2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Human-Guided Planning for Complex Manipulation Tasks Using the Screw Geometry of Motion

Dasharadhan Mahalingam and Nilanjan Chakraborty

***Abstract*— In this paper, we present a novel method of motion planning for performing complex manipulation tasks by using human demonstration and exploiting the screw geometry of motion. We consider complex manipulation tasks where there are constraints on the motion of the end effector of the robot. Examples of such tasks include opening a door, opening a drawer, transferring granular material from one container to another with a spoon, and loading dishes to a dishwasher. Our approach consists of two steps: First, using the fact that a motion in the task space of the robot can be approximated by using a sequence of constant screw motions, we segment a human demonstration into a sequence of constant screw motions. Second, we use the segmented screws to generate motion plans via screw-linear interpolation for other instances of the same task. The use of screw segmentation allows us to capture the invariants of the demonstrations in a coordinate-free fashion, thus allowing us to plan for different task instances from just one example. We present extensive experimental results on a variety of manipulation scenarios showing that our method can be used across a wide range of manipulation tasks.**

I. Introduction

The use of robot manipulators in service industry, domestic environments, as well as in assistive robotics to help people who have lost the use of hands, depends on the ability of the robots to perform *complex manipulation tasks*. Complex manipulation tasks, as opposed to simple reaching or pick-and-place tasks, are characterized by the presence of constraints on the motion of the end effector of the robot. Figure 1 shows some typical complex manipulation tasks, where the constraints can arise from the mechanical structure of the objects or from the nature of the task itself. Operating articulated objects (like opening/closing doors, windows, drawers, and bottle caps) imposes constraints on the motion of the robot end effector depending on the type of the joint (like revolute, prismatic, and helical).

Furthermore, tasks like scooping and pouring, loading a dish into a dishwasher rack, requires constraints on the end effector motion. These constraints essentially characterize the task (please see the results section for a more detailed discussion) and the constraints may change during the motion. For example, in the scooping and pouring task in Figure 1, the spoon should not rotate while transferring, it should not translate while pouring and during scooping there is a different constraint on the motion which is hard to describe.

The authors are with the Department of Mechanical Engineering at Stony Brook University (SBU), Stony Brook, NY 11794, USA. {dasharadhan.mahalingam, nilanjan.chakraborty}@stonybrook.edu. This work was supported in part by NSF award CMMI 1853454, a Stony Brook OVPR Seed Grant, and a SUNY research grant.

Fig. 1: Typical Complex Manipulation Tasks: (From left to right) Manipulation of object constrained by a revolute joint; Manipulation of object constrained by a prismatic joint; Scooping and pouring; Arranging dishes in a dish rack.

Typically, for the robot to perform the task, a robotics expert needs to come up with a mathematical representation of these constraints (which may not always be easy, think about the scooping task) and they need to be pre-programmed. However, many modern light weight robots are designed with easily accessible tools to allow a non-expert in robotics to give a kinesthetic demonstration, i.e., hold the hand of the robot and show how to do a task. Any constraint that characterizes the task is *embedded in these demonstrations*. Therefore, in principle, it is possible to utilize these constraints (even from a single example) for motion planning for a different instance of the task. Note that an instance of a task, or task instance for short, is defined by the poses of task-related objects. Different task instances correspond to different poses of the task-related objects for the same task. *The goal of this paper is to develop a method that can utilize the embedded task constraints in a single kinesthetic demonstration of a task from a human to plan motions for a different instance of the same task.*

The end-effector motion of a robot is a curve in the group of rigid body motions, i.e., $SE(3)$. Our approach exploits the following characteristic of rigid body motion in $SE(3)$ which is implied by Chasles theorem: *Any curve in $SE(3)$, which represents a rigid body motion, can be approximated arbitrarily closely by a sequence of constant screw motions (or one-parameter subgroups of $SE(3)$).* Note that when the end-effector motion is constrained by a mechanical joint, its motion is a single constant screw motion, i.e., the motion is constrained to lie in a one-parameter subgroup of $SE(3)$ [1].

Based on the above observation, we present a novel two-step solution approach: (a) Given a kinesthetic demonstration of a task, we segment the motion of the end effector in the task space into a sequence of constant screws. (b) For the new task instance, using the segmented screws, we compute a motion plan based on Screw linear interpolation (ScLERP) which automatically ensures that the constant screw constraints embedded in the demonstrated motion are satisfied. The sequence of constant screws represent the task constraints in a coordinate invariant manner, i.e., it does

not depend on the location of the reference frame on the end-effector. By extracting the task constraints which are enforced either due to the presence of joints or due to the inherent nature of the task, we are able to generalize a single demonstration to different task instances. This two-step motion planning method is the *key contribution* of this paper. We also provide extensive experimental results that validates the ability of our approach to generalize from a single example. The key novelty of our method is the combination of data with classical ideas in screw theory that exploits the natural kinematic structure of motion. We show that exploiting the structure of motion can lead to data-efficient approaches for planning using human demonstration as a guide.

II. Related Work

The user-guided manipulation task planning approach presented in this paper is based on the hypothesis that, in principle, even one successful (kinesthetic) demonstration can be used to either implicitly or explicitly encode the constraints that characterize the task and generate motion plans for new instances of a task. Coordinate free shape descriptors of rigid body motion that compactly describe a given demonstration on an object and transfer to new instances with new initial and goal pose of the object is proposed in [2], [3]. However, the constraints present in the tasks are not explicitly extracted (i.e., the motion invariants are not the constraints). These methods [2], [3] are restricted to motions in $SE(3)$, and no corresponding joint space path is reported. No real robot experiments are reported and although they can be used for tasks like pouring where there is a single constraint, for more complex tasks like scooping and pouring (that we consider in this paper) where the task constraints change during the motion, it is not clear that one can use these methods directly.

Another approach to generate motions from a single demonstration is the use of Dynamical movement primitives (DMPs) [4]–[7]. While this is an elegant bio-inspired approach, DMPs again do not consider task constraints explicitly. Furthermore, they are not coordinate-invariant, i.e., their performance depend on the coordinate system attached to the end-effector, which does not allow them to generalize to regions of the task space that are not near the demonstration [3]. In contrast, our method explicitly extracts the task constraints which are expressed as a sequence of constant screw motions that are coordinate-invariant.

Probabilistic approaches to Learning from Demonstration [8], e.g., those using Gaussian Mixture Models (GMM) [9], [10] or Hidden Markov Models (HMM) [11]–[14] and policy learning approaches, e.g., [15], [16] require multiple demonstrations. Trajectory segmentation based approaches that decompose complex movements as a sequence of primitive motions such as [17], [18] either require a predefined library of primitive motions or multiple demonstrations of a task to extract the sequence of primitive motions. Approaches to manipulate articulated objects such as [19], [20] try to estimate the kinematic model (prismatic/revolute/rigid) which maximizes the posterior probability of the object pose observations. In [21], the authors use a neural network to predict the screw parameters of articulated objects from observed depth images. However, these methods cannot be extended to manipulation of non-articulated objects where the motion constraints arise from the nature of the tasks, and some of them are also data hungry. In contrast, our approach aims to extract the motion constraints of both articulated and non-articulated object using a single demonstration.

This work builds on our prior work in [22], [23]. The key distinction again is that here we extract the task constraints explicitly as a sequence of constant screw motions, which allows us to perform a more complex set of tasks.

III. Mathematical Preliminaries

We now present the background knowledge on the screw geometry of rigid body motion required for this work.

Dual Quaternions and Rigid Displacements: A dual quaternion $\mathbf{D}$ is defined as $\mathbf{D} = \mathbf{P} + \epsilon\mathbf{Q}$ where $\mathbf{P}, \mathbf{Q}$ are quaternions, and $\epsilon^2 = 0, \epsilon \neq 0$. The conjugate, norm, and inverse of the dual quaternion $\mathbf{D}$ is represented as $\mathbf{D}^* = \mathbf{P}^* + \epsilon\mathbf{Q}^*$, $\|\mathbf{D}\| = \sqrt{\mathbf{D} \otimes \mathbf{D}^*} = \sqrt{\mathbf{P} \otimes \mathbf{P}^* + \epsilon(\mathbf{P} \otimes \mathbf{Q}^* + \mathbf{Q} \otimes \mathbf{P}^*)}$, and $\mathbf{D}^{-1} = \mathbf{D}^*/\|\mathbf{D}\|^2$, respectively. Another definition for the conjugate of $\mathbf{D}$ is represented as $\mathbf{D}^\dagger = \mathbf{P}^* - \epsilon\mathbf{Q}^*$. Multiplication of two dual quaternions $\mathbf{D}_1 = \mathbf{P}_1 + \epsilon\mathbf{Q}_1$ and $\mathbf{D}_2 = \mathbf{P}_2 + \epsilon\mathbf{Q}_2$ are performed as $\mathbf{D}_1 \otimes \mathbf{D}_2 = (\mathbf{P}_1 \otimes \mathbf{P}_2) + \epsilon(\mathbf{P}_1 \otimes \mathbf{Q}_2 + \mathbf{Q}_1 \otimes \mathbf{P}_2)$ where $\otimes$ on the right represents quaternion multiplication. The dual quaternion $\mathbf{D}$ is a *unit dual quaternion* if $\|\mathbf{D}\| = 1$, i.e., $\|\mathbf{P}\| = 1$ and $\mathbf{P} \otimes \mathbf{Q}^* + \mathbf{Q} \otimes \mathbf{P}^* = 0$, and consequently, $\mathbf{D}^{-1} = \mathbf{D}^*$. Unit dual quaternions can be used to represent the group of rigid body displacements $SE(3)$. A rigid body displacement (or transformation) is represented by a unit dual quaternion $\mathbf{D}_T = \mathbf{Q}_R + \frac{\epsilon}{2}\mathbf{Q}_p \otimes \mathbf{Q}_R$ where $\mathbf{Q}_R$ is the unit quaternion corresponding to rotation and $\mathbf{Q}_p = (0, \boldsymbol{p}) \in \mathbb{H}$ corresponds to the translation. Here, we define $\mathbb{D}$ to represent the set of unit dual quaternions.

Screw Displacement: Chasles-Mozzi theorem states that the general Euclidean displacement/motion of a rigid body from the origin $\boldsymbol{I}$ to $\boldsymbol{T} = (\boldsymbol{R}, \boldsymbol{p}) \in SE(3)$ can be expressed as a rotation θ about a fixed axis $\mathcal{S}$, called the *screw axis*, and a translation d along that axis. Plücker coordinates can be used to represent the screw axis by $\boldsymbol{\omega}$ and $\boldsymbol{m}$, where $\boldsymbol{\omega} \in \mathbb{R}^3$ is a unit vector that represents the direction of the screw axis $\mathcal{S}$, $\boldsymbol{m} = \boldsymbol{r} \times \boldsymbol{\omega}$, and $\boldsymbol{r} \in \mathbb{R}^3$ is an arbitrary point on the axis. Thus, the screw parameters are defined as $\boldsymbol{\omega}, \boldsymbol{m}, h, \theta$, where h is the pitch of the screw and and θ is its magnitude. In general, $h \geq 0$, with $h = \infty$ with θ replaced by d for pure translation. **A *constant screw motion* is a motion where the parameters $\boldsymbol{\omega}, \boldsymbol{m}$, and h stays constant throughout the motion**.

The screw displacements can be expressed by the dual quaternions as $\mathbf{D}_T = \mathbf{Q}_R + \frac{\epsilon}{2}\mathbf{Q}_p\mathbf{Q}_R = (\cos\frac{\Phi}{2}, L\sin\frac{\Phi}{2})$ where $\Phi = \theta + \epsilon d$ is a dual number and $L = \boldsymbol{\omega} + \epsilon\boldsymbol{m}$ is a dual vector. A power of the dual quaternion $\mathbf{D}_T$ is then defined as $\mathbf{D}_T^\tau = (\cos\frac{\tau\Phi}{2}, L\sin\frac{\tau\Phi}{2})$, $\tau > 0$.

Screw Linear Interpolation (ScLERP): To perform a one

degree-of-freedom smooth screw motion (with a constant rotation and translation rate) between two object poses in $SE(3)$, screw linear interpolation (ScLERP) can be used. The ScLERP provides a *straight line* in $SE(3)$ which is the closest path between two given poses in $SE(3)$. If the poses are represented by unit dual quaternions $\mathbf{D}_1$ and $\mathbf{D}_2$, the path provided by the ScLERP is derived by $\mathbf{D}(\tau) = \mathbf{D}_1 \otimes (\mathbf{D}_1^{-1} \otimes \mathbf{D}_2)^\tau$ where $\tau \in [0,1]$ is a scalar path parameter. As τ increases from 0 to 1, the object moves between two poses along the path $\mathbf{D}(\tau)$ by the rotation $\tau\theta$ and translation τd. Let $\mathbf{D}_{12} = \mathbf{D}_1^{-1} \otimes \mathbf{D}_2$. To compute $\mathbf{D}_{12}^\tau$, the screw coordinates $\boldsymbol{\omega}, \boldsymbol{m}, h, \theta$ are first extracted from $\mathbf{D}_{12} = \mathbf{P} + \epsilon\mathbf{Q} = (p_0, \boldsymbol{p}_r) + \epsilon(q_0, \boldsymbol{q}_r) = (\cos\frac{\theta}{2}, \boldsymbol{\omega}\sin\frac{\theta}{2}) + \epsilon\mathbf{Q}$ by $\boldsymbol{\omega} = \boldsymbol{p}_r/\|\boldsymbol{p}_r\|$, $\theta = 2\,\mathrm{atan2}(\|\boldsymbol{p}_r\|, p_0)$, $d = \boldsymbol{p}\cdot\boldsymbol{\omega}$, and $\boldsymbol{m} = \frac{1}{2}(\boldsymbol{p}\times\boldsymbol{\omega} + (\boldsymbol{p} - d\boldsymbol{\omega})\cot\frac{\theta}{2})$ where $\boldsymbol{p}$ is derived from $2\mathbf{Q}\mathbf{P}^* = (0, \boldsymbol{p})$ and $\mathrm{atan2}(\cdot)$ is the two-argument arctangent. Then, $\mathbf{D}_{12}^\tau = (\cos\frac{\tau\Phi}{2}, L\sin\frac{\tau\Phi}{2})$ is directly derived from $\left(\cos\frac{\tau\theta}{2}, \sin\frac{\tau\theta}{2}\boldsymbol{\omega}\right) + \epsilon\left(-\frac{\tau d}{2}\sin\frac{\tau\theta}{2}, \frac{\tau d}{2}\cos\frac{\tau\theta}{2}\boldsymbol{\omega} + \sin\frac{\tau\theta}{2}\boldsymbol{m}\right)$. Note that $h = \infty$ corresponds to pure translation.

Distance Metric in SE(3): The heuristic which we follow to determine the distance between two rigid body transformations in space is, to determine the distance in position and orientation separately. Let $\mathbf{D}_1 = \mathbf{Q}_1 + \frac{\epsilon}{2}\mathbf{p}_1 \otimes \mathbf{Q}_1$ and $\mathbf{D}_2 = \mathbf{Q}_2 + \frac{\epsilon}{2}\mathbf{p}_2 \otimes \mathbf{Q}_2$ be two poses in dual quaternion representation.

- The distance in position between two poses is given by the Euclidean norm, $\boldsymbol{d}_p(\mathbf{D}_1, \mathbf{D}_2) = ||\mathbf{p}_1 - \mathbf{p}_2||$.
- The distance in orientation between the two poses is, $\boldsymbol{d}_\phi(\mathbf{D}_1, \mathbf{D}_2) = min\,(||\mathbf{Q}_1 - \mathbf{Q}_2||, ||\mathbf{Q}_1 + \mathbf{Q}_2||)$ as defined in [24].

Neighbourhood of a pose: For any pose represented using unit dual quaternions as $\mathbf{D}$, we define $\{\mathbf{D}' \in \mathbb{D} | \boldsymbol{d}_{\boldsymbol{p}}(\mathbf{D}, \mathbf{D}') \leq \varepsilon_p, \boldsymbol{d}_{\boldsymbol{\phi}}(\mathbf{D}, \mathbf{D}') \leq \varepsilon_\phi\}$ to be its $(\varepsilon_p, \varepsilon_\phi)-$neighbourhood. The parameters ε_p and ε_ϕ define the size of the neighbourhood.

IV. Problem Statement

We have a kinesthetic demonstration of a manipulation task where the joint angles of the robot are recorded (see Figures 2 and 4). Using the forward kinematics map, we can obtain the sequence of poses the end effector goes through to accomplish the task. Let $\mathcal{D} = \{\mathbf{D}_1, \mathbf{D}_2, \cdots, \mathbf{D}_n\}$ be the sequence of end effector poses and $\mathcal{O} = \{\mathbf{O}_1, \mathbf{O}_2, ..., \mathbf{O}_v\}$ be the poses of the task relevant objects written using a unit dual quaternion representation of $SE(3)$. The sequence of end effector poses in $\mathcal{D}$ is the task space representation of the demonstrated path. Any task-relevant constraints that characterize the manipulation task are implicitly present in $\mathcal{D}$. ***Given,*** $(\mathcal{D}, \mathcal{O})$***, i.e., a task demonstration and the poses of the task-relevant objects in the demonstration, compute the motion plan required to perform a new task instance with initial end-effector pose*** $(\mathbf{D}_1')$***, final pose*** $(\mathbf{D}_n')$***, and the task-relevant object poses*** $(\mathcal{O}')$***, such that, the task-relevant constraints are satisfied.*** For manipulation of articulated objects both the sets $\mathcal{O}$ and $\mathcal{O}'$ are empty and the problem reduces to: ***Compute the motion plan for the task when given*** $\mathcal{D}$***, a new*** $\mathbf{D}_1'$ ***and required magnitude of motion*** θ'.

To accomplish the above, we use a screw geometry-based representation of the underlying demonstrated motion. In particular, we divide the problem above into two sub-problems.

Screw Segmentation Problem: Given a demonstrated motion in task space and the poses of the task related objects, segment the motion as a sequence of constant screw motions.

Motion Generation Problem using Segmented Demonstration: Given a sequence of constant screw motions that is known to accomplish a given task, for a new task instance of the same task (defined by the initial pose and goal pose of the end effector as well as the poses of the task-relevant objects), compute a motion plan that accomplishes the task instance using the sequence of constant screw motions that was determined from the demonstration.

The rationale for using a screw segmentation of the motion is two-fold. First, from Chasles' theorem it can be inferred that any path in $SE(3)$ can be approximated arbitrarily closely as a sequence of constant screw motions. This is analogous to the fact that any curve in $\mathbb{R}^3$ can be approximated arbitrarily closely by a sequence of straight line segments. Second, the screw representation is a coordinate-invariant representation (meaning that it does not depend on the choice of the coordinate frame at the end effector of the robot) and also maps the motion to a single parameter subgroup of $SE(3)$, which potentially allows better generalization properties.

V. Screw Segmentation Problem

In this section, we will discuss our solution for the screw segmentation problem. Let $\mathcal{D} = \{\mathbf{D}_1, \mathbf{D}_2, ..., \mathbf{D}_n\}$ be a sequence of recorded end-effector poses in unit dual quaternion representation. As stated earlier, we can represent $\mathcal{D}$ as a sequence of constant screw displacements $\boldsymbol{\delta}_1, \boldsymbol{\delta}_2, ..., \boldsymbol{\delta}_u$ applied to the initial pose $\mathbf{D}_1$ where $1 \leq u < n$, or as a sequence of poses $\{\mathbf{E}_1, \mathbf{E}_2, ..., \mathbf{E}_u\} \subset \mathcal{D}$, where $\mathbf{E}_1 = \boldsymbol{\delta}_1 \otimes \mathbf{D}_1$ and $\mathbf{E}_i = \boldsymbol{\delta}_i \otimes \mathbf{E}_{i-1}$ for $i = 2, ..., u$. Here, $\mathbf{E}_i$ represents the end pose of each constant screw segment (in sequence) and u is the number of constant screws present in the motion $\mathcal{D}$.

Checking if a motion segment is constant screw: We now present a method to check if a consecutive sequence of poses, $\mathcal{D}' = \{\mathbf{D}_i, \mathbf{D}_{i+1}, ..., \mathbf{D}_{j-1}, \mathbf{D}_j\} \subset \mathcal{D}$, belong to a constant screw segment, and, if so, extract the screw parameters. The relative transformation between the initial pose $\mathbf{D}_i$ and any other pose on the recorded sequence is given by,

$$\mathbf{D}_{ik} = \mathbf{D}_k \otimes \mathbf{D}_i^*,\ k = i+1, \cdots, j. \tag{1}$$

where $\mathbf{D}_{ik}$ is of the form $\mathbf{D}_{ik} = \mathbf{Q}_{ik} + \frac{\epsilon}{2}\mathbf{p}_{ik} \otimes \mathbf{Q}_{ik}$. Let the motion be generated by the screw $(\boldsymbol{\omega}_{ij}, \mathbf{m}_{ij}, h_{ij}, \theta_{ij})$. If the observation is noiseless, for any motion that is not pure translation, $\boldsymbol{\omega}_{ij}$ and θ_{ij} can be obtained from

$$\mathbf{Q}_{ij} = \left(\cos\frac{\theta_{ij}}{2}, \boldsymbol{\omega}\sin\frac{\theta_{ij}}{2}\right) \tag{2}$$

by using the standard procedure to obtain a rotation axis and angle from a unit quaternion. We obtain $\boldsymbol{v}_{ij}$ using

$$\boldsymbol{v}_{ij} = \left[(\mathbf{I} - e^{\hat{\boldsymbol{\omega}}_{ij}\theta})\hat{\boldsymbol{\omega}}_{ij} + \theta_{ij}\boldsymbol{\omega}_{ij}\boldsymbol{\omega}_{ij}^T\right]^{-1}\mathbf{p}_{ij} \tag{3}$$

Fig. 2: MANIPULATION OF ARTICULATED OBJECTS: User Provided Kinesthetic Demonstration (Row 1 from Left to Right) : Revolute Joint Demo #3 Start Pose (Revolute Joint Closed Position), Revolute Joint Demo #3 End Pose (Revolute Joint Open Position), Prismatic Joint Demo #3 Start Pose (Prismatic Joint Open Position), Prismatic Joint Demo #3 End Pose (Prismatic Joint Closed Position); **Grasping Pose Changes across Trials** (Row 2 from Left to Right) : Changes in position and orientation of how the end-effector grasps the object across conducted Trials when using Demo #3 for Revolute Joint (Image #1 - Trial #2, Image #2 - Trial 4) and when using Demo #3 for Prismatic Joint (Image #3 - Trial #3, Image #4 - Trial #4).

where the 3×3 matrix $\left[(\mathbf{I} - e^{\hat{\boldsymbol{\omega}}_{ij}\theta_{ij}})\hat{\boldsymbol{\omega}}_{ij} + \theta_{ij}\boldsymbol{\omega}_{ij}\boldsymbol{\omega}_{ij}^T\right]$ is always invertible. The pitch, h_{ij}, is then

$$h_{ij} = \boldsymbol{\omega}_{ij}^T \boldsymbol{v}_{ij} \tag{4}$$

and we get $\mathbf{m}_{ij}$ by using $\mathbf{m}_{ij} = \boldsymbol{v}_{ij} - h_{ij}\boldsymbol{\omega}_{ij}$. If $h_{ij} = 0$, then the motion is pure rotation. For pure translation, $h = \infty$ and $\boldsymbol{m} = \mathbf{0}$ by definition. We obtain $\boldsymbol{\omega}$ and θ from

$$\boldsymbol{\omega} = \frac{\mathbf{p}_{ij}}{\|\mathbf{p}_{ij}\|}, \qquad \theta = \|\mathbf{p}_{ij}\| \tag{5}$$

Ideally, if the sequence $\mathcal{D}'$ is generated by the unit constant screw, $(\boldsymbol{\omega}_{ij}, \mathbf{m}_{ij}, h_{ij})$, then $\forall\ i \leq k \leq j$, there is a $\theta_{ik} = \tau\theta_{ij}$, $0 \leq \tau \leq 1$, such that, $\mathbf{D}_k$ is obtained by ScLERP between $\mathbf{D}_i$ and $\mathbf{D}_j$. But, due to the presence of noise, $\mathbf{D}_k$ does not lie on the ideal screw motion from $\mathbf{D}_i$ to $\mathbf{D}_j$. However, using line search on the parameter τ, $0 \leq \tau \leq 1$ and ScLERP, we can determine a $\theta_{ik} = \tau\theta_{ij}$ which gives us a $\mathbf{D}'_k$ that lies on the screw motion from $\mathbf{D}_i$ to $\mathbf{D}_j$ and within a neighbourhood $(\varepsilon_p, \varepsilon_\phi)$ of the pose $\mathbf{D}_k$. If no such θ_{ik} exists then $\mathbf{D}_k$ does not lie on the screw motion from $\mathbf{D}_i$ to $\mathbf{D}_j$ and the motion segment $\mathcal{D}'$ is not a constant screw motion. Thus, the approach above can be used to check (1) if the motion is a constant screw motion (2) if the motion is pure translation or general screw motion and (3) extract the screw parameters for both if the motion is a constant screw motion.

Note that although we perform a point estimate using $\mathbf{D}_{ij}$ to determine the screw parameters, we ensure that any pose from $\mathcal{D}'$ lies within the neighborhood $(\varepsilon_p, \varepsilon_\phi)$ from the computed constant screw. Thus, our approach is robust to noise in the data, and correctly predicts if the motion is a constant screw (and also the corresponding parameters), within an error tolerance. Because of noise, pure translation will never have $h = \infty$. Therefore, in implementation, we have to first check if the motion is pure translation within the rotational error tolerance, ε_ϕ, and if it is, we use Equation 5 to obtain the screw parameters.

Extracting constant screw segments: Algorithm 1 uses the discussion above[1] to segment the demonstration $\mathcal{D}$. Starting from the initial pose $\mathbf{D}_1$, we search sequentially in $\mathcal{D}$, to find the first pose $\mathbf{D}_j$ such that $\{\mathbf{D}_1, \cdots \mathbf{D}_j\}$ is not a constant screw segment, but $\{\mathbf{D}_1, \cdots \mathbf{D}_{j-1}\}$ is a constant screw segment. Then we repeat the search for a constant screw segment from $\mathbf{D}_j$ onward and stop when $j = n$ (lines 4 to 10). Our implementation of the screw segmentation algorithm takes approximately 10 seconds for segmenting the demonstrations on a Intel i9 processor with 32 GB RAM.

Algorithm 1: Segment given motion into a sequence of constant screws

```
1  def getScrewSegments(D, ε_p, ε_φ):
2      Set u = 1, i = 1, E_u = D_n
3      while i <= n do
4          for j = i + 1 to n do
5              D' = {D_i, ..., D_j}
6              if getScrewParameters(D', ε_p, ε_φ)
                 returns ω, m, θ, h then
7                  continue
8              else
9                  E_u = D_{j-1},  u = u + 1,  i = j
10                 break
11     return {E_1, E_2, ..., E_u}
```

The output of Algorithm 1 is the segmentation, $\{\mathbf{E}_1, \mathbf{E}_2, ..., \mathbf{E}_u\}$ of the motion in $\mathcal{D}$. Note that if $u = 1$, we get a single screw segment, which occurs for articulated objects like a door or a drawer. This sequence of constant screws describe the task constraints in a coordinate invariant manner, i.e., it does not depend on the choice of the end effector coordinate frame. By extracting the task constraints that arise either due to the presence of joints or due to the inherent nature of the task, we are able to generalize a single demonstration to different task instances using the planning method described in Section VI.

[1]The subroutine `getScrewParameters` $(\mathcal{D}', \varepsilon_p, \varepsilon_\phi)$ in line 6 implements this check. Due to space constraints we do not show the subroutine in an algorithm format, which is given in the extended version [25]

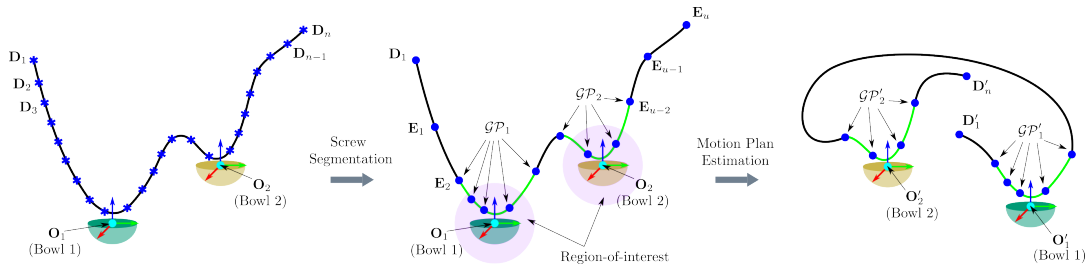

Fig. 3: SCHEMATIC SKETCH FOR MOTION ESTIMATION OF COMPLEX TASKS: Overview of motion estimation for a scoop and pour task. The task involves scooping contents from Bowl 1 with a spoon and pouring it into Bowl 2. The $SE(3)$ poses are represented as points to reduce clutter; **Left** - The provided demonstration $\mathcal{D}$ consisting of a sequence of $SE(3)$ poses along with pose of the objects $\mathbf{O}_1$ (Bowl 1) and $\mathbf{O}_2$ (Bowl 2); **Center** - Segmenting the provided demonstration $\mathcal{D}$ into a sequence of constant screws $\{\mathbf{D}_1, \mathbf{E}_1, ..., \mathbf{E}_u\}$ and determining the **Key Segments** (Coloured Green) associated with each object. The task-relevant constraints $\mathcal{GP}_1$ (Bowl 1), $\mathcal{GP}_2$ (Bowl 2) are identified from the Key Segments; **Right** - Computing the motion plan for new initial pose $\mathbf{D}'_1$, final pose $\mathbf{D}'_2$ and object poses $\mathbf{O}'_1$ (Bowl 1) and $\mathbf{O}'_2$ (Bowl 2) by determining the **Guiding Poses** $\mathcal{GP}'$ utilizing the task-relevant constraints (Coloured Green) obtained from the previous step.

VI. MOTION GENERATION FROM SEGMENTED DEMONSTRATION

We will now describe our motion generation method for tasks involving manipulation of articulated objects as well as non-articulated objects, using the segmented demonstration.

Manipulation of Articulated Objects: For articulated objects (like a door/drawer), whose motion is constrained by a joint, the segmented screw from the demonstration consists of a single screw which is the screw corresponding to the joint axis. Let $(\boldsymbol{\omega}, \mathbf{m}, h)$ be the unit screw obtained from the demonstration. Given a new initial pose $\mathbf{D}'_1$ of the end-effector and magnitude of motion θ', we can use the screw parameters to compute the screw displacement $\boldsymbol{\delta}' = (\cos\frac{\theta'+\epsilon h\theta'}{2}, (\boldsymbol{\omega} + \epsilon\mathbf{m})\sin\frac{\theta'+\epsilon h\theta'}{2})$ and the new final configuration $\mathbf{D}'_n = \boldsymbol{\delta}' \otimes \mathbf{D}'_1$. The motion plan from $\mathbf{D}'_1$ to $\mathbf{D}'_n$ for manipulating the articulated object is computed using the ScLERP motion planner [1] so that the motion satisfies the screw constraint. Our approach does not need *a priori* information about the articulation model to estimate the joint constraints. Also, ScLERP will generate end-effector poses that satisfies the constraint imposed by the joint without explicitly considering it [1]. The direction of motion can be changed by appropriately choosing the sign of θ'.

Manipulation Tasks for Non-Articulated Objects: Recall that we have a sequence of constant screws $\mathcal{E} = \{\mathbf{E}_1, \mathbf{E}_2, ..., \mathbf{E}_u\}$, $u \leq (n-1)$ (each consecutive pair $(\mathbf{E}_i, \mathbf{E}_{i+1})$, $i = 1, \cdots, n-1$, gives a constant screw) along with the task-relevant object poses $\{\mathbf{O}_1, \mathbf{O}_2, \cdots, \mathbf{O}_v\}$. Heuristically, we define the task-relevant constraints as ***the sequence of screw segments obtained from the provided demonstration such that they lie inside a region-of-interest surrounding the task-related objects and expressed with respect to the local frame of reference of the associated task-related object***. Each task-relevant object has its own region-of-interest and a sequence of screws associated with it. Let us refer to the screw segments that lie inside a region-of-interest as ***key segments***. Here, we define the region-of-interest of a task-related object to be a spherical volume or a cuboid centered at the object frame with dimensions that is set based on the type of the object. However it can also be chosen to be any other shape. The key-segments associated with task related object, i, is denoted as $\{\mathbf{G}_{i,1}, \mathbf{G}_{i,2}, ..., \mathbf{G}_{i,m_i}\} \subset \mathcal{E}$, where m_i is the number of key-segments associated with the object. The task-relevant constraints for object i expressed in its local frame of reference are then computed as,

$$\mathcal{GP}_i = \{\mathbf{O}_i^* \otimes \mathbf{G}_{i,1}, \mathbf{O}_i^* \otimes \mathbf{G}_{i,2}, ..., \mathbf{O}_i^* \otimes \mathbf{G}_{i,m_i}\} \quad (6)$$

Given the new poses of the task related objects, $\{\mathbf{O}'_1, \mathbf{O}'_2, ..., \mathbf{O}'_v\}$, for each object i, the task-relevant constraints can then be transferred from (6) as,

$$\mathcal{GP}'_i = \{\mathbf{O}'_i \otimes \mathbf{O}_i^* \otimes \mathbf{G}_{i,1}, ..., \mathbf{O}'_i \otimes \mathbf{O}_i^* \otimes \mathbf{G}_{i,m_i}\} \quad (7)$$

Once the task-relevant constraints for the new task instance have been computed for each object, we construct the sequence of constant screws that the end-effector should follow to successfully execute the task as,

$$\mathcal{GP}' = \{\mathbf{D}'_1, \mathcal{GP}'_1, \mathcal{GP}'_2, ..., \mathcal{GP}'_v, \mathbf{D}'_n\} \quad (8)$$

Each element of $\mathcal{GP}'$ is called a ***Guiding Pose***. Figure 3 gives a schematic illustration of the entire process of computing the guiding poses for the scooping and pouring task. Using the ScLERP Motion Planner [1] to sequentially move through each pose in $\mathcal{GP}'$ ensures that the task-relevant constraints are satisfied while the motion is being executed. The ScLERP Motion Planner is realtime and runs at 1 kHz frequency.

VII. EXPERIMENTAL RESULTS

To evaluate our approach, we conducted a total of 118 experimental trials across four tasks: (1) manipulation of an object constrained by a revolute joint, (2) manipulation of an object constrained by a prismatic joint, (3) scooping and pouring, and (4) arranging dishes in a rack. The experiments are performed using the Baxter robot from Rethink Robotics [26]. Additional information on the experiments have been included in the accompanying video.

Manipulation of Articulated Objects: For each of task 1 and 2, which involves the manipulation of articulated objects, we test our algorithm on 3 different demonstrations and 13 experimental trials for each demonstration. The gripper held the object in a different place with a different pose across all the 78 trials and 3 demonstrations. The purpose of varying the gripper poses is to show that, practically, the coordinate invariance implies that irrespective of how we hold the object, we can use the extracted screw from just a

single demonstration to manipulate the object. Recall that we had two parameters in our segmentation algorithm, namely, ε_p and ε_ϕ. Across all our demonstrations, we set the values of the parameters as $\varepsilon_p = 1$ cm and $\varepsilon_\phi = 0.1$.

(1) Object constrained by a Revolute Joint: The object used for this task is a wooden block constrained to rotate about a revolute joint (see Figure 2). Although each of the demonstrations was for opening the wooden block, for each demonstration, the robot was made to perform the task of both opening (8 times) and closing (5 times) it by an angle of $45°$ by holding the handle at different poses. All the 39 trials were successful. Furthermore, the estimate of the constant screw, which corresponds to the axis of the revolute joint was close for all the demonstrations (the difference being due to the clearance in the revolute joint and the noise in the robot encoders). Please see the extended version for the detailed numerical estimates [25].

(2) Object constrained by a Prismatic Joint: The object used for this task is a wooden block constrained to translate along a prismatic joint axis (see Fig. 2). Among the 3 demonstrations, the first two were for opening the block while the third demonstration was for closing the block from the open position. For each demonstration, opening was performed 8 times and closing the block was performed 5 times, for a distance of 30 cm with a random grasping pose for each trial (See Figure 2). Here again, all the experimental trials were successful, which highlights the importance of using the screw geometry of motion.

Manipulation Tasks for Non-Articulated Objects: Here, the values of the screw segmentation parameters was set to be $\varepsilon_p = 1$ cm and $\varepsilon_\phi = 0.15$ for all demonstrations.

(1) Scooping and Pouring: In this task the robot scoops rice from one bowl using a spoon which the robot is already grasping and pours it into another bowl which is placed at a different position (see Figure 4). The region of interest used for obtaining the key segments was chosen to be a sphere of radius 20 cm for both bowls.

There were 2 different demonstrations and for each demonstration 8 trials were conducted making a total of 16 experimental trials. Each trial was conducted by varying pose of one or both the bowls. In particular, the heights of the bowls were also changed among some trials, so that merely mimicking the demonstrated path would result in a collision. During each trial, it was ensured that the level of rice in the bowl was the same as in the demonstration. Due to the lack of perception, our framework cannot account for the level of rice in the bowl. All of the trials resulted in successful scooping and transfer of rice from one bowl to another.

(2) Arranging Dishes: In this task we pick a plate placed on the table and insert it into one of the slots of a dish rack that is placed on the table. This exemplar task was chosen because it consists of a challenging collision avoidance problem when the dish has to be placed in the dish rack. The geometry of the dishes and the slots of the dish rack are such that putting in the dishes top down vertically would result in the dish getting jammed and not reaching the bottom of the rack. The dish has to be put in at an angle and moved in a way that is hard to describe in words or write equations for, but easier for a human to demonstrate. There were 3 different demonstrations provided and for each demonstration, we performed 8 experimental trials. Thus, a total of 24 experimental trials were performed. Between different trials, the pose of the dish and the dish rack, as well as the slot in which the dish is being inserted was varied. To extract the key segments the region of interest were chosen to be a sphere of radius 15 cm for the dish and a cube of side 45 cm for the dish rack. Also, the grasping information, i.e., where the gripper should be closed relative to the pose of the object is determined from the demonstration and is used to grasp the object during execution.

Fig. 4: COMPLEX MANIPULATION TASKS: Scoop and Pour Task (Row 1 from Left to Right) - Demo #1 (Images 1 & 2), Trial #8 using Demo #1 (Images 3 & 4); **Arranging Dishes Task** (Row 2 from Left to Right) - Demo #3 (Images 1 & 2), Trial #8 using Demo #3 (Images 3 & 4).

To summarise the conducted trials, there was one failure among the trials conducted using demonstration 1 and three failures among the trials conducted using demonstration 3. All the trials conducted using demonstration 2 were successful. All the failures are related to the joint accuracy limits of Baxter. Due to these inaccuracies, the plate was either placed in the neighbouring slot instead of the intended slot or hit the raised edges of the rack and got stuck.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we combined classical ideas in screw theory with kinesthetic demonstration to present a novel approach for complex manipulation planning that extracts coordinate-invariant representation of the manipulation constraints from the demonstrated instance, and use them to plan motions for new task instances. We show extensive experimental results on challenging manipulation tasks to show the usefulness of our method. The expression of task constraints as a sequence of constant screws would potentially allow us to provide demonstrations on one robot and execute the task on another robot with completely different hardware architectures.

Currently, our method does not take joint limits of the manipulator or collision avoidance into consideration. There may be obstacles on the interpolated path between the guiding poses that the end-effector needs to follow, or there may be obstacles that might come in contact with the robot links. While the latter case can be overcome by integrating obstacle avoidance in the ScLERP based motion planner as proposed in [27], the former case requires modifying the task constraints. In future work we plan on incorporating collision avoidance with the current approach.

REFERENCES

[1] A. Sarker, A. Sinha, and N. Chakraborty, "On screw linear interpolation for point-to-point path planning," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 9480–9487.

[2] J. De Schutter, "Invariant Description of Rigid Body Motion Trajectories," *Journal of Mechanisms and Robotics*, vol. 2, no. 1, 11 2009, 011004. [Online]. Available: https://doi.org/10.1115/1.4000524

[3] M. Vochten, T. De Laet, and J. De Schutter, "Generalizing demonstrated motion trajectories using coordinate-free shape descriptors," *Robotics and Autonomous Systems*, vol. 122, p. 103291, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889019303288

[4] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: Learning attractor models for motor behaviors," *Neural Comput.*, vol. 25, no. 2, pp. 328–373, Feb. 2013. [Online]. Available: http://dx.doi.org/10.1162/NECO_a_00393

[5] M. Hersch, F. Guenter, S. Calinon, and A. Billard, "Dynamical system modulation for robot learning via kinesthetic demonstrations," *IEEE Transactions on Robotics*, vol. 24, no. 6, pp. 1463–1467, Dec 2008.

[6] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *2009 IEEE International Conference on Robotics and Automation*, May 2009, pp. 763–768.

[7] M. Saveriano, F. Franzel, and D. Lee, "Merging position and orientation motion primitives," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 7041–7047.

[8] A. G. Billard, S. Calinon, and R. Dillmann, *Learning from Humans*. Cham: Springer International Publishing, 2016, pp. 1995–2014. [Online]. Available: https://doi.org/10.1007/978-3-319-32552-1_74

[9] 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.

[10] C. Pérez-D'Arpino and J. A. Shah, "C-learn: Learning geometric constraints from demonstrations for multi-step manipulation in shared autonomy," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4058–4065.

[11] S. Calinon, F. D'halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and reproduction of gestures by imitation," *IEEE Robotics & Automation Magazine*, vol. 17, no. 2, pp. 44–54, 2010.

[12] S. Calinon, A. Pistillo, and D. G. Caldwell, "Encoding the time and space constraints of a task in explicit-duration hidden markov model," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 3413–3418.

[13] S. Niekum, S. Chitta, A. G. Barto, B. Marthi, and S. Osentoski, "Incremental semantically grounded learning from demonstration." in *Robotics: Science and Systems*, vol. 9. Berlin, Germany, 2013, pp. 10–15 607.

[14] E. G. Herrero, J. Ho, and O. Khatib, "Understanding and segmenting human demonstrations into reusable compliant primitives," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 9437–9444.

[15] S. Bahl, A. Gupta, and D. Pathak, "Hierarchical Neural Dynamic Policies," in *Proceedings of Robotics: Science and Systems*, Virtual, July 2021.

[16] A. Gupta, V. Kumar, C. Lynch, S. Levine, and K. Hausman, "Relay policy learning: Solving long horizon tasks via imitation and reinforcement learning," *Conference on Robot Learning (CoRL)*, 2019.

[17] F. Meier, E. Theodorou, F. Stulp, and S. Schaal, "Movement segmentation using a primitive library," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 3407–3412.

[18] S. H. Lee, I. H. Suh, S. Calinon, and R. Johansson, "Autonomous framework for segmenting robot trajectories of manipulation task," *Autonomous Robots*, vol. 38, no. 2, pp. 107–141, Feb 2015. [Online]. Available: https://doi.org/10.1007/s10514-014-9397-9

[19] J. Sturm, V. Pradeep, C. Stachniss, C. Plagemann, K. Konolige, and W. Burgard, "Learning kinematic models for articulated objects," in *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, ser. IJCAI'09. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2009, p. 1851–1856.

[20] J. Sturm, C. Stachniss, and W. Burgard, "A probabilistic framework for learning kinematic models of articulated objects," *J. Artif. Int. Res.*, vol. 41, no. 2, p. 477–526, may 2011.

[21] A. Jain, R. Lioutikov, C. Chuck, and S. Niekum, "Screwnet: Category-independent articulation model estimation from depth images using screw theory," in *arXiv preprint*, 2020.

[22] R. Laha, A. Rao, L. F. C. Figueredo, Q. Chang, S. Haddadin, and N. Chakraborty, "Point-to-Point Path Planning Based on User Guidance and Screw Linear Interpolation," ser. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. Volume 8B: 45th Mechanisms and Robotics Conference (MR), 08 2021, v08BT08A010. [Online]. Available: https://doi.org/10.1115/DETC2021-71814

[23] R. Laha, R. Sun, W. Wu, D. Mahalingam, N. Chakraborty, L. F. Figueredo, and S. Haddadin, "Coordinate invariant user-guided constrained path planning with reactive rapidly expanding plane-oriented escaping trees," in *2022 International Conference on Robotics and Automation (ICRA)*, May 2022, pp. 977–984.

[24] D. Q. Huynh, "Metrics for 3d rotations: Comparison and analysis," *Journal of Mathematical Imaging and Vision*, vol. 35, no. 2, pp. 155–164, 2009.

[25] D. Mahalingam and N. Chakraborty, "Human-guided planning for complex manipulation tasks using the screw geometry of motion," https://arxiv.org/abs/2209.05672, 2022. [Online]. Available: https://arxiv.org/abs/2209.05672

[26] R. Robotics, "Baxter hardware specifications," https://sdk.rethinkrobotics.com/wiki/Hardware_Specifications.

[27] A. Sinha, A. Sarker, and N. Chakraborty, "Task Space Planning With Complementarity Constraint-Based Obstacle Avoidance," ser. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. Volume 8B: 45th Mechanisms and Robotics Conference (MR), 08 2021, v08BT08A012. [Online]. Available: https://doi.org/10.1115/DETC2021-72009

[28] R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[29] P. Mike, V. Hwang, S. Chitta, and M. Likhachev, "Learning to plan for constrained manipulation from demonstrations," in *Proceedings of Robotics: Science and Systems*, Berlin, Germany, June 2013.

[30] H. B. Mohammadi, S. Hauberg, G. Arvanitidis, G. Neumann, and L. D. Rozo, "Learning riemannian manifolds for geodesic motion skills," in *Robotics: Science and Systems*, 2021. [Online]. Available: https://doi.org/10.15607/RSS.2021.XVII.082

[31] B. Akgun, M. Cakmak, K. Jiang, and A. L. Thomaz, "Keyframe-based learning from demonstration," *International Journal of Social Robotics*, vol. 4, no. 4, pp. 343–355, Nov 2012. [Online]. Available: https://doi.org/10.1007/s12369-012-0160-0