2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Dual Quaternion based Compliant Movement Primitives for Deformable Object Manipulation

Amir Samai[2], John Thomas[1], Mohammad Alkhatib[1], Erol Ozgur[1], and Youcef Mezouar[1]

Abstract— **Learning from demonstration effectively transfers human manipulation skills to robots. It can be especially useful for imitating industrial manipulation tasks which are performed by humans and are difficult to model such as deformable object manipulation. Manipulation of deformable objects often requires not only accurate tracking of the demonstration trajectory using a robot end-effector, but also the accommodation of interaction forces. Precise tracking of such trajectories while ignoring these interaction forces leads to overly stiff, unsafe, or unsuccessful executions. We address this problem by proposing *Dual Quaternion based Compliant Movement Primitives* (DQ-CMP). DQ-CMP couples a dual-quaternion based Dynamic Movement Primitive for compact 6-DoF pose encoding with learnable wrench primitives. This combination reproduces synchronized motion and force behaviors directly at the end effector. The method is robot-agnostic and singularity-free at the representation-level, as it operates in operational space using dual quaternions. From a few demonstrations, required wrenches for unseen initial configurations are predicted using Gaussian process regression defined on the pose manifold. This enables generalization of the learned wrenches across different starting poses. We validate the method on real-robot experiments including a shoe-sole detachment for recycling and bending of stiff foam inside a box. Results show compliant, safe task execution and successful generalization to new initial poses.**

Index Terms - **Deformable object manipulation, Detachment task, Compliant Movement Primitives, Dynamic Movement Primitives**

I. INTRODUCTION

Manipulation of deformable objects is a challenging problem due to sensing and modeling difficulties, high-dimensional state space, and complex contact interactions that change with deformation [1], [2], [3]. Model-based strategies such as vision-based methods attempt to infer object shape or fit local or global models such as shape Jacobians or Finite Element Methods [2]. These methods can work well in controlled settings but face practical limitations. Visual perception struggles with occlusions and ambiguous projections under complex deformations [3]. Physics-based models are expensive to build and update online, which often leads to brittle or slow manipulation behaviors in realistic scenarios [1]. These difficulties motivate model-free alternatives that can capture the implicit task knowledge used by humans during manipulation and avoid the need for explicit deformation models.

*This work was supported by Interreg-VI Sudoe and European Regional Development Fund through the REMAIN project (S1/1.1/E0111)

[1] Université Clermont Auvergne, CNRS, Clermont Auvergne INP, Institut Pascal, F-63000 Clermont-Ferrand, john.thomas@sigma-clermont.fr

[2] Université Le Havre Normandie, ISEL, Le Havre

Learning-from-Demonstration (LfD) is a model-free method that sidesteps many of the sensing and modeling challenges by encoding know-how knowledge from expert demonstrations [2]. LfD naturally captures motion timing, sequencing and interaction strategies directly from examples, which simplifies data collection and enables generalization on new tasks or platforms. Most LfD approaches for motion encoding rely on motion primitives such as Dynamic Movement Primitives (DMPs).

DMPs provide stable and tunable representations of desired trajectories [4]. In addition, DMPs formulated with unit dual quaternions allow compact, singularity-free representation of 6-DoF end-effector motion in operational space [5]. However, motion-only approaches are insufficient for many deformable-object tasks where safe and effective execution depends critically on the interaction forces (e.g., detaching a sole from a shoe or bending a stiff foam) [6], [3]. Following a recorded motion with high stiffness risks damaging the object or failing to overcome necessary contact forces. Consequently, LfD methods that reproduce motion alone cannot guarantee compliant, contact-rich manipulation [7].

Compliant Movement Primitives (CMPs) extend motion primitives by adding wrenches components to encode task dynamics. This reproduces jointly motion and interaction forces and enables compliant execution without explicit modeling of environment dynamics [8], [9], [7]. State-of-the-art CMP formulations have shown strong performance in contact-rich tasks, but they are expressed in robot joint-space and are robot-specific [10]. This robot and joint centric formulation complicates data collection, reduces transferability across platforms, and limits direct control over end-effector wrenches in task space [4]. To address these limitations, we propose **DQ-CMP**, a dual-quaternion-based operational-space reformulation of CMP. Unlike classical joint-space CMP formulations, **DQ-CMP** encodes task dynamics directly as phase-synchronized *end-effector wrench primitives* within a unified $SE(3)$ representation. Our method leverages the dual-quaternion DMP for compact 6-DoF pose encoding while replacing joint torque primitives with operational-space wrench primitives. These learned interaction behaviors are expressed directly at the end-effector. Furthermore, we integrate a Gaussian-process regression model [11] defined on the pose manifold to generalize learned wrenches across varying initial configurations.

The main contributions of this paper are:

- We extend dual-quaternion Dynamic Movement Primitives (DQ-DMP) with a force-learning component to form **DQ-CMP**, enabling end-effector compliance

through learned interaction wrenches.
- Learning and reproduction are performed entirely in operational space, which simplifies data collection, reduces robot-specific dependencies, and improves cross-platform transferability.
- We introduce wrench primitives and leverage a Gaussian process regression scheme adapted to the $SE(3)$ manifold to generalize learned wrenches to novel initial poses.
- We formulate CMPs using dual-quaternion algebra to obtain a coupled, compact, and singularity-free representation of translation and rotation, well suited for contact-rich deformable-object manipulation.
- We validate the approach on real-robot experiments involving deformable-object tasks, demonstrating compliant execution, synchronized motion–force reproduction, and generalization to new starting poses.

The remainder of the paper is organized as follows. Section II surveys related work on motion and force learning for deformable-object manipulation. Section III introduces mathematical preliminaries required to understand the proposed approach. Section IV presents the entire approach including the learning algorithm, lower level controller, and the $SE(3)$-aware Gaussian process used for wrench generalization. Section V reports experimental evaluations on real-robot tasks and analyzes performance. Finally, Section VI concludes the paper and outlines directions for future work.

II. Related Work

LfD or Imitation learning is a popular method to teach robots to do complex task where there are multiple constraints and issues to consider. This work sits at the intersection of three related research threads. (i) Motion based methods most notably Dynamic Movement Primitives (DMPs). (ii) Compliant motion and force encoding, commonly addressed with Compliant Movement Primitives (CMPs). (iii) Statistical generalization of learned interaction parameters (forces/wrenches). Below we review representative and closely related works in each area, and highlight the limitations that motivate an operational-space, dual-quaternion CMP with manifold-aware generalization.

A. Dynamic Movement Primitives (DMPs)

DMPs provide a widely used framework for encoding demonstrated trajectories as globally stable dynamical systems. It is modulated by a learned nonlinear forcing term, typically approximated with weighted Gaussian basis functions [12]. Their desirable properties are simplicity (often requiring only a single demonstration), global stability, and robust temporal/spatial scaling. This has made them a standard choice for motion encoding in LfD [4].

Early operational-space DMP implementations represented translation and orientation separately (e.g., three position and four for orientation in quaternion space as separate transformation systems) [13]. Such decompositions create practical issues. (i) Using four transformation systems for orientation would require post-normalization step to maintain unit-norm constraint. (ii) Separate treatment can decouple rotation and translation in ways that are undesirable for coupled 6-DoF tasks. Quaternion-based variants addressed the unit-norm constraint and noise filtering for orientation trajectories [14], [15], but still treat rotation and translation as distinct channels.

To obtain a single, coupled, singularity-free pose representation, unit dual quaternions have been proposed to represent 6-DoF pose trajectories within a single DMP transformation system [5]. Dual-quaternion based DMPs (DQ-DMPs) compactly encode translation and rotation as a coupled entity and permit compact representation of twists using dual quaternions [16]. DQ-DMPs thereby avoids normalization issues (by using unit dual quaternions) which occurs while considering separate transformation systems for orientation and preserve the geometric coupling between orientation and position. This leads to significant advantages for complex 6-DoF manipulation tasks. However, the existing DQ-DMP work focuses on motion reproduction and velocity shaping and does not incorporate interaction forces or compliant behaviors required for contact-rich deformable-object manipulation.

B. Compliant Movement Primitives (CMPs)

Compliant Movement Primitives augment trajectory primitives (DMPs) with learned force profiles (originally encoded as torque primitives in joint space) to reproduce both motion and task dynamics for compliant execution [8], [7]. CMPs have been successfully applied to contact-rich periodic tasks (e.g., turning a crank) and other scenarios where joint-level impedance profiles are essential for task success.

Most classical CMP formulations learn *joint-space* torque primitives (TPs) alongside DMP weights, and combine them to produce compliant behavior during execution. While effective on a given robot, joint-space CMPs require access to joint-torque signals and are tightly coupled to the robot kinematics and dynamics. This complicates data collection, limits portability across platforms, and makes task transfer difficult [4]. Several works have investigated statistical generalization of CMP parameters to avoid re-learning for each trajectory variant [9], [10]. However, the joint-space formulation of many CMPs couples the learned torque primitives to the robot's specific kinematics and dynamics, which may reduce portability across different platforms compared to operational-space formulations. This motivates CMP formulations that operate directly on end-effector wrenches rather than joint torques, enabling simpler data collection and easier transfer between robots.

C. Wrench Generalization

To make CMPs practical across task variations, many works have explored mapping task descriptors to CMP parameters using statistical models. Gaussian process regression (GPR) has been used to predict CMP parameters (both motion and torque components) from task descriptors such as object mass or goal height. This provides principled uncertainty estimates in Euclidean task-descriptor spaces [9], [17].

However, when the task descriptor is an end-effector pose, the underlying space is the non-Euclidean manifold $SE(3)$ (rigid motions), and naïve vectorizations (e.g., Euler angles or angle-axis pseudo-vectors) can produce large errors and poor generalization for samples with substantial rotational differences [18].

Recent work has therefore focused on defining valid kernels and GP models on pose manifolds. Valid Gaussian kernels on $SE(3)$ based on angle-axis representations [18] and on homogeneous transformation matrix embeddings have been proposed [19]. It shows improved accuracy for pose-conditioned prediction tasks while preserving kernel positive-definiteness properties needed by GPR [18], [19]. Dual-quaternion-based kernels have also been developed and applied for human motion estimation and visual tracking. It provides accurate mean and uncertainty estimates for pose regression problems by leveraging the compact dual-quaternion pose representation [11]. Such manifold-aware GP models enable principled generalization of wrench or torque primitives to new initial poses and task variations.

III. Preliminaries

Object	Notation
Scalar numbers	$a, b, c \in \mathbb{R}$
Quaternions and Vectors	Bold italics: $\boldsymbol{q}, \boldsymbol{q}_r \in \mathbb{H}$
Dual Quaternions and Dual Vectors	Hat, bold italics: $\hat{\boldsymbol{q}}, \hat{\boldsymbol{q}}_r \in \mathbb{H}_D$

TABLE I: Mathematical Notations

In this section, we provide the necessary mathematical background regarding dual quaternions and screw based kinematics, aimed towards understanding Sec. IV.

According to Chasles's theorem, rigid motion between two frames can be represented as screw displacement [20]. Screw displacement consists of a rotational angle $\theta \in \mathbb{R}^+$ about an axis $\boldsymbol{\ell} \in \mathbb{R}^3$, displacement $d \in \mathbb{R}$ along the axis, and a moment vector $\boldsymbol{m} \in \mathbb{R}^3$ satisfying $\boldsymbol{m} \perp \boldsymbol{\ell}$. Screw displacement $\hat{\boldsymbol{\vartheta}} \in \mathbb{R}^6$, can be represented as a dual vector and thereby endowing it with the algebraic structure of dual numbers

$$\hat{\boldsymbol{\vartheta}} = \theta\boldsymbol{\ell} + \varepsilon(d\boldsymbol{\ell} + \theta\boldsymbol{m}), \tag{1}$$

where ε is the dual unit with $\varepsilon^2 = 0$.

Rigid motion between two frames can also be described by using a pure quaternion $\boldsymbol{q}_{tra}$ representing translation and a unit quaternion $\boldsymbol{q}_{rot}$ representing orientation. These two quaternions can be combined together using the dual unit to form a compact and efficient representation denoted as unit dual quaternion $\hat{\boldsymbol{x}} \in \mathbb{R}^8$ [16]

$$\hat{\boldsymbol{x}} = \boldsymbol{q}_{rot} + \varepsilon\frac{1}{2}\boldsymbol{q}_{tra}\boldsymbol{q}_{rot} = \boldsymbol{q}_r + \varepsilon\boldsymbol{q}_d, \tag{2}$$

where the primary part of the dual quaternion is $\mathcal{P}(.) = \boldsymbol{q}_r = \boldsymbol{q}_{rot}$ and the dual part is $\mathcal{D}(.) = \boldsymbol{q}_d = \frac{1}{2}\boldsymbol{q}_{tra}\boldsymbol{q}_{rot}$. Given a unit dual quaternion, the translational quaternion can be extracted using $\boldsymbol{q}_{tra} = 2\mathcal{D}(\hat{\boldsymbol{x}})\mathcal{P}(\hat{\boldsymbol{x}})^*$. Unit dual quaternion can be obtained from screw displacement using the exponential map [21]

$$\hat{\boldsymbol{x}} = e^{\hat{\boldsymbol{\vartheta}}/2} = \cos\left(\frac{\theta + \varepsilon d}{2}\right) + (\boldsymbol{\ell} + \varepsilon\boldsymbol{m})\sin\left(\frac{\theta + \varepsilon d}{2}\right). \tag{3}$$

The conjugate of quaternion represents inverse transformation

$$\hat{\boldsymbol{x}}^* = \boldsymbol{q_r}^* + \varepsilon\boldsymbol{q_d}^*. \tag{4}$$

Given two unit dual quaternions $\hat{\boldsymbol{x}}_1$ and $\hat{\boldsymbol{x}}_2$, then the screw displacement between them can be computed using the logarithmic mapping of their spacial difference [22]

$$\hat{\boldsymbol{\vartheta}}_e = 2\ln\big(\hat{\boldsymbol{x}}_{(2-1)}\big), \tag{5}$$

where $\hat{\boldsymbol{x}}_{(2-1)} = \hat{\boldsymbol{x}}_1^*\hat{\boldsymbol{x}}_2$.

Given two quaternions $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ representing rotations we can define a distance measurement as [11]

$$d_{\text{arc}}(\boldsymbol{q}_1, \boldsymbol{q}_2) = \min|\arccos(\pm\,\text{Re}(\boldsymbol{q}_2\boldsymbol{q}_1^*))|, \tag{6}$$

where $\text{Re}(.)$ indicates the real part of the quaternion.

IV. Dual Quaternion based Compliant Movement Primitives

In this section, we describe the entire pipeline proposed in this work to execute tasks involving manipulation of deformable objects. The general block diagram of the methodology followed is shown in Fig. 1. It consists of five parts. (i) We collect the demonstration trajectory that results in the completion of the task. (ii) We use DQ-DMP proposed in [5] which learns the desired trajectory in operational space from a single demonstration. (iii) We learn the task dynamics involved by operating the robot autonomously to follow the learned trajectory in high stiffness. We use WPs to encode these interaction forces. (iv) We generalize the WPs to different initial poses. (v) We track the trajectory in compliance (low stiffness) by using the predicted WPs, and we use a Cartesian impedance controller which use the pose error derived from screw displacement.

A. Collecting Pose trajectory

We collect the pose trajectory using kinesthetic guidance where we grasp the end effector at an appropriate part of the deformable object. Then we move the robot to perform the desired task. The trajectory followed by the end-effector is recorded from the internal controller of the robot.

B. Pose Learning via Dual Quaternion based Dynamic Movement Primitives

We first provide the formulation for 6-DoF DQ-DMP system introduced in [5] and a brief overview for the sake of completeness

$$\tau\dot{\hat{\boldsymbol{\omega}}} = K\hat{\boldsymbol{\vartheta}} - D\hat{\boldsymbol{\omega}} - K\hat{\boldsymbol{\vartheta}}_0 s + K\boldsymbol{f_d}(s) \tag{7}$$

$$\tau\dot{\hat{\boldsymbol{\vartheta}}} = \hat{\boldsymbol{\omega}}, \tag{8}$$

where $\hat{\boldsymbol{\vartheta}} = 2\ln(\hat{\boldsymbol{x}}^*\,\hat{\boldsymbol{x}}_g)$, $\hat{\boldsymbol{\vartheta}}_0 = 2\ln(\hat{\boldsymbol{x}}_0^*\,\hat{\boldsymbol{x}}_g)$ are the screw displacements from current pose and initial pose respectively

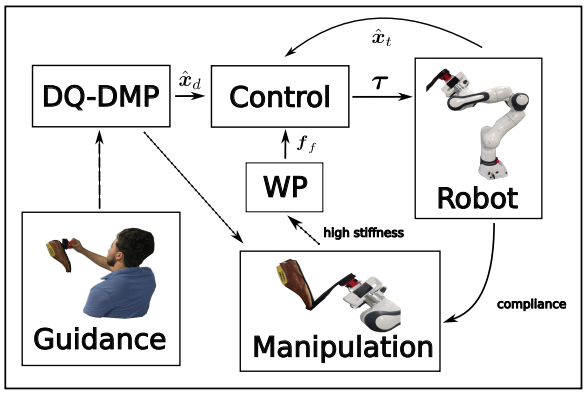


Fig. 1: Block diagram representing the entire pipeline considered. The dashed arrow represent encoding of non-linear function in DQ-DMP using kinesthetic guidance. The dotted arrow represent encoding of WP by autonomously making robot follow the learned trajectory from DQ-DMP in high stiffness. The solid arrows indicate the final control architecture used for task execution where manipulation is achieved in compliance or low stiffness.

to the goal pose. $\{\hat{\boldsymbol{x}},\ \hat{\boldsymbol{x}}_0,\ \hat{\boldsymbol{x}}_g\}$ are the current, initial and goal pose respectively. $\{\hat{\boldsymbol{\omega}},\ \hat{\dot{\boldsymbol{\omega}}}\}$ are twist velocity and acceleration. K, D are scalar stiffness and damping parameters of the DMP system, τ is the scalar temporal scaling factor and $s \in \mathbb{R}$ is the canonical phase variable that is exponentially regulated using time constant α_s

$$\dot{s} = -\alpha_s s. \tag{9}$$

$\boldsymbol{f_d}(s) \in \mathbb{R}^6$ is the nonlinear forcing term defined as

$$\boldsymbol{f_d}(s) = \frac{\sum_{i=1}^{N} \psi_i(s)\, \boldsymbol{w}_d\, s}{\sum_{i=1}^{N} \psi_i(s)},\ \ \psi_i(s) = \exp\left(-h_i(s-c_i)^2\right), \tag{10}$$

where $\{\psi_i(s)\}$ are Gaussian basis functions with width h_i and center c_i and $\boldsymbol{w}_d \in \mathbb{R}^6$ their learned weights.

For pose learning, given a recorded demonstration, we first filtered it by using DQ-DMP without the forcing term in (7) similar to [5]. Then we follow the standard approach [4] to obtain the non-linear forcing term associated with the recorded trajectory by inverting (7) while using the recorded demonstration in DQ-DMP. From this forcing term, we obtain the weights $\boldsymbol{w}_d$. These weights are then used to reproduce the forcing term using (10) that ensures that the final trajectory resembles the shape of the demonstration. DQ-DMP system (7, 8) is then solved while accommodating different initial poses of the end-effector to obtain the learned trajectory.

C. Force Learning via Wrench Primitives

In addition to learning the end-effector trajectory, we also encode the end-effector wrench experienced for task completion as Wrench Primitives. These are essential for enabling compliant behavior during reproduction, particularly in contact-rich tasks. Let $\boldsymbol{f}_f(s) \in \mathbb{R}^6$ denote the wrench applied by end-effector during the manipulation of deformable object. To reproduce this interaction, we encode the force $\boldsymbol{f}_f(s)$ similar to that of nonlinear forcing term $\boldsymbol{f_d}(s)$. The task force involved is represented using a weighted sum of radial basis functions (RBFs) as

$$\boldsymbol{f}_f(s) = \frac{\sum_{i=1}^{N} \psi_i(s)\boldsymbol{w}_f}{\sum_{i=1}^{N} \psi_i(s)}, \tag{11}$$

where $\boldsymbol{w}_f \in \mathbb{R}^6$ are the learned weights, $\psi_i(s)$ are phase-dependent Gaussian basis functions and s is the shared phase variable as defined in the DQ-DMP formulation. This formulation allows for time-consistent learning of force profiles aligned with the motion trajectory. By using a common phase variable s, the poses and forces are temporally synchronized, ensuring coherent motion-force coordination.

For force learning, we first record the end-effector wrench while tracking the learned trajectory. To achieve accurate tracking, we operate the lower level controller given in (18) at high stiffness (k_w) without using the feed-forward term. The high-stiffness regime is used only during a short and controlled data-collection phase, to ensure accurate trajectory tracking while logging interaction wrenches. These interaction forces are then encoded as learned weights associated with the RBFs as seen in (11). The resulting learned wrench is then used as a feed-forward term in the control law, improving task execution in contact scenarios.

D. Generalizing Wrench Primitives

It is important to generalize the WPs to new conditions such as different initial poses. The task completion would demand a learned trajectory from new initial poses and the forces required need to be adapted to this new scenario. We use the statistical generalization method considered in [9]. We perform multiple demonstrations to measure interaction forces from different initial poses and encode them as learned weights $\boldsymbol{w}_f$ as described in the previous section. The demonstration number j is associated with a task descriptor, i.e., initial starting pose $\hat{\boldsymbol{x}}_0^{(j)}$ and learned weight $\boldsymbol{w}_f^j$ with each dimension as $w_{f_i}^{(j)}$. We used the GPR [17] to predict the weights needed for a new learned trajectory without requiring a new demonstration. We describe below the three parts involved in force learning.

1) Gaussian Process Model: Each scalar weight $w_{f_i}^{(j)}$ is modelled as $\mathcal{GP}$ adapted to $SE(3)$ manifold using dual quaternions [11]

$$w_{f_i}^{(j)}(\hat{\boldsymbol{x}}_0) \ \sim\ \mathcal{GP}_{\mathbb{H}_D}\left(0,\, k_{\mathbb{H}_D}(\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{x}}_0')\right), \tag{12}$$

where $\hat{\boldsymbol{x}}_0$ is the starting pose of the end-effector and $\hat{\boldsymbol{x}}_0'$ is a pose of the training set. The kernel $k_{\mathbb{H}_D}(\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{x}}_0')$ represents the covariance that encodes the similarity between poses, and we use a valid kernel for $SE(3)$ manifold [11]

$$k_{\mathbb{H}_D}(\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{x}}_0') = \sigma_f^2 \exp\left(-\frac{\left[d_{\mathbb{H}_D}\left(\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{x}}_0'\right)\right]^2}{2l_f^2}\right), \tag{13}$$

where $d_{\mathbb{H}_D}\left(\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{x}}_0'\right)$ is the associated dual quaternion distance measurement, $\ell_f > 0$ is the length scale, and $\sigma_f^2 > 0$

is the signal variance. The dual quaternion distance measurement is defined as

$$d_{\mathbb{H}_D}\left(\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{x}}_0'\right) = \sqrt{\left[d_{\text{arc}}\left(\mathbf{q}_0, \mathbf{q}_{\text{t}_{(0'-0)}}\right)\right]^2 + \left\|\mathbf{q}_{\text{tra}_{(0'-0)}}\right\|^2}, \tag{14}$$

where $\mathbf{q}_0$ denotes zero rotation, $d_{\text{arc}}(.,.)$ is the arc metric defined between two unit quaternions,

2) Prediction: Given a new start pose $\hat{\boldsymbol{x}}_{0_{\text{new}}} \in \mathbb{R}^6$, the predictive mean weight is

$$w_{f_i}^{\text{pred}} = k_{\mathbb{H}_D}(\hat{\boldsymbol{x}}_0, \hat{\boldsymbol{x}}_0')\left[\mathbf{K}_{\mathbb{H}_D} + \sigma_n^2 \mathbf{I}_{6\times 6}\right]^{-1} \mathbf{y}_i, \tag{15}$$

where $\mathbf{K}_{\mathbb{H}_\mathbf{D}} \in \mathbb{R}^{N_g \times N_g}$ is the kernel Gram matrix, N_g is the number of demonstration used on training, $\left\{\boldsymbol{x}_0^{(1)}, \ldots, \boldsymbol{x}_0^{(N_g)}\right\}$ are the training descriptors, $\boldsymbol{y}_i = \left(w_{f_i}^{(1)}, \ldots, w_{f_i}^{(N_g)}\right)^T \in \mathbb{R}^{N_g}$ are the corresponding training weights, and $\sigma_n^2 > 0$ is the variance of i.i.d Gaussian noise considered. The Gram matrix consists of kernels between each sample in the training set

$$\mathbf{K}_{\mathbb{H}_D} = \begin{bmatrix} k_{\mathbb{H}_D}(\hat{\boldsymbol{x}}_0^{(1)}, \hat{\boldsymbol{x}}_0^{(1)}) & \cdots & k_{\mathbb{H}_D}(\hat{\boldsymbol{x}}_0^{(1)}, \hat{\boldsymbol{x}}_0^{(N_g)}) \\ \vdots & \ddots & \vdots \\ k_{\mathbb{H}_D}(\hat{\boldsymbol{x}}_0^{(N_g)}, \hat{\boldsymbol{x}}_0^{(1)}) & \cdots & k_{\mathbb{H}_D}(\hat{\boldsymbol{x}}_0^{(N_g)}, \hat{\boldsymbol{x}}_0^{(N_g)}) \end{bmatrix}. \tag{16}$$

3) Force Reconstruction: The prediction step is repeated for each dimension to obtain $\boldsymbol{w}_f^{\text{pred}}$ that enables to obtain the predicted force as

$$\boldsymbol{f_f}^{\text{pred}}(s) = \frac{\sum_{i=1}^{N} \psi_i(s)\, \boldsymbol{w}_f^{\text{pred}}}{\sum_{i=1}^{N} \psi_i(s)}. \tag{17}$$

E. Autonomous Execution of Task

The final step in the methodology is the autonomous operation of the robot to complete the task in compliance. A modified version of the standard Cartesian impedance controller is used as lower level controller. It ensures the tracking of the learned pose trajectory at low stiffness by accommodating the predicted force $\boldsymbol{f}_{\boldsymbol{f}}^{\mathbf{pred}}(s)$ to achieve compliance.

$$\boldsymbol{\tau} = \boldsymbol{J}^T(\boldsymbol{q})\left(-\mathbf{K}_\omega \boldsymbol{\xi} - \mathbf{D}_\omega \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{f}_{\boldsymbol{f}}^{\mathbf{pred}}(s)\right), \tag{18}$$

where $\boldsymbol{\xi}$ is the pose error, $\mathbf{K}_w = k_w \mathbf{I}_{6\times 6}$ is the stiffness matrix defined using scalar k_w, $\mathbf{D}_\omega$ is the damping matrix, $\boldsymbol{J}(\boldsymbol{q})$ is geometric Jacobian, and $\boldsymbol{q}$ is the joint vector. The pose error is defined using dual quaternions and screw displacements as

$$\boldsymbol{\xi} = \text{vec}_6(\hat{\boldsymbol{\vartheta}}_e) = \begin{pmatrix} d_e \boldsymbol{\ell}_e + \theta_e \boldsymbol{m}_e \\ \theta_e \boldsymbol{\ell}_e \end{pmatrix}, \tag{19}$$

where $\hat{\boldsymbol{\vartheta}}_e = 2\ln(\hat{\boldsymbol{x}}_t^* \hat{\boldsymbol{x}}_d) = \theta_e \boldsymbol{\ell}_e + \boldsymbol{\varepsilon}(d_e \boldsymbol{\ell}_e + \theta \boldsymbol{m}_e)$ and $\text{vec}_6(\hat{\boldsymbol{\vartheta}}_e) = \left(\mathcal{D}(\hat{\boldsymbol{\vartheta}}_e)^T,\ \mathcal{P}(\hat{\boldsymbol{\vartheta}}_e)^T\right)^T$. Such a pose error have been used in the past for kinematic controller in [23], [22], for admittance controller in [24] and for resolved acceleration controller in [25]. In the next section we will describe the experimental results obtained.

V. Experimental Results

We present a comprehensive evaluation of our DQ-CMP method. We start by presenting the experiment setup. Then, we compare the DQ-CMP against DQ-DMP for two distinct use cases. (i) *Shoe sole detachment* task where a shoe sole needs to be removed from the rest of the shoe using a single robot. (ii) *Foam-in-box* task which denotes an industrial task of bending a flexible object with high stiffness inside a box using two robots. Finally, we evaluate DQ-CMP performance for WPs generalization.

(a) FR3 grasping the shoe sole attached to a rigid structure.

(b) Operator collecting data by kinesthetic guidance.

Fig. 2: Experimental setup for performing shoe sole detachment task using a Franka Research 3 (FR3) robot

(a) Bi-manual manipulation of high stiffness foam.

(b) Operator collecting data using kinesthetic guidance.

Fig. 3: Experimental setup for foam bending task using two Franka Research 3 (FR3) robot.

A. Experimental Setup

For the first use case, *shoe sole detachment*, we have an experimental setup consisting of a static platform to which we fix one part of the shoe, rigidified using a shoe last (Fig. 2a). We attach the shoe sole using Velcro. On the other side of the shoe sole, we attach a gripper aid that helps the gripper of FR3 to firmly grasp the sole. In Fig. 2b, we show how an operator performs the task to collect the demonstrated trajectory.

For the second use case, *foam-in-box*, we use two FR3 robots with default grippers (Fig. 3a). The object needs to be bent in a U-shape to be inserted in the box that is placed in between the two robots. We obtain the demonstration trajectory through kinesthetic teaching by an operator as shown in Fig. 3b.

We use the estimated wrenches provided by the manufacturer of FR3, expressed in the base frame for both learning and execution. FR3 robots have torque sensors integrated at each joint, which accurately estimate external contact forces. For both tasks, we recorded the demonstration

trajectory at 30 Hz. We used $N = 100$ Gaussian basis functions with a DQ-DMP stiffness gain of $K = 25$ and a damping parameter of $D = \sqrt{4K\,\tau}$ with $\tau = 1$ to obtain the critical damping behavior. For generalizing WPs, we obtained training samples from 6 different initial poses. The laptop used for experimentation validation has an Intel(R) Core(TM) i7-7820HQ CPU with 8 cores running at 2.90GHz.

B. Comparing DQ-DMP with DQ-CMP

This section compares baseline DQ-DMP (trajectory reproduction only) with DQ-CMP (trajectory + wrench primitives) in both tasks. We evaluate (i) trajectory tracking fidelity (pose RMSE with respect to the learned trajectory) and (ii) successful task completion as a function of controller stiffness.

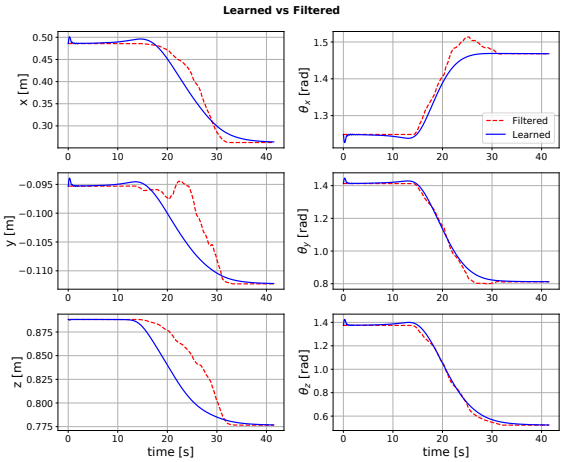


Fig. 4: Comparison between filtered demonstration trajectory (dashed red line) with the learned trajectory from DQ-DMP (solid blue line) for *shoe sole detachment* task, represented in base frame.

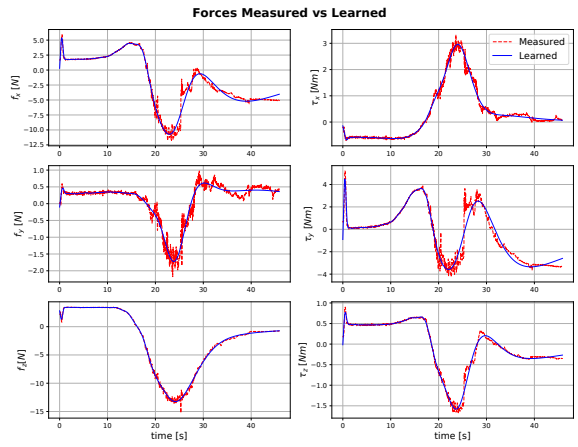


Fig. 5: Comparison between measured end-effector wrench (red dashed line) and the associated WPs (blue solid lines) obtained from executing the task at high stiffness from the same initial pose as the demonstrated trajectory for *shoe sole detachment* task, represented in base frame.

Figure 4 shows the filtered demonstration and the learned DQ-DMP trajectory for the shoe-detachment task from the demonstrated start pose. DQ-DMP reproduces the demonstrated pose sequence and converges to the demonstrated goal. DQ-CMP uses the same learned pose trajectory along with learned wrench primitives. Figure 5 shows these learned wrench primitives (WPs) and the measured end-effector wrench when executing at high stiffness. The WPs closely match the measured interaction wrench.

To evaluate robustness to controller compliance, we executed the same learned motion with five different values of stiffness (k_w) used in lower level controller (18), i.e., $k_w = \{50,\ 100,\ 200,\ 300,\ 400\}$. Fig. 6 plots RMSE for pose error ($\boldsymbol{\xi}$) evaluated from screw displacement in (19) across stiffness values, comparing DQ-DMP against DQ-CMP. At high stiffness ($k_w = 400$), both DQ-DMP and DQ-CMP track the learned trajectory well and achieve similar low RMSE values. As stiffness decreases, DQ-DMP tracking error increases substantially and task execution fails at the lowest stiffness ($k_w = 50$) where the shoe sole does not detach. DQ-CMP maintains low RMSE across stiffness values by adding the learned WP feedforward term. Even at low stiffness, the predicted interaction wrench enables successful detachment and produces RMSE comparable to the high-stiffness executions.

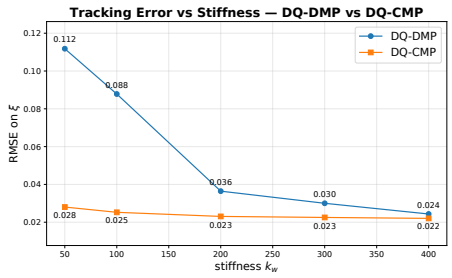


Fig. 6: Comparison between task execution using DQ-DMP and DQ-CMP by evaluating the deviation of executed trajectory from the learned trajectory for *shoe sole detachment* task. We compare RMSE values based on pose error ($\boldsymbol{\xi}$) for the task execution at various values of stiffness k_w.

We observed the same qualitative effect in the bi-manual *foam-in-box* task. Fig. 7 shows reproduction of the learned bimanual trajectories by the DQ-CMP. Fig. 8 displays measured and learned wrenches for both arms at high stiffness of $k_w = 700$. The learned WPs capture the required interaction behavior for bending; during execution the wrench magnitude increases toward task completion, consistent with the required bending forces.

Overall, these experiments demonstrate that learning and applying end-effector wrenches in operational space (DQ-CMP) enables compliant, task-successful execution even under low controller stiffness, whereas trajectory-only DQ-DMP cannot guarantee task success when the controller is compliant.

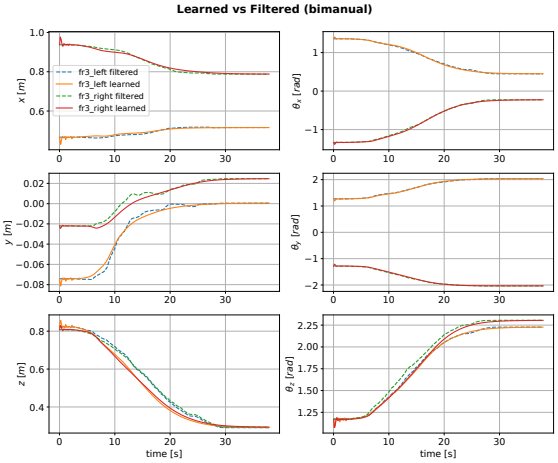
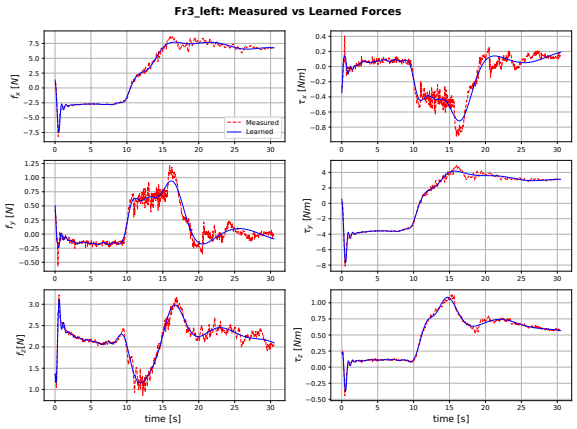
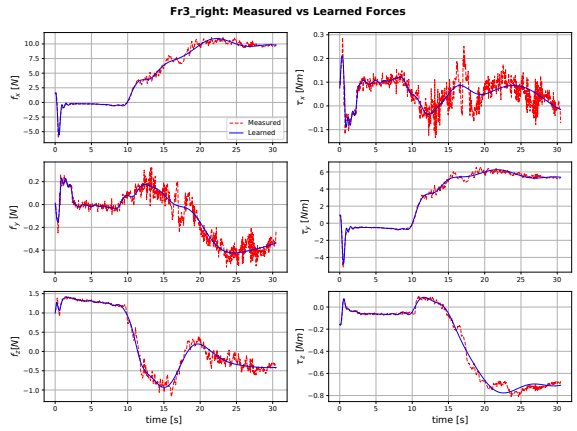


Fig. 7: Comparison between the filtered demonstration trajectory (dashed lines) and the trajectory learned by DQ-DMP (solid lines) for *foam-in-a-box* task using the two arms of FR3, represented in base frame of left arm.



(a) Force learning (left arm)



(b) Force learning (right arm)

Fig. 8: Comparison between measured end-effector wrench (red dashed line) and forces learned using Wrench Primitives (blue solid lines) obtained for bimanual *foam-in-a-box* task, represented in individual base frames.

C. Evaluation of Generalization of WPs

Here we evaluate whether WPs learned from a small set of demonstrations generalize to new, unseen initial poses. For the shoe-detachment task we trained a GPR model defined on the pose manifold (dual-quaternion kernel) using 6 training initial poses. The GPR maps the initial end-effector pose to the WP weights required for the demonstrated task. During testing, we selected a novel initial pose not seen during training and executed the DQ-CMP at low stiffness $k_w = 50$, using the GPR-predicted WP.

Fig. 9 shows the learned DQ-CMP trajectory from the new initial pose compared to the filtered demonstration. The DQ-CMP trajectory generalizes the pose motion to the new start. Fig. 10 compares the measured end-effector wrench during execution with the WP reconstructed from GPR-predicted weights. The predicted WP closely matches the measured wrench and enables successful detachment at low stiffness. Utilization of this predicted wrench enabled the successful completion of the task at low stiffness.

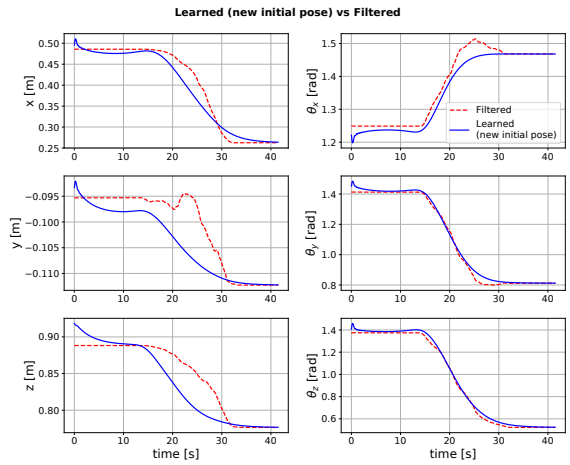


Fig. 9: Comparison between filtered demonstration trajectory (dashed red line) with the learned DQ-DMP trajectory from a different initial pose (solid blue line) for *shoe sole detachment* task, represented in base frame.

These results confirm that, (i) a small number of demonstrations (here 6) suffice to learn a predictive mapping from start pose to required WP using a manifold-aware GPR, and (ii) the predicted WPs enable compliant, safe task execution from unseen initial poses.

Overall, the experimental evidence shows that DQ-CMP reproduces both motion and interaction dynamics in operational space, yielding representation-level robot-independence and singularity-free behavior that is robust to controller compliance. Wrench generalization via pose-manifold GPR makes the approach practical for new starting poses without re-demonstration. Together, these properties make DQ-CMP suitable for contact-rich deformable-object tasks where both motion and force must be carefully coordinated.

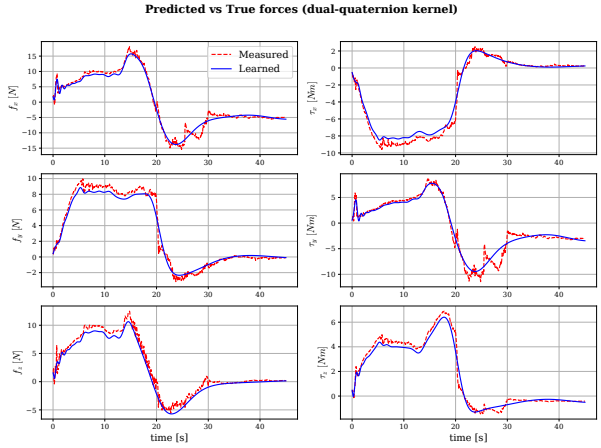


Fig. 10: Comparison between measured end-effector wrench (red dashed line) and the predicted WP (blue solid lines) reconstructed from the predicted weights using GPR from a new initial pose for *shoe sole detachment* task, represented in base frame.

VI. CONCLUSION

In this work, we extend DQ-DMP to accommodate task-specific forces that are difficult to model. We proposed the idea of WPs that captures the interaction forces involved in manipulation of deformable objects. These WPs are generalized using GPR technique to different initial poses of the end-effector. The methodology developed is experimentally validated in two uses cases, one involving the detachment task of shoe sole using a single robot and the other bimanual manipulation of a foam with high stiffness to insert inside a box. We show results to validate both accurate tracking of task-specific trajectories at low stiffness values and the accuracy of the predicted WP. This concludes that the proposed solution is effective in scenarios such as detachment task where the remaining part of the shoe sole must not be compromised while completing the task. The undamaged part of shoe can later be used to re-fabricate a new shoe and thereby boost a circular economy.

As part of the future work, there are three possible directions of improvement. (i) A broader study considering various other tasks with large pose variations and comparing with additional force-aware baselines. (ii) Verifying safety for the environment and the human operator by performing a detailed stability analysis. (iii) Including shape feedback information to achieve the desired shape of the deformable object while completing the task.

REFERENCES

[1] H. Yin, A. Varava, and D. Kragic, "Modeling, learning, perception, and control methods for deformable object manipulation," *Science Robotics*, vol. 6, 2021.

[2] J. Zhu, A. Cherubini, C. Dune, D. Navarro-Alarcon, F. Alambeigi, D. Berenson, F. Ficuciello, K. Harada, J. Kober, X. Li, J. Pan, W. Yuan, and M. Gienger, "Challenges and outlook in robotic manipulation of deformable objects," *IEEE Robotics and Automation Magazine*, vol. 29, no. 3, pp. 67–77, 2022.

[3] J. Sanchez, J. A. Corrales, B. C. Bouzgarrou, and Y. Mezouar, "Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 688–716, 2018.

[4] M. Saveriano, F. J. Abu-Dakka, A. Kramberger, and L. Peternel, "Dynamic movement primitives in robotics: A tutorial survey," *The International Journal of Robotics Research*, vol. 42, no. 13, pp. 1133–1184, 2023.

[5] R. Chandra, V. H. Giraud, M. Alkhatib, and Y. Mezouar, "Dual quaternion based dynamic movement primitives to learn industrial tasks using teleoperation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 3757–3763.

[6] M. Rambow, T. Schauß, M. Buss, and S. Hirche, "Autonomous manipulation of deformable objects based on teleoperated demonstrations," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 2809–2814.

[7] M. Deniša, T. Petrič, A. Gams, and A. Ude, "A review of compliant movement primitives," in *Robot Control*, E. G. Hurtado, Ed. Rijeka: IntechOpen, 2016, ch. 1.

[8] T. Petrič, A. Gams, L. Žlajpah, and A. Ude, "Online learning of task-specific dynamics for periodic tasks," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 1790–1795.

[9] M. Deniša, A. Gams, A. Ude, and T. Petrič, "Learning compliant movement primitives through demonstration and statistical generalization," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 5, pp. 2581–2594, 2016.

[10] T. Petrič, A. Gams, L. Colasanto, A. J. Ijspeert, and A. Ude, "Accelerated sensorimotor learning of compliant movement primitives," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1636–1642, 2018.

[11] M. Lang, M. Kleinsteuber, and S. Hirche, "Gaussian process for 6-dof rigid motions," *Autonomous Robots*, vol. 42, no. 6, pp. 1151–1167, 2018.

[12] S. Schaal, *Dynamic Movement Primitives -A Framework for Motor Control in Humans and Humanoid Robotics*. Tokyo: Springer Tokyo, 2006, pp. 261–280.

[13] 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*, 2009, pp. 763–768.

[14] P. Pastor, L. Righetti, M. Kalakrishnan, and S. Schaal, "Online movement adaptation based on previous sensor experiences," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 365–371.

[15] R. Weitschat, A. Dietrich, and J. Vogel, "Online motion generation for mirroring human arm motion," in *2016 ICRA*, 2016, pp. 4245–4250.

[16] B. V. Adorno, "Robot kinematic modeling and control based on dual quaternion algebra—part i: Fundamentals." 2017.

[17] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*, ser. Adaptative computation and machine learning series. University Press Group Limited, 2006.

[18] M. Lang and S. Hirche, "Computationally efficient rigid-body gaussian process for motion dynamics," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1601–1608, 2017.

[19] M. Omainska, J. Yamauchi, A. Lederer, S. Hirche, and M. Fujita, "Rigid motion gaussian processes with se(3) kernel and application to visual pursuit control," *IEEE Control Systems Letters*, vol. 7, pp. 2665–2670, 2023.

[20] J. M. Selig, "Geometric fundamentals of robotics," in *Monographs in Computer Science*, 2004.

[21] B. Kenwright, "A survey on dual-quaternions," 2023. [Online]. Available: https://arxiv.org/abs/2303.14765

[22] H. Abaunza, R. Chandra, E. Özgür, J. A. C. Ramón, and Y. Mezouar, "Kinematic screws and dual quaternion based motion controllers," *Control Engineering Practice*, vol. 128, p. 105325, 2022.

[23] E. Özgür and Y. Mezouar, "Kinematic modeling and control of a robot arm using unit dual quaternions," *Robotics and Autonomous Systems*, vol. 77, pp. 66–73, 2016.

[24] M. d. P. A. Fonseca, B. V. Adorno, and P. Fraisse, "Coupled task-space admittance controller using dual quaternion logarithmic mapping," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6057–6064, 2020.

[25] R. Chandra, J. A. Corrales-Ramon, and Y. Mezouar, "Resolved-acceleration control of serial robotic manipulators using unit dual quaternions," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 8500–8505, 2020, 21st IFAC World Congress.