface.

Furthermore, although the proposed approach is capable of encoding and reproducing trajectories on its own, it can be combined with other methods such as DMPs and keyframe-based LfD [Akgun *et al.*, 2012]. Integration with DMPs can provide us with a different reproduction method while keyframe-based LfD enables the system to take advantage of smoother trajectories while retaining their important characteristics.

References

[Abbena *et al.*, 2006] Elsa Abbena, Simon Salamon, and Alfred Gray. *Modern differential geometry of curves and surfaces with Mathematica*. CRC press, 2006.

[Ahmadzadeh and Chernova, 2016] Seyed Reza Ahmadzadeh and Sonia Chernova. A geometric approach for encoding demonstrations and learning new trajectories. In *Robotics: Science and Systems (RSS 2016), Workshop on Planning for Human-Robot Interaction: Shared Autonomy and Collaborative Robotics*, pages 1–3, Ann Arbor, MI, USA, June 2016. IEEE.

[Akgun *et al.*, 2012] Baris Akgun, Maya Cakmak, Karl Jiang, and Andrea L. Thomaz. Keyframe-based learning from demonstration. *International Journal of Social Robotics*, 4(4):343–355, 2012.

[Argall *et al.*, 2009] Brenna D. Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5):469–483, 2009.

[Calinon *et al.*, 2007] Sylvain Calinon, Florent Guenter, and Aude Billard. On learning, representing, and generalizing a task in a humanoid robot. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 37(2):286–298, 2007.

[Dong and Williams, 2012] Shuonan Dong and Brian Williams. Learning and recognition of hybrid manipulation motions in variable environments using probabilistic flow tubes. *International Journal of Social Robotics*, 4(4):357–368, 2012.

[Farouki and Sverrisson, 1996] Rida Amt Farouki and Ragnar Sverrisson. Approximation of rolling-ball blends for free-form parametric surfaces. *Computer-Aided Design*, 28(11):871–878, 1996.

[Grimes *et al.*, 2006] David B. Grimes, Rawichote Chalodhorn, and Rajesh PN Rao. Dynamic imitation in a humanoid robot through nonparametric probabilistic inference. In *Robotics: science and systems*, pages 199–206. Cambridge, MA, 2006.

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

[Hilbert and Cohn-Vossen, 1952] David Hilbert and Stephan Cohn-Vossen. Geometry and the imagination. *Chelsea, New York*, 1952.

[Ijspeert *et al.*, 2002] Auke J. Ijspeert, Jun Nakanishi, and Stefan Schaal. Movement imitation with nonlinear dynamical systems in humanoid robots. In *Robotics and Automation (ICRA), 2002 IEEE International Conference on*, volume 2, pages 1398–1403. IEEE, 2002.

[Ijspeert *et al.*, 2013] Auke J. Ijspeert, Jun Nakanishi, Heiko Hoffmann, Peter Pastor, and Stefan Schaal. Dynamical movement primitives: learning attractor models for motor behaviors. *Neural computation*, 25(2):328–373, 2013.

[Majumdar and Tedrake, 2016] Anirudha Majumdar and Russ Tedrake. Funnel libraries for real-time robust feedback motion planning. *arXiv preprint arXiv:1601.04037*, 2016.

[Reiner *et al.*, 2014] Benjamin Reiner, Wolfgang Ertel, Heiko Posenauer, and Markus Schneider. Lat: A simple learning from demonstration method. In *Intelligent Robots and Systems (IROS), 2014 IEEE/RSJ International Conference on*, pages 4436–4441. IEEE, 2014.

[Schneider and Ertel, 2010] Markus Schneider and Wolfgang Ertel. Robot learning by demonstration with local gaussian process regression. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 255–260. IEEE, 2010.

[Shani and Ballard, 1984] Uri Shani and Dana H Ballard. Splines as embeddings for generalized cylinders. *Computer Vision, Graphics, and Image Processing*, 27(2):129–156, 1984.

[Shanmugavel *et al.*, 2007] Madhavan Shanmugavel, Antonios Tsourdos, Rafał Zbikowski, and Brian 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*, pages 20–23, 2007.

[Vijayakumar *et al.*, 2005] Sethu Vijayakumar, Aaron D'souza, and Stefan Schaal. Incremental online learning in high dimensions. *Neural computation*, 17(12):2602–2634, 2005.