



Quick Setup of Force-Controlled Industrial Gluing Tasks Using Learning From Demonstration

Iñigo Iturrate*, Aljaz Kramberger and Christoffer Sloth

SDU Robotics, Maersk McKinney Moller Institute, University of Southern Denmark, Odense, Denmark

OPEN ACCESS

Edited by:

Jacopo Aleotti,
University of Parma, Italy

Reviewed by:

Adam Conkey,
The University of Utah, United States

Anuj Pasricha,
University of Colorado Boulder,
United States

Marco Carnevale,
Università di Pavia, Italy

*Correspondence:

Iñigo Iturrate
inju@mmmi.sdu.dk

Specialty section:

This article was submitted to
Robot and Machine Vision,
a section of the journal
Frontiers in Robotics and AI

Received: 31 August 2021

Accepted: 15 October 2021

Published: 05 November 2021

Citation:

Iturrate I, Kramberger A and Sloth C
(2021) Quick Setup of Force-
Controlled Industrial Gluing Tasks
Using Learning From Demonstration.
Front. Robot. AI 8:767878.
doi: 10.3389/frobt.2021.767878

This paper presents a framework for programming in-contact tasks using learning by demonstration. The framework is demonstrated on an industrial gluing task, showing that a high quality robot behavior can be programmed using a single demonstration. A unified controller structure is proposed for the demonstration and execution of in-contact tasks that eases the transition from admittance controller for demonstration to parallel force/position control for the execution. The proposed controller is adapted according to the geometry of the task constraints, which is estimated online during the demonstration. In addition, the controller gains are adapted to the human behavior during demonstration to improve the quality of the demonstration. The considered gluing task requires the robot to alternate between free motion and in-contact motion; hence, an approach for minimizing contact forces during the switching between the two situations is presented. We evaluate our proposed system in a series of experiments, where we show that we are able to estimate the geometry of a curved surface, that our adaptive controller for demonstration allows users to achieve higher accuracy in a shorter demonstration duration when compared to an off-the-shelf controller for teaching implemented on a collaborative robot, and that our execution controller is able to reduce impact forces and apply a constant process force while adapting to the surface geometry.

Keywords: learning from demonstration, parameter estimation, force control, gluing, adaptive control

1 INTRODUCTION

Learning from demonstration enables programming of a variety of robotic tasks in a way that is intuitive to a non-expert end-user. Previously, this method was utilized in an industrial context to teach robot based assembly tasks e.g., peg-in-hole insertion task, as well as polishing and grinding tasks. However, most existing approaches are applied on artificial or simplified benchmarks, and often do not consider the tight tolerances and process requirements with regards to the applied force that occur in a real manufacturing scenario. They often require multiple task demonstrations in order to learn task constraints or achieve robust task executions. Lastly, they often do not encompass the full pipeline of demonstration—encoding—execution from an end-user point of view.

In this paper we present a method for fast programming of a force-controlled industrial gluing task from a single user demonstration. We propose the use of adaptive admittance control during the demonstration phase to enable the user to kinesthetically move the robot more intuitively by providing a better control response, allowing them to provide more accurate task demonstrations.

From a user demonstration of the task, we obtain information on the kinematic trajectory of the robot. This allows us to then encode the motion in a motion primitive that allows both direct replay

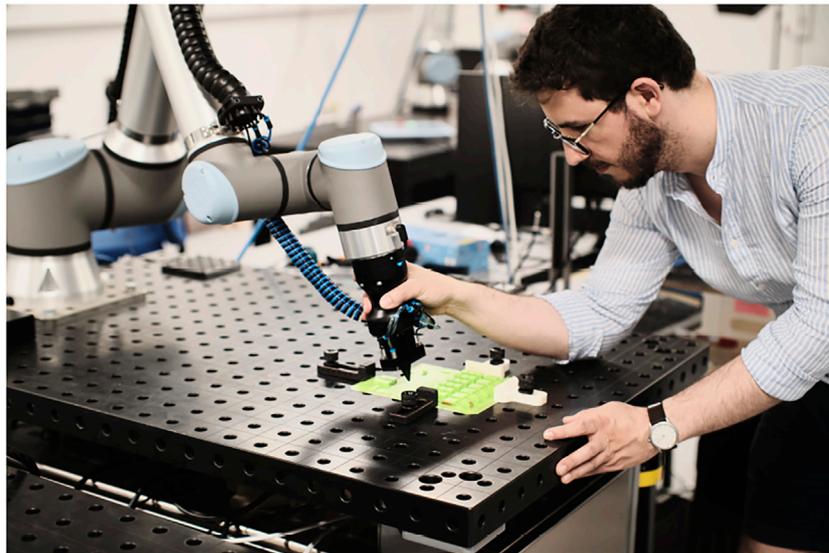


FIGURE 1 | User demonstration of a gluing-like task on our benchmark system.

and generalization of the task. By detecting the contact state of the robot and estimating the surface normal, we can segment the task into phases of unconstrained (free-air) and constrained (in-contact) movement. Simultaneously, we estimate the constrained axes of the robot along the trajectory. This allows us to formulate a parallel position/force controller for the subsequent execution of the task, such that we can either imitate the forces applied during the demonstration or allow the user to specify a constant contact force, as is typical in industrial applications.

We evaluate our system on a benchmark that is closely based on a real industrial gluing task related to printed circuit board assembly, as shown in **Figure 1**, showing that our method can ensure constant and safe interaction forces and suitable trajectories in a task with low contact-force requirements.

2 RELATED WORK

With the evolution of collaborative robots, new user-friendly efficient and flexible ways for automating industrial tasks which were previously done with specialized automation equipment are possible (Gašpar et al., 2020). Furthermore, methods such as Learning by demonstration (LbD) (Billard et al., 2008) and force control can be exploited, thanks to the integrated sensors and controllers of the collaborative robot systems.

In robotic learning by demonstration scenarios, robot trajectories that describe the kinematic information are recorded (Deniša and Ude, 2015). On the other hand, for tasks associated with industrial robotics e.g., assembly (Abu-Dakka et al., 2015) or polishing (Gams et al., 2013), dynamic data is equally important. These two data sets lay the basis for in-contact execution of robot tasks, where the learned trajectory cannot be replayed but has to be adaptable to changes in the

environment. To facilitate this requirement, in a lot of works, robot trajectories are represented with Dynamic Movement Primitives (Ijspeert et al., 2013; Ude et al., 2014). With this framework, both kinematic and dynamic trajectories can be represented in a unified manner, and can be enhanced with additional properties such as modulation and time scaling.

The benefits of the DMP framework can be efficiently exploited for in-contact tasks, when coupling the DMP in a force-based LbD scenario, as shown in Koropouli et al. (2012) and Kormushev et al. (2011). In the work of Rozo et al. (2013) a two stage force LbD approach was presented, where in the first stage they recorded the positions and orientations of the desired movement and in the subsequent stage the corresponding forces and torques.

Simultaneous control of force and position for teaching and execution tasks is physically impossible (Stramigioli, 2001), and therefore these approaches all have to make compromises between prioritising force control or position control. In literature, classical position/force controllers are still adopted for dealing with this challenge. One of the early works in this field was presented by Asada and Izumi (1989). The measured data was used to automatically program the position and force set points for the hybrid force control framework (Raibert and Craig, 1981) used in Cartesian space. Furthermore, simultaneous position and force control frameworks in the LbD literature primarily focus on selecting the dimension of the constraint frame on which the force or position control is applied (Pernel et al., 2017) and the selection of the best control frame (Ureche et al., 2015).

The main motivation for constraint selection methods is that the dimension of the constraint frame that is exposed to a high variance in force and low variance in position should favor force control, and vice versa (Ureche et al., 2015). Constraint frames are usually chosen manually based on the task specifications Raibert

and Craig (1981). Common choices include the tool frame (Kronander and Billard, 2014; Peternel et al., 2017) and the surface normals (Deng et al., 2016; Conkey and Hermans, 2019). The latter can also be estimated, e.g. from the velocity of the contact point (Karayiannidis et al., 2014; Sloth and Iturrate, 2021). Conkey and Hermans (2019) present a framework for learning a dynamic constraint frame aligned to the direction of desired force using CDMPs and force control to ensure that the force is aligned in the normal plane with the direction of the movement. In (Kober et al., 2015), multiple demonstrations of a task are used to segment a series of separate movement primitives, each associated with a specific discrete frame of reference. The transitions between each of the primitives—and therefore compliant frames—are then separately addressed. Kumar and Rani (2021) present a new hybrid force/position control approach for time-varying constrained re-configurable manipulators for physical human robot interaction with environment of unknown stiffness. A review of hybrid position/force controllers is presented in Ortenzi et al. (2017).

An alternative to hybrid force/position control is impedance control with varying parameters. By changing the stiffness parameters of the impedance controller, the interaction forces can be adjusted between the robot and the environment (Buchli et al., 2011). Stiffness adaptation can be learned from demonstration in multiple ways, and typically requires multiple demonstrations or iterative corrections of the path. In Lee et al. (2015), the mean and covariance information of the trajectories from multiple demonstrations of a task are used as an approximation of the desired stiffness along different regions of the path. Kronander and Billard (2012, 2014) allow the user to iteratively re-execute the task and apply external disturbances on the robot such that it deviates from the demonstrated path. The magnitude of these deviations is then used analogously to stiffness. Pastor et al. (2011) introduced a method for real-time adaptation of demonstrated trajectories for task exertions, depending on the measured sensory data. They developed an adaptive regulator for learning and adaptation of demonstrated motion, where actual and learned force feedback was utilized. Schindlbeck and Haddadin (2015) presented a unified impedance approach for safe online parameter adaptation, based on passivity and energy tanks. Furthermore, a review of online adaptation of impedance control parameters for human robot interaction is presented in (Ficuciello et al., 2015; Müller et al., 2018) and in-contact execution tasks are presented by Abu-Dakka and Saveriano (2020). To adapt the controller parameters one can estimate the compliance of the interaction between robot and environment. This approach is taken in Santos and Cortesão (2018), where the perceived stiffness is estimated and used for controller gain scheduling.

For tasks where impedance control cannot be utilized because of the physical limitation of the used systems, admittance control (Siciliano et al., 2009) can be used for teaching and execution of tasks in contact with the environment. A variable admittance controller, for safe human-robot interaction, where the admittance control parameters were adapted based on the passivity criterion was presented by Ferraguti et al. (2019). Furthermore, variable admittance control can be adopted in

human-robot cooperation tasks by means of learning (Dimeas and Aspragathos, 2014, 2015). Finally, Li et al. (2020) consider the robot to be in contact with both human and environment, similar to the condition considered in this paper.

It is important to ensure the stability and guarantee satisfactory performance despite varying controller gains and environmental compliance. Elegant stability conditions are set up in Müller et al. (2019) for a physical human-robot system with delays in the human reaction. The system considered in this paper has varying parameters in addition to input delays; thus, the methods presented in Briat (2014) may be used for the stability analysis and performance specification.

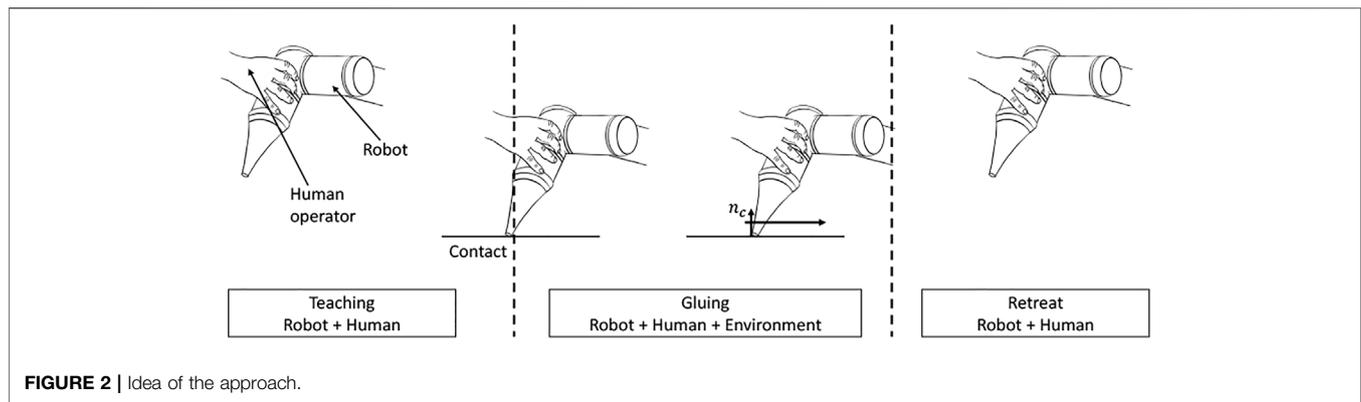
Recently, automated gluing applications have become very popular with the development of new industrial machines and tools, which can be efficiently utilized by collaborative robots. The majority of the work focuses on defining the robot trajectories based on the CAD description of the industrial object (Castelli et al., 2021). Other approaches, utilize simple Cartesian space industrial manipulators coupled with computer vision (Pagano et al., 2020). The vision system enables localization and reconstruction of the shape of objects to which glue should be applied. The gathered information is used to construct a motion trajectory that can be executed with the specialized manipulators, achieving the desired industrial performance.

3 PROBLEM FORMULATION AND CONTRIBUTIONS

Given a gluing or similar in-contact processing or dispensing task, our objective is to develop a full pipeline comprising the demonstration, encoding and execution that enables a robust and high-quality execution of the task with as little setup as possible required from the end-user. In order to make the approach more universally deployable in an industrial setting, we propose to learn from a single user demonstration, while exploiting the semantics of the task to adapt the robot controllers.

A key characteristic of the addressed task is that it is composed of free-air approach and retraction motions, but the actual dispensing or gluing consists in an in-contact motion whereby the manipulator is partially constrained by the environment. In particular, this constraint will be given by a surface normal during contact. In addition to this, in the proposed learning from demonstration scenario using kinesthetic teaching, the human is also in contact with the robot, thus partially constraining its motion as well.

An illustration of how these concepts fit into our system pipeline is shown in **Figure 2**. From an initial approach motion consisting only of the robot in contact with the human, the human will then move the robot into contact with the target surface and perform a sliding motion for the actual dispensing. This will correspond to a state where the entire system dynamics are comprised of the robot, the human and the environment. Finally, a retraction motion will move the robot away from the surface and into a final state where the dynamics will be given by the robot and human. If we consider the subsequent autonomous execution of the task, the dynamics will be the same, with the exception of the human's contribution.



We consider our main contribution to be our controller design for kinesthetic teaching of processing tasks. We propose an adaptive controller architecture that takes into account the different constraints of the task at different phases of the execution. In particular, our approach to demonstration comprises the following:

- 1) We learn continuous kinematic and dynamic task constraints from a single demonstration.
- 2) We use variable-gain admittance control for kinesthetic teaching, where the damping is continuously adapted based on a velocity-force rule in order to match the user's intention and reduce the physical effort needed during teaching.
- 3) We couple this controller with a contact detection algorithm, such that we are able to detect the start of the dispensing phase and increase controller damping to guarantee stability, even in cases of sudden impact with a stiff environment.
- 4) Once contact with a surface is established, we additionally couple our gain scheduler with a surface normal estimator. The formulation of this estimator is guaranteed to converge given adequate excitation and furthermore guarantees the preservation of the unit norm. By estimating the surface normal, we are able to adapt the eigenstructure of gain matrices, such that we maintain higher damping in the direction(s) constrained by the environment, while being able to lower the damping—and thus assist the user—in the non-constrained directions.

In addition to our controller for demonstration, we also propose a second controller for task execution. This controller adopts a similar structure to the one used for teaching, albeit with a few modifications, and exhibits the following characteristics:

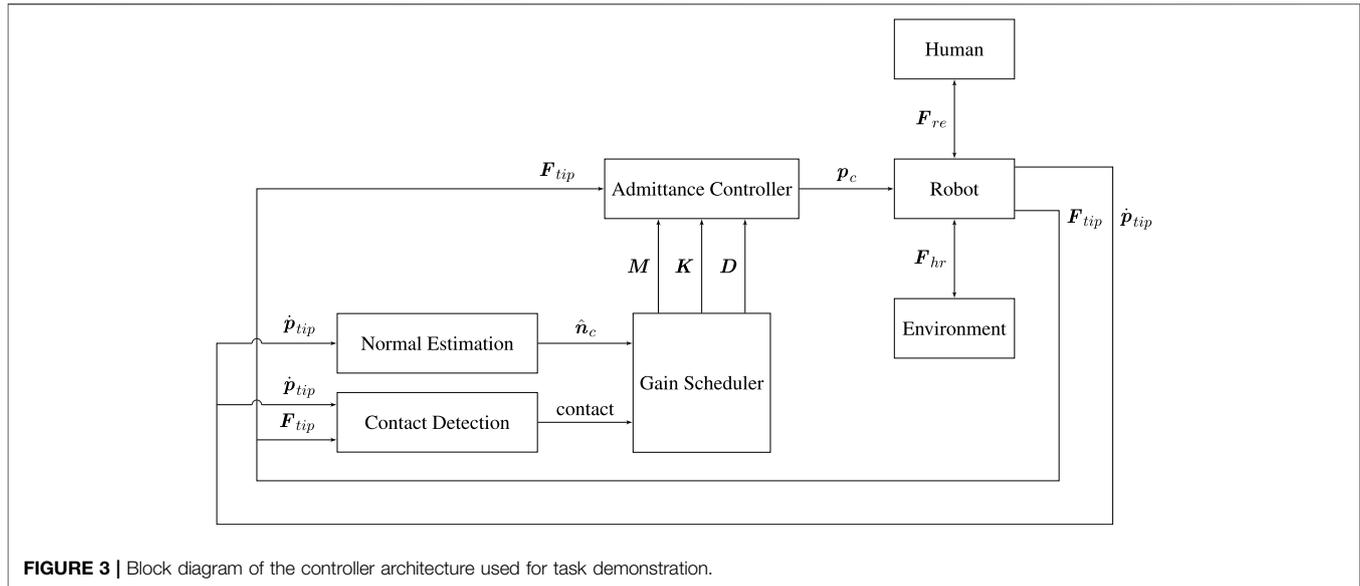
- 1) We use parallel position/force control to guarantee that both the kinematic and dynamic targets of the task are met.
- 2) Task kinematics are encoded as Dynamic Movement Primitives (DMPs). This allows us to learn from a single demonstration, while adopting an encoding that enables modulation of the motions. We specifically use modulation of the DMP goal to ensure a smooth transition between the approach and process motions, and to reduce the impact forces upon first contact with the target surface.

- 3) Task dynamics are preserved by encoding the output of the normal estimator during demonstration as Radial Basis Functions (RBFs) synchronized with the DMPs. By applying a user-defined process force magnitude in the direction of the encoded normal vector, we ensure that contact with the surface is maintained. We also use this information to vary the gains of the parallel position/force controller, such that force control is applied only on the normal axis, while the other axes are position-controlled.

3.1 Comparison to Other Approaches

Our work is conceptually similar to Kronander and Billard (2012); Kober et al. (2015); Lee et al. (2015); Conkey and Hermans (2019), in that we are interested in learning task constraints from demonstration, such that we can then adapt the parameters/gains of a compliant controller to the structure of the task. However, in contrast to other approaches (Kronander and Billard, 2012; Kober et al., 2015; Lee et al., 2015), which require either multiple (typically more than three) demonstrations of the task or multiple execution iterations in order to determine these constraints, our approach can do so from a single demonstration by exploiting the surface normal estimator to determine the task geometry. We consider this one of the main contributions of our proposed approach, as single-demonstration LbD methods are more widely applicable in industrial scenarios.

Our work is methodologically most similar to Conkey and Hermans (2019). While their method is also able to learn task constraints from a single demonstration, they rely on the forces observed during the demonstration to do so. Such an approach assumes that the demonstration forces correspond with the forces of an ideal execution and, furthermore, does not consider that the forces measured by the sensor during demonstration will not only be task/environmental forces, but will also be coupled with those applied by the human demonstrator to kinesthetically move/teach the robot. These two sets of forces will not always be aligned; thus, using them to learn task constraints will result in an alignment error in the constraint frame. By using the estimated surface normal as the desired direction that the forces should be applied in, our approach does not suffer from this coupling. The magnitude of the desired force can simply be specified by the user



or could, e.g., be computed from the average of the magnitude of the demonstration forces in the normal direction.

As indicated above, in order to reduce contact force transients during the execution, we modulate the goal of the DMP slowly into the contact surface. Although this method is similar to Conkey and Hermans (2019), our use of it is qualitatively different. Whereas Conkey and Hermans (2019) use goal modulation reactively to adapt to unforeseen generalizations of the task—i.e., a contact that is established either earlier or later than anticipated—we do so proactively by purposely setting the initial goal higher than originally demonstrated, to ensure that impact forces are always reduced.

Note also that contrary to the above-mentioned works by using surface normal estimation as the basis for determining task constraints, our approach is able to use these constraints not only to adapt the controller during the execution but also during the demonstration phase to actively aid the user. This is, indeed, not possible for approaches that determine task constraints a posteriori based on positional variance or segmentation (Kronander and Billard, 2012; Kober et al., 2015; Lee et al., 2015).

4 METHODOLOGY

The purpose of this section is to describe the proposed approach to obtain high-quality robotic gluing based on learned behaviors from kinesthetic teaching. It is challenging to learn the gluing behavior, as it is a task that requires contact between robot and environment; thus, both the motion and environment geometry and dynamic parameters should be learned from a single demonstration. As it is chosen to use only one demonstration for learning the task, the parameters estimated from the demonstration should be improved from executions of the task. We describe the entire process of 1) Programming by demonstration, 2) Encoding of robot behavior, and 3) Execution of desired gluing control; however, the main focus

of the paper is the demonstration phase, as this is the novel contribution of the paper.

We propose a unified controller structure for the kinesthetic teaching and execution of tasks with interaction between robot and environment. **Figure 3** is a block diagram of the controller that runs during demonstration. The controller is an adaptive admittance controller, where the adaptation is conducted based on the surface normal to the environment when the robot and environment is in contact. This ensures stability and improves the quality of the demonstrated behaviors. A similar control structure is used for the execution, as shown in **Figure 5**; however, a parallel position/force controller is used instead of the admittance controller.

4.1 Adaptive Admittance Control for Kinesthetic Demonstration

The purpose of this section is to describe the admittance controller that is used for the demonstration of the gluing task. The admittance controller uses the notation from Caccavale et al. (1999) and is shown below

$$\begin{aligned} & \begin{bmatrix} M_p & 0 \\ 0 & M_o \end{bmatrix} \begin{bmatrix} \Delta \ddot{p}_{cd} \\ \Delta \dot{\omega}_{cd}^d \end{bmatrix} + \begin{bmatrix} D_p & 0 \\ 0 & D_o \end{bmatrix} \begin{bmatrix} \Delta \dot{p}_{cd} \\ \Delta \omega_{cd}^d \end{bmatrix} \\ & + \begin{bmatrix} K_p & 0 \\ 0 & K_o \end{bmatrix} \begin{bmatrix} \Delta p_{cd} \\ \epsilon_{cd}^d \end{bmatrix} = \begin{bmatrix} f \\ \mu^d \end{bmatrix}, \end{aligned} \quad (1)$$

where $M_p, M_o, D_p, D_o, K_p, K_o$ are 3×3 matrices, $f \in \mathbb{R}^3$ is the force applied to the end-effector of the robot given in Base frame, $\Delta p_{cd} = p_c - p_d$, $\mu^d \in \mathbb{R}^3$ is the torque applied to the end-effector given in desired frame, $\Delta \omega_{cd} = \omega_c - \omega_d$, and $\epsilon_{cd}^d = \eta_d \epsilon_c - \eta_c \epsilon_d - S(\epsilon_c) \epsilon_d$. The skew-symmetric operator is denoted by S and the unit quaternion is (η, ϵ) , where ϵ is the vector-part.

The objective of this paper is to adapt the controller according to the interaction with both human and environment. In particular the interaction with the environment can cause instability if the

controller gains are not adapted. Therefore, the surface normal is estimated during demonstration to allow the damping of the controller to be increased only in the direction of the environment.

4.1.1 Surface Normal Estimation

The surface normal is estimated during contact using the following sliding condition

$$n_c^T \dot{p}_c = 0, \quad (2)$$

where n_c is the surface normal and \dot{p}_c is the velocity of the contact point given by

$$\dot{p}_c = \dot{p}_e + S(\omega)R_e e^r, \quad (3)$$

and ω is the angular velocity of the end-effector, and R_e is a rotation matrix from base frame to end-effector frame, and e^r is a vector from the end-effector to the contact point, which is assumed to be constant.

Since e^r is assumed to be known, the estimate of n_c can be accomplished based on the estimation method presented in Karayiannidis et al. (2014) summarized in the following proposition. The proposition uses the projection matrix onto the orthogonal complement of a column vector $a \in \mathbb{R}^3$, defined as $\bar{P}: a \mapsto I - aa^T$, where I is a 3×3 identity matrix.

Proposition 1. *The integral adaptive law*

$$\dot{\hat{n}}_c = -\gamma_n \bar{P}(\hat{n}_c) L_n(t) \hat{n}_c \quad (4)$$

$$\dot{L}_n = -\beta_n L_n + \frac{1}{1 + \|\dot{p}_c\|^2} \dot{p}_c \dot{p}_c^T, \quad L_n(0) = 0, \quad (5)$$

where $L_n \in \mathbb{R}^{3 \times 3}$ guarantees that

- 1) the norm of the estimate $\hat{n}_c(t)$ is invariant, i.e., given that $\|\hat{n}_c(0) = 1\|$, $\|\hat{n}_c(t)\| = 1$ for all $t \geq 0$
- 2) if $\vartheta(0) \in (-\frac{\pi}{2}, \frac{\pi}{2})$ then $\vartheta(t) \in (-\frac{\pi}{2}, \frac{\pi}{2})$ for all $t > 0$, where ϑ is the angle between n_c and \hat{n}_c
- 3) $\lim_{t \rightarrow \infty} \|\hat{n}_c\| = 0$
- 4) if \dot{p}_c is persistently excited then ϑ converges to zero exponentially which implies that \hat{n}_c converges exponentially to n_c with a rate that can be tuned by γ_n .

The proposition is based on Theorem 4.3.3 in Karayiannidis et al. (2014) and ensures that \hat{n}_c remains a unit vector. The surface normal estimate is initialized based on the velocity \dot{p}_c when contact is detected, i.e.,

$$\hat{n}_c(0) = \frac{\dot{p}_c(0)}{\|\dot{p}_c(0)\|}$$

The projection matrix $\bar{P}(\hat{n}_c)$ in Eq. 4 ensures that the norm of \hat{n}_c is preserved, since \hat{n}_c is tangent to the unit sphere; thus, we can compute the instantaneous angular velocity $\omega(t)$ from $\hat{n}_c(t)$ and $\dot{\hat{n}}_c(t)$ as

$$\omega(t) = \hat{n}_c(t) \times \dot{\hat{n}}_c(t)$$

This property must be preserved when implementing the algorithm in discrete time. To accomplish this, we exploit the

instantaneous angular velocity $\omega(t)$ for the update of the surface normal estimate. In particular, we implement the surface normal estimation using the following difference equations:

$$\hat{n}_c(k+1) = e^{T_s S(\omega(k))} \hat{n}_c(k) \quad (6)$$

$$L_n(k+1) = (1 - T_s \beta_n) L_n(k) + T_s \frac{1}{1 + \|\dot{p}_c\|^2} \dot{p}_c \dot{p}_c^T, \quad L_n(0) = 0 \quad (7)$$

where T_s is the sample time and

$$\omega(k) = \hat{n}_c(k) \times (-\gamma_n L_n(k) \hat{n}_c(k)).$$

4.1.2 Contact Detection

A statistical method is used for contact detection; in particular, we consider the signal

$$\varphi = \dot{p}_{\text{tip}}^T f_{\text{tip}}, \quad (8)$$

which will be negative when a collision occurs since a force is applied in the opposite direction of the motion. To achieve a robust collision detection, the non-restarting cumulative sum method (CUSUM) is applied to φ (Gandy and Lau, 2013).

4.1.3 Parameter Adaptation

This section describes how the gains of the admittance controller are changed based on the dynamics of the demonstrator and the environment. Our approach is to change the eigenaxis of the parameter matrices according to the surface normal, and change the eigenvalues according to a force-dependent heuristic.

There exist multiple approaches for adapting the gains of an admittance controller, including force-dependent variable impedance control (Müller et al., 2018), velocity-dependent variable impedance control (Ficuciello et al., 2015), and passivity-based approaches (Ferraguti et al., 2019).

A force-dependent variable impedance control makes it easier to accelerate and decelerate the robot, as the damping is given by (Müller et al., 2018):

$$D_{VIC}(\Gamma) = -\alpha \text{diag}(\Gamma) \text{diag}(\text{sign}(\dot{x})) + D_0, \quad (9)$$

where Γ is the wrench vector, $\alpha > 0$ is a tuning parameter, \dot{x} is the Cartesian velocity, and D_0 is a diagonal matrix.

To ensure stability of the impedance controller when interacting with a human that varies the active damping and stiffness, a lower bound can be found for the damping that ensures stability; this is denoted D_{obs} . Finally, the damping of the force-dependent variable impedance control is given by

$$D = \max\{D_{VIC}, D_{obs}\}. \quad (10)$$

To analyze the stability of the impedance controlled robot during interaction with both environment and human, one can describe the system as a linear parameter varying (LPV) system. The varying parameters include the damping of the impedance control, the compliance of the human and the compliance of the environment. The stability of such

systems has been considered in several work, including Müller et al. (2019), but here several parameters are considered to be constant. The presented stability criterion relaxes this assumption.

The model used in similar physical human-robot collaboration works such as Müller et al. (2019) is linear, but most of the gains are varying in the considered application. In particular, the compliance parameters of the human and stiffness of the environment may vary. These parameters may be estimated or considered to be unknown but bounded to some set of feasible parameter values. Also the damping of the controller is varying but known (in the force-dependent variable impedance control, it depends on the measured wrench). Finally, delays are present in the system originating from the active response of the human and delays in the control software.

The variations in the system dynamics may be modelled using an N -dimensional exogenous variable $\rho(t)$, but since the response of the controller may be delayed compared to the response of the system, the delayed variable $\rho_h(t) = \rho(t - h)$ is introduced where $h \in [0, \bar{h}]$ is a constant time-delay.

Due to the linearity of the considered system dynamics, the impedance controlled robot can be modelled as the following LPV system with time-delays, see Briat (2014),

$$\dot{x}(t) = A(\rho(t), \rho_h(t))x(t) + A_h(\rho(t), \rho_h(t))x(t - h) + E(\rho(t), \rho_h(t))w(t) \tag{11}$$

$$z(t) = C(\rho(t), \rho_h(t))x(t) + C_h(\rho(t), \rho_h(t))x(t - h) + F(\rho(t), \rho_h(t))w(t) \tag{12}$$

$$x(s) = \phi(s), s \in [-\bar{h}, 0], \tag{13}$$

where x is the state of the system, w is a disturbance input, z is the performance output, A , A_h , and E define the system dynamics, and C , C_h , and F define the performance measure, and $\phi(s)$ provides the initial condition. The value of the parameter vector ρ must be within the compact set Δ_ρ , and the rate of variation of the parameters should be within the set $\Delta_v = [-1, 1]^N$. To simplify notation, we define the Hermitian operator (for real matrices) as $\text{He}(A) = A + A^T$, denote the set of positive definite $n \times n$ matrices by $\mathbb{S}_{>0}^n$, and define the following sets:

$$\mathcal{P}^v = \{\rho: \mathbb{R}_+ \rightarrow \Delta_\rho: \dot{\rho}(t) \in \Delta_v, t \geq 0\} \tag{14}$$

$$\mathcal{P}_h = \{(\rho, \rho_h): \mathbb{R}_+ \rightarrow \Delta_\rho \times \Delta_\rho: \rho \in \mathcal{P}^v, \rho_h(t) = \rho(t - h), t \geq 0\} \tag{15}$$

$$\Delta_\rho^h = \{(\rho, \rho_h) \in \Delta_\rho \times \Delta_\rho: |\rho_{hi} - \rho_i| \leq \bar{h}, i = 1, \dots, N\}. \tag{16}$$

The stability of the time-delayed LPV system can be analyzed by the use of the following theorem (Briat, 2014, Theorem 6.3.3), which is based on identifying a parameter-dependent Lyapunov-Krasovskii functional due to the existence of time-delays. Also, a desired performance can be specified via a mixed sensitivity description and optimized by minimizing γ in the following theorem.

Theorem 1. Assume that there exists a continuously differentiable matrix function $P: \Delta_\rho^h \rightarrow \mathbb{S}_{>0}^n$, a matrix function

$Q: \Delta_\rho \rightarrow \mathbb{S}_{>0}^n$, a constant matrix $R \in \mathbb{S}_{>0}^n$ and a scalar $\gamma > 0$ such that the LMI

$$\begin{bmatrix} \Psi_{11} & \Psi_{12} & P(\rho, \rho_h)E(\rho, \rho_h) & C(\rho, \rho_h)^T & \bar{h}A(\rho, \rho_h)^T R \\ * & -Q(\rho_h) - R & 0 & C_h(\rho, \rho_h)^T & \bar{h}A_h(\rho, \rho_h)^T R \\ * & * & -\gamma I_p & F(\rho, \rho_h) & \bar{h}E(\rho, \rho_h)^T R \\ * & * & * & -\gamma I_q & 0 \\ * & * & * & * & -R \end{bmatrix} < 0 \tag{17}$$

holds for all $\rho, \rho_h \in \Delta_\rho^h$ and all $\nu, \nu_h \in \{-1, 1\}^N \times \{-1, 1\}^N$ where

$$\Psi_{11} = \text{He}[P(\rho, \rho_h)A(\rho, \rho_h)] + \sum_{i=1}^N \left(\nu_i \frac{\partial P(\rho, \rho_h)}{\partial \rho_i} + \nu_{hi} \frac{\partial P(\rho, \rho_h)}{\partial \rho_{hi}} \right) + Q(\rho) - R \tag{18}$$

$$\Psi_{12} = P(\rho, \rho_h)A(\rho, \rho_h) + R \tag{19}$$

Then the LPV system is asymptotically stable for all constant delays $h \in [0, \bar{h}]$ and $(\rho, \rho_h) \in \mathcal{P}_h$ and the L_2 -gain from w to z is less than γ .

Theorem 1 applies to the considered robotic system; however, to ease the computation of the stability condition, the system model should be written as a polytopic LPV system. Then it is sufficient to evaluate Eq. 17 at the vertices of the parameter space, see Briat (2014) for details.

4.2 Learning Task Kinematics

In order to encode a gluing trajectory from a user demonstration in a flexible manner that will allow us to reparameterize and reuse the learned task, we adopt Dynamic Movement Primitives as our representation for robot kinematic trajectories.

4.2.1 Cartesian-Space Dynamic Movement Primitives

In SE(3), dynamic movement primitives (Schaal, 2006; Ijspeert et al., 2013; Ude et al., 2014) represent robot trajectories as a second order dynamical system of the form:

$$\tau \begin{bmatrix} \dot{z} \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} \alpha_y (\beta_y (g - \gamma) - z) \\ \alpha_o (\beta_o 2 \log(g_o * \bar{q}) - \eta) \end{bmatrix} + \begin{bmatrix} f_p(x) \\ f_o(x) \end{bmatrix} \tag{20}$$

$$\tau \begin{bmatrix} \dot{y} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} z \\ \frac{1}{2} \eta * q \end{bmatrix} \tag{21}$$

$$\tau \dot{x} = -\alpha_x x, \tag{22}$$

where $\alpha_y, \alpha_o \in \mathbb{R}^+$ and $\beta_y, \beta_o \in \mathbb{R}^+$ are constant gains, $g \in \mathbb{R}^3$ and $g_o \in \text{SO}(3)$ are attractor (goal) points for the positional and orientational components of the DMP, respectively, $\tau \in \mathbb{R}^+$ is a time constant corresponding to the duration of the movement, x is a phase variable, q is a unit quaternion, and $*$ denotes the quaternion product. Notice that in the absence of terms $f_p(x)$ and $f_o(x)$, Eqs 20, 21 behave as a mass-spring-damper, and will converge towards g and g_o regardless of their initial condition. In this sense, constants $\alpha_y, \alpha_o, \beta_y$ and β_o are usually chosen for critical damping. By adding the forcing terms $f_p(x)$ and $f_o(x)$, given by Eqs 23, 24, the system is modulated to fit an arbitrary trajectory, typically provided by demonstration. The forcing terms

$$f_p(x) = \frac{\sum_{i=1}^N \Psi_i(x) w_i^p}{\sum_{i=1}^N \Psi_i(x)} x, \quad (23)$$

$$f_o(x) = \frac{\sum_{i=1}^N \Psi_i(x) w_i^o}{\sum_{i=1}^N \Psi_i(x)} x, \quad (24)$$

$$\Psi_i(x) = \exp(-h_i(x - c_i)^2), \quad (25)$$

consist of $N \in \mathbb{Z}^+$ Gaussian radial basis functions (RBF) (Eq. 25) with centers c_i , widths h_i and attached weight vectors $w_i \in \mathbb{R}^3$.

The canonical system Eq. 22, controls the time evolution of the system and at the same time guarantees convergence of Eqs 20, 21, as the influence of $f_p(x)$ and $f_o(x)$ will vanish as $x \rightarrow 0$.

Given a demonstration trajectory $\{y_{\text{demo}}, \dot{y}_{\text{demo}}, \ddot{y}_{\text{demo}}\}$, Eqs 20, 21 can be reformulated:

$$f_{p,\text{desired}} = \tau^2 \ddot{y}_{\text{demo}} - \alpha_y (\beta_y (g - y_{\text{demo}}) - \tau \dot{y}_{\text{demo}}), \quad (26)$$

such that the open parameters in Eqs 23, 24, can be learned by, e.g., least-squares weighted linear regression. The same approach can similarly be applied to the orientation.

4.3 Encoding Task Dynamics

Process tasks, such as gluing, involve dynamic interaction with the environment and require that contact between the robot and the workpiece be maintained throughout the task execution. This implies that a target force and torque trajectory must be generated and inputted as a feed-forward signal into an appropriate robot controller (such as a hybrid or parallel position/force controller).

Typically, force-learning within the context of DMPs has consisted of running the robot task open-loop once the kinematic trajectory has been encoded, while the task interaction forces and torques are recorded. These force and torque profiles are assumed to correspond to the desired task profiles and can then be encoded as a mixture of RBFs (Nemec et al., 2013; Abu-Dakka et al., 2015).

As the kinematic trajectory taught by the user is not likely to be optimal, the forces and torques learned using the previously mentioned approach might not be desirable and therefore not result in the best possible task execution. This is particularly the case for process tasks such as gluing, where often a constant contact force is necessary in order to ensure even distribution of glue along the target surface. Thus, we propose to instead record the output of the surface normal estimator (see Section 4.1.1) during the user demonstration and encode this as RBFs:

$$n_{c,d}(x) = \frac{\sum_{i=1}^K \Psi_i(x) w_i^{n_c}}{\sum_{i=1}^K \Psi_i(x)} x. \quad (27)$$

The user can then manually input a desired force magnitude F_{in} —which is often a known process variable—and this will then be applied in the direction of the surface normal, i.e. the desired force is $f_d(x) = F_{in} n_{c,d}(x)$.

4.4 Execution

Given the encodings of both the task kinematics (Section 4.2) and dynamics (Section 4.3) the controller chosen for task execution

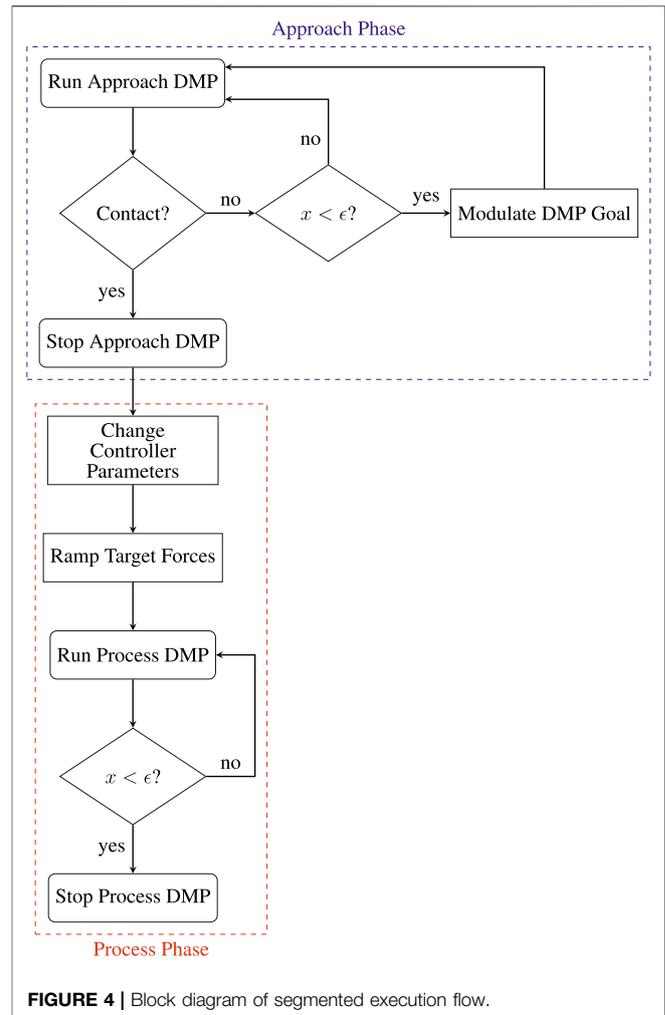


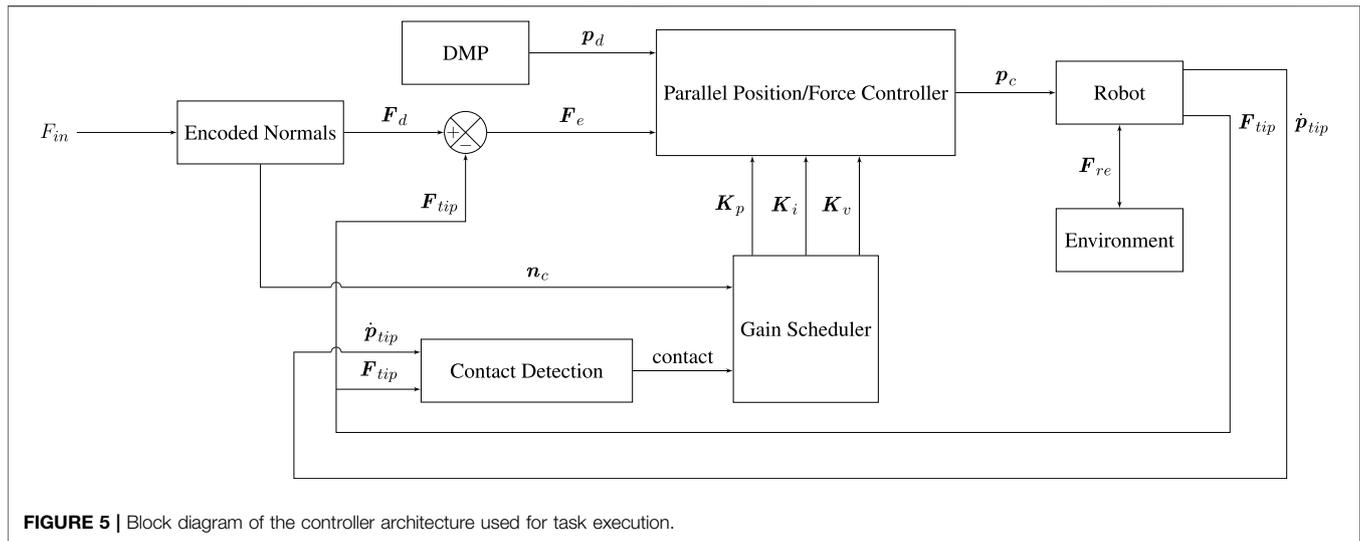
FIGURE 4 | Block diagram of segmented execution flow.

must ensure tracking of both the target position and orientation and the target forces and torques. Simultaneously, it must handle the prioritization of these two tasks in a way that guarantees both the successful outcome of the task and the stability of the system. Here, we apply segmentation in order to more effectively handle the different requirements of different stages of the task, as well as to allow us to apply strategies to increase the robustness of the execution.

The remainder of this section will present our approach to task segmentation and how it applies to task execution, and our proposed parallel position/force controller.

4.4.1 Segmentation and Execution Flow

A key aspect to our overall approach is segmentation, whereby we divide the overall task into an *approach phase* and a *process phase* based on the output of the contact detector during the demonstration. The approach phase includes all robot motions prior to establishing contact with the target surface, while the process phase includes all in-contact interactions with the surface, as well as retraction motions where the robot leaves the surface. Technically, a third *retraction phase* could easily be defined to encompass these motions, but this has not been implemented in the current system iteration.



Segmentation of the task phases is crucial to our method, as it allows us to use the semantic information contained in the different phases to apply appropriate control schemes and approaches to generalization. Namely, the execution is split into two separate DMPs, one for each of the phases, and these DMPs are run with different termination conditions. **Figure 4** shows the segmented execution flow.

The approach DMP is modulated in a non-standard way by raising its goal a preset distance in the direction normal to the initial contact point in the surface. Once the robot reaches this point, if contact has not been established, the DMP goal is slowly moved down along the direction of the surface normal into the surface. This approach allows us to minimize impact forces, thus ensuring safe interaction with the PCB. A suitable trade-off between execution speed and impact forces can be achieved by regulating the speed of modulation of the goal, depending on the specific application. While this approach is similar to Conkey and Hermans (2019), note that they use it in a reactive manner, where the goal is modulated downwards towards the surface only in the case that contact is not established at the same point as it occurred in the demonstration, whereas we use it proactively in order to purposely reduce impact forces.

Once contact has been established with the surface, the starting point of the process DMP will be modulated to coincide with the robot position at contact, in this way avoiding discontinuities. Next, the parallel position/force controller parameters will be changed to appropriate values for the contact state. The target force will be obtained from the normal estimator, but instead of being directly inputted to the controller, will be ramped up slowly to avoid sudden overshoots. Finally, the process DMP will be run in parallel with the target forces obtained from the normal estimator. The next section will examine the details of the parallel position/force controller used for this purpose.

4.4.2 Parallel Force/Position Control

The purpose of this section is to describe the parallel force/position control applied for the execution of the gluing behavior. The controller uses the standard formulation from

(Siciliano et al., 2009, Section 9.4.3). The force-controller component is chosen to be a PIV control, i.e., a PI control on the force with velocity damping as explained in (Pérez-Ubeda et al., 2020). In addition, the controller gains are adapted according to the surface normal. The block diagram shown in **Figure 5** shows the use of the controller during execution of the gluing task.

Notice the structural similarities with the admittance controller used for demonstration in **Figure 3**. The most notable differences are the removal of the normal estimation block and addition of a DMP and encoded normals block. The DMP provides the positional input to the parallel position/force controller. The encoded normals are used for two purposes: 1) In combination with an input process force, F_{in} provided by the user they specify the force input to the parallel position/force controller. 2) Through the gain scheduler, they specify the structure for the gain matrices of the parallel position/force controller, such that force-control is applied only in the axis perpendicular to the surface of contact, while the other two axes are purely position-controlled. Note that contact detection is also incorporated in a manner similar to during the demonstration. In this case, while contact has not yet been detected, the gains of the force-control component of the parallel position/force controller will be set to zero, such that the robot is purely position controlled. Once a contact is detected, gain-switching occurs, and the controller begins tracking the desired forces, F_d , so as to minimize the force error, F_e .

5 EXPERIMENTAL EVALUATION

In order to evaluate the performance of our proposed system, we constructed a benchmark from a real-world use-case provided by a partner company. The original use-case involves gluing electronic components to a PCB. For our evaluation, we 3D-printed a replica of the PCB with the attached components. Similarly, we substituted the original gluing dispenser tip with

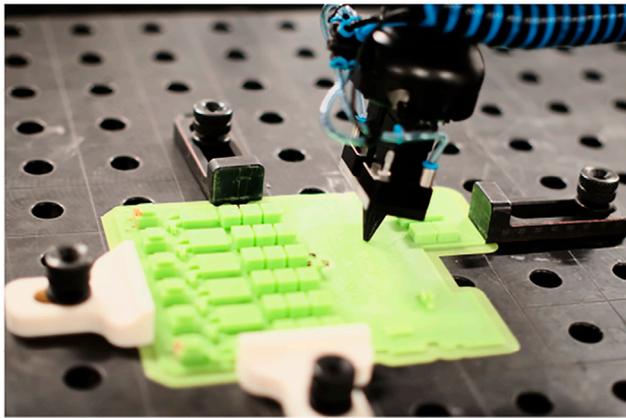


FIGURE 6 | Robot executing the taught gluing-like task on our benchmark system. Notice the 3D-printed mock-up PCB (in green) and the 3D-printed mock-up gluing tip in a conical shape attached to the robot end-effector.

a 3D-printed model. Both the PCB and gluing tip mock-ups can be observed in **Figure 6**. Note that, due to being 3D-printed, our mock-ups actually introduce more challenges from a control perspective than the original components. Printing imperfections on the surface of the PCB mock-up increase friction and introduce high frequency components in the forces that the parallel position/force controller must then address. The 3D-printed tip is also stiffer, which can challenge the stability of the system when compared to the more compliant true gluing tip.

We used a Universal Robots UR10e collaborative robot manipulator for our experiments. The robot is equipped with an internal force-torque (FT) sensor at the wrist 3 joint, close to the tool flange. All force-torque measurements and control were performed using the output of this sensor. The controllers were implemented in Python 3, using the *ur_rtde*¹ library for communication with the robot, both for streaming data and for sending commands to the internal robot controller. Our Python control application and the internal robot controller both run at a frequency of 500 Hz.

The following sections will evaluate our full system pipeline, from a demonstration of a task, as shown in **Figure 1**, to its segmentation, encoding and subsequent execution. We will present results for each of the system components previously described in **Section 4**. Note that all figures in the following sections are expressed in robot base frame, that is, with the z -axis normal to the table surface and its positive direction pointing out of the table, while the $x - y$ plane coincides with the table plane.

5.1 Demonstration and Encoding

We will first evaluate the components of the system involved in the demonstration and task encoding pipeline. This consists of

the surface normal estimator, damping adaptation scheme for the admittance controller, and DMPs for both the approach and process phases.

5.1.1 Admittance Controller Parameter Adaptation

Figure 7 shows adaptation of the damping term in the admittance controller used for kinesthetic teaching of the task, as outline in **Section 4.1.2**. Notice that the damping is varied slightly in all directions between $t = 0$ s and $t \approx 3$ s, that is during the free air (approach) phase of the task. This is followed by an abrupt increase in damping in all axes of both the position and the orientation) at $t \approx 3$ s, when contact with the surface of the PCB/table is detected. This ensures that the stability of the interaction is maintained despite the sudden increase in the environmental stiffness. The high damping value is maintained for another half a second, after which the parameters are allowed to vary again, albeit this time around the increased damping value as base value. This is particularly noticeable in the positional axes at $t \approx 8$ s. Here, the user attempted to increase the speed of movement in a direction parallel to the plane of contact. This caused a decrease in damping in the x - and y -axes, accompanied by a slight increase in damping in the z -axis, as the robot was not moving in this plane. Once the robot leaves the surface during the retraction motion at $t = 15$ s and is moved again in free air, the damping in the z -axis very rapidly decreases. In all of these cases, the damping adaptation aims to assist the user in accomplishing their demonstration intention.

5.1.2 Comparison of Admittance Controller vs. Freedrive for Demonstration

To test the usability of our adaptive admittance controller for demonstration, we devised an experiment where we compared it to the Free-drive control mode implemented in the Universal Robots. We programmed the robot to draw a reference figure-eight trajectory on a flat steel surface using a permanent marker. This was subsequently used as a visual reference in a small-scale user experiment consisting of six subjects. The subjects were instructed to kinesthetically teach the robot to follow the reference as closely as possible, without any requirements as to speed or time of the demonstration. Each subject repeated the demonstration three times with each controller, alternating between freedrive and our proposed adaptive admittance scheme.

To account for different timescales, we used Dynamic Time Warping (DTW) (Müller, 2007) to temporally align all demonstrations to the reference trajectory. We used DTW distance as a measure of deviation from the reference. Likewise, we measured the duration of the demonstration. The results are shown in **Table 1**, where each value is the average of the three attempts for that condition.

Although the sample size is not large enough to achieve statistical significance, the results show a clear trend that users achieve lower distances from the reference in a shorter amount of time when using our adaptive admittance controller compared to when using Freedrive.

5.1.3 Surface Normal Estimator

We evaluated the surface normal estimator on a separate curved surface based on an eigenfunction of the L-shaped membrane, as

¹https://gitlab.com/sdurobotics/ur_rtde.

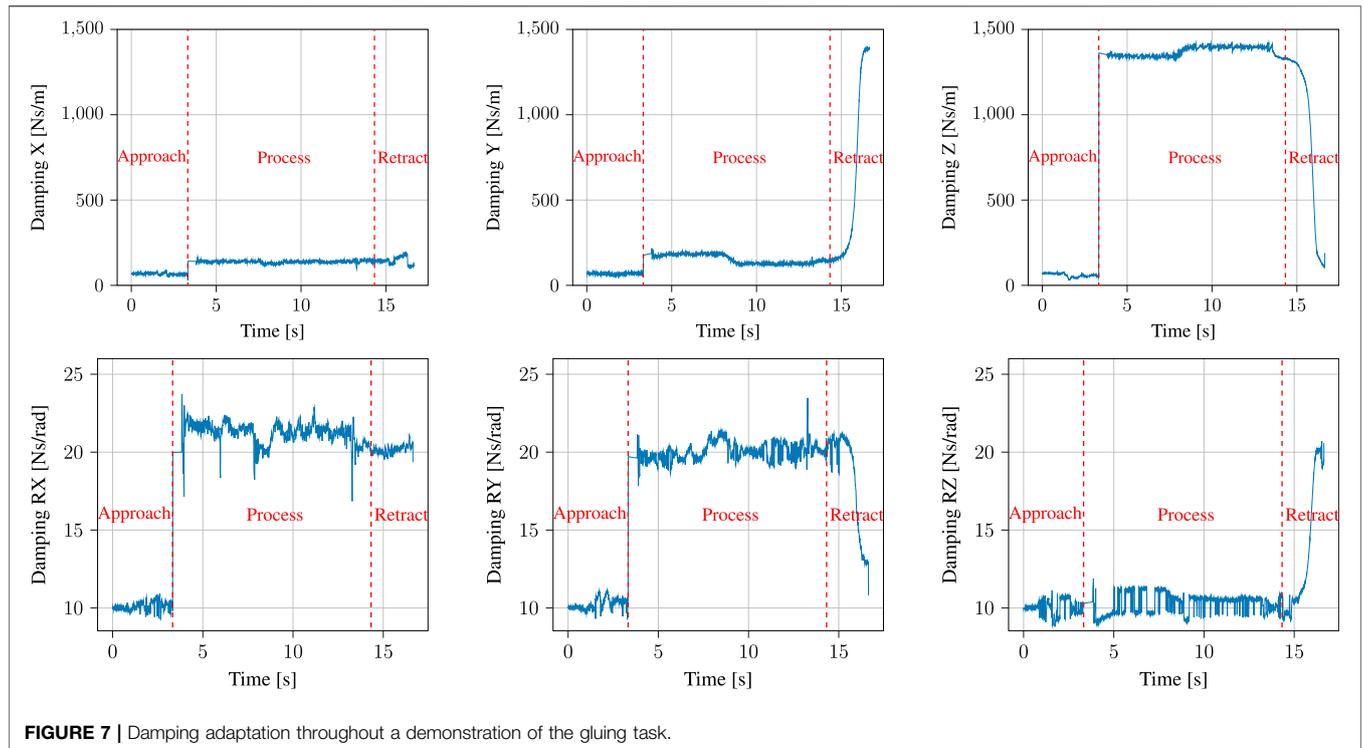


FIGURE 7 | Damping adaptation throughout a demonstration of the gluing task.

TABLE 1 | Comparison of the mean distance from reference (lower is better) and mean duration (lower is better) between our adaptive admittance controller and Universal Robots' freedrive controller for an experiment with six subjects. Each number is an average of three attempts performed by that subject.

Subject	Mean distance from reference		Mean duration	
	Adaptive admittance (ours)	Freedrive	Adaptive admittance (ours)	Freedrive
A	16.1	25.0	28.9	34.5
B	52.3	79.3	18.6	27.3
C	40.0	42.1	17.3	22.5
D	72.2	80.5	25.3	27.5
E	58.0	105.7	18.1	34.6
F	62.1	83.8	13.3	21.1
All	50.1	69.4	20.2	27.9

shown in Figure 8. This surface is composed of a flat plane in one of its quadrants, which transitions into a smooth continuous function along the other three quadrants. Using a compliant controller, the robot was taught a motion along the surface, starting on the planar quadrant and moving onto and along the surface peak. The surface normal estimator was initialized to the value $[0, 0, -1]$. Figure 9 shows the evolution of the angular error along the path. An error of $\sim 6^\circ$ is maintained along the planar quadrant of the surface. At $t = 4$ s the robot tool is moved onto the curved section of the surface, resulting in a discontinuous jump in the error up to $\sim 12^\circ$. This is to be expected, as the surface is in fact discontinuous at this point. The estimator then begins converging and the error decreases to 8° at $t = 5$ s. A maximum angular error of $\sim 16^\circ$ is reached at $t \approx 6.5$ s, coinciding with the area along the surface that exhibits the fastest rate of curvature change, after which the error decreases to $\sim 9^\circ$ at $t \approx 8.5$ s.

5.1.4 DMP Encoding

The segmented approach phase of the demonstration, together with the output of the corresponding DMP learned from it are shown in Figure 10 for the position and Figure 11 for the orientation, respectively. Likewise, the process phase is shown in Figures 12, 13. Notice, particularly in the y -axis of component of Figure 10 and quaternion w - and z -components of Figure 11 that some error in the least-squares fit for the RBFs is present. However, at the scale this is apparent, it has little effect on the robot motion and is irrelevant to the approach phase overall.

5.2 Execution

Now that we have examined the components involved in demonstration and encoding processes, we will analyze the results of execution, particularly with regards to the performance of the parallel position/force controller.

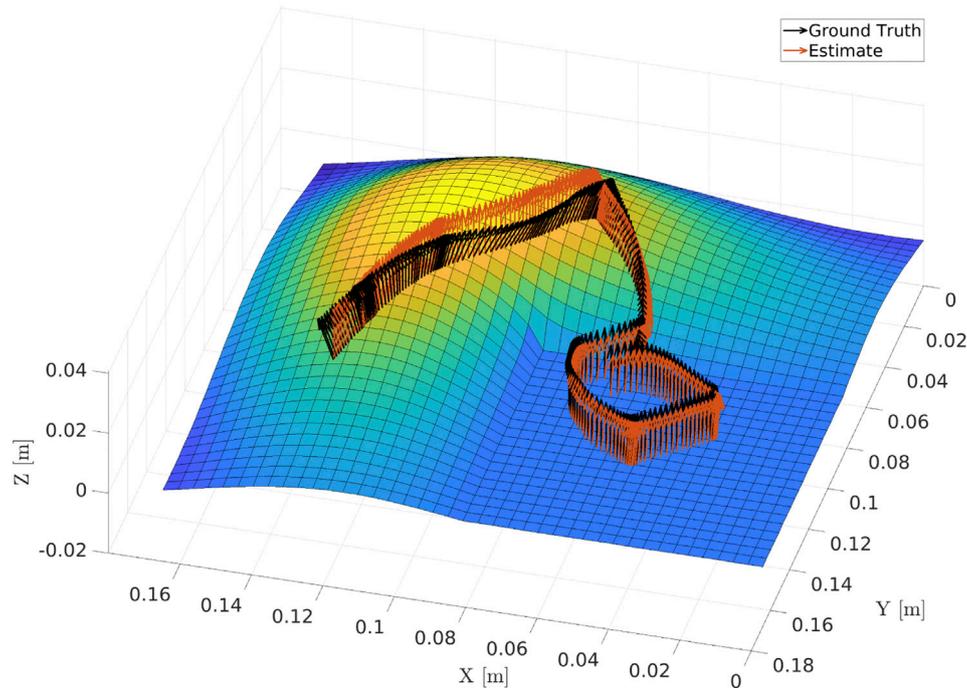


FIGURE 8 | Output of the surface normal estimator compared to the ground truth value for a taught robot tool path along a curved surface defined by an eigenfunction of the L-shaped membrane.

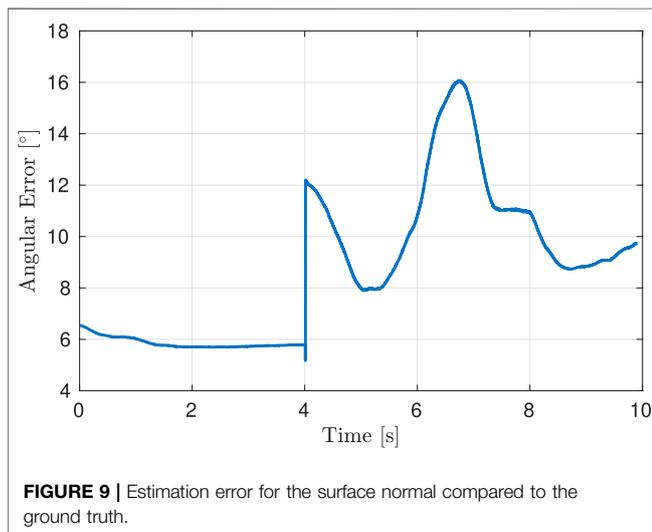


FIGURE 9 | Estimation error for the surface normal compared to the ground truth.

Figure 14 shows the target force reference of the controller, together with the actual forces measured by the robot f/t sensor end-effector during execution. The detected contact point with the surface of the PCB is indicated by a red dotted line at $t \approx 4.5$ s. Note that the output of the normal estimator previously shown in **Figure 8** multiplied by the user-inputted process force magnitude, which in this case was set to 5 N, is the target force. Furthermore, this means that the majority of the force control happens in the z -axis, as the robot was nearly perpendicular to

the contact surface throughout the task. We will consequently focus most of our analysis on the z -axis component. During contact, the robot experiences an impact force of ~ 9 N (together with ~ 2.5 N in the x -axis). Such a low impact force is achievable due to our slow modulation of the approach DMP goal, as described in **Section 4.4.1** and visualized in **Figure 4**. Following this, the force target is ramped up from 0 to 5 N between impact and $t \approx 7.5$ s, after which it maintains a nearly constant process force of 5 N. Notice that after $t \approx 17.5$ s, the robot is unable to maintain the target force. This is because the position reference moves it up and away from the contact surface, as can be seen in **Figure 15**. This can similarly be observed in the x -axis force, and is reflected again in the x -axis of **Figure 15**, as the prevalence of the force target means that the robot does not actually meet the position target. Since the retraction motion is not considered crucial to the successful completion of the task, it is currently not explicitly handled by the segmentation algorithm. Were it so, it would suffice with setting either the target force or the force control gains to zero in order to more precisely follow the position target.

A prevalent issue visible in **Figure 14** is the amount of noise present in the FT sensor measurements. Notice that this is noticeable even before impact, at $t < 4.5$ s, when the robot is in free air. This shows that this is an issue in the sensor used in the Universal Robots themselves, and not in our system. Indeed, similar performance to our results here has been reported by, e.g. Pérez-Ubeda et al. (2020) when applying force control on Universal Robots arms.

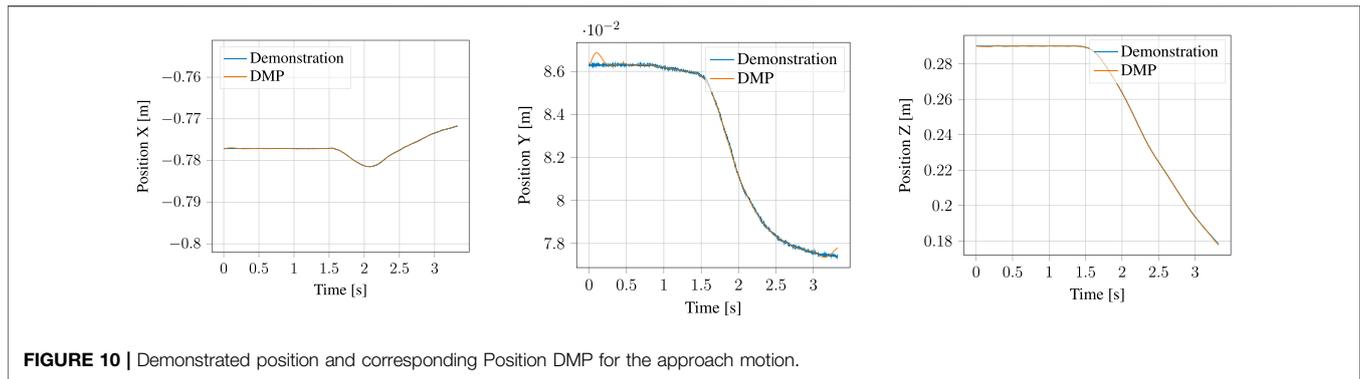


FIGURE 10 | Demonstrated position and corresponding Position DMP for the approach motion.

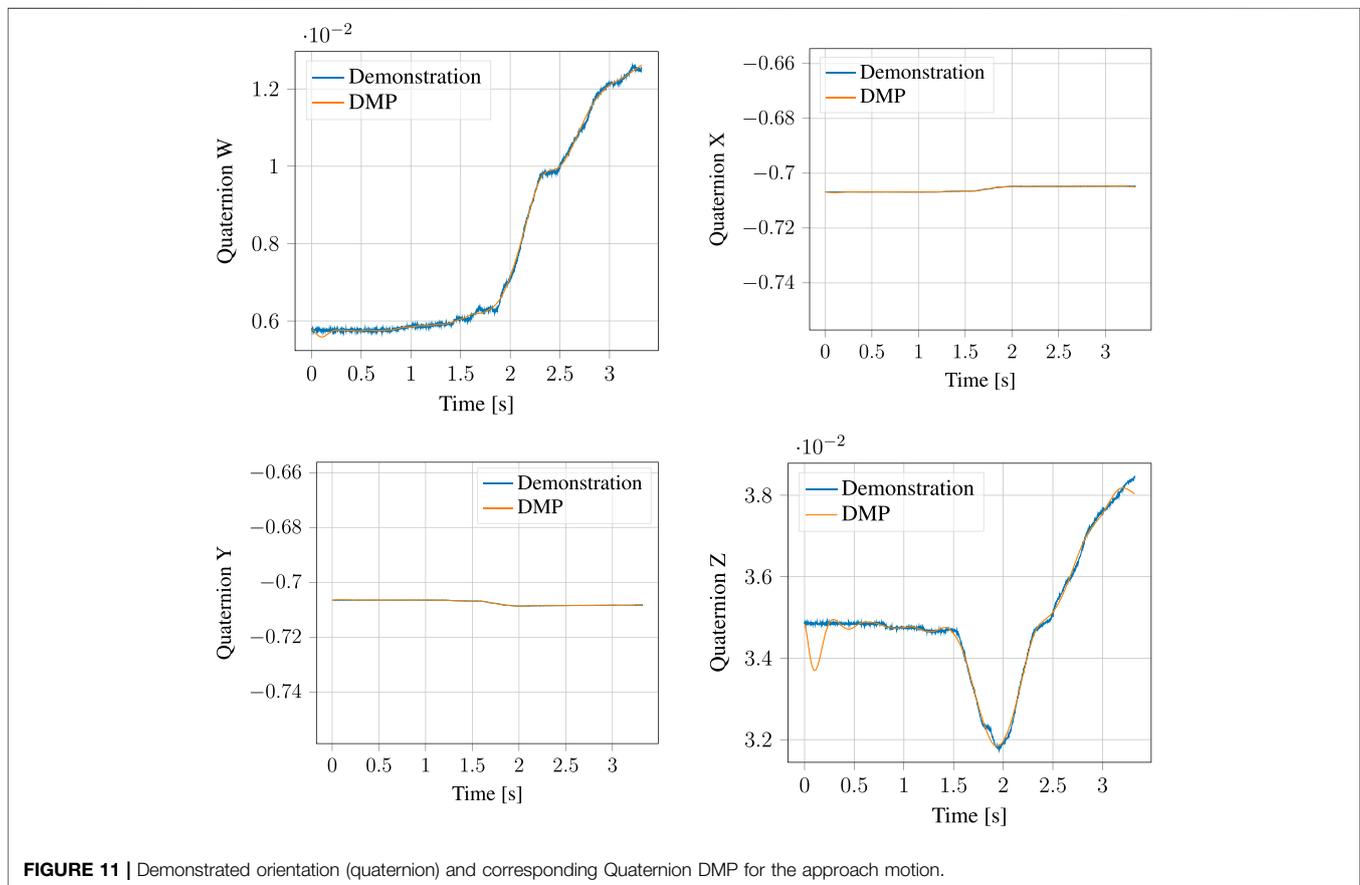


FIGURE 11 | Demonstrated orientation (quaternion) and corresponding Quaternion DMP for the approach motion.

6 DISCUSSION AND PERSPECTIVES

In the previous section, we evaluated our system on a benchmark inspired by a real industrial gluing use case, and analyzed the performance of the different system components both during a task demonstration and during the subsequent execution of the learned primitive. During demonstration, we showed that the adaptive damping in the admittance controller can assist the user in accomplishing their intended motion, while at the same time guaranteeing a stable interaction despite a changing environmental stiffness. Similarly, we showed that our surface normal estimator works satisfactory and allows the admittance

controller to change the structure of its gain matrices, such that the normal direction maintains a higher damping to ensure stability, while the damping in the other directions is reduced to assist the user. During execution, we showed that our controller is able to track positional targets while applying the target force profile in the normal direction. We also showed the system's ability to reduce impact forces by using DMP goal modulation.

However, there are many factors that were not fully considered and that could be improved upon in future iterations of the system.

When it comes to our adaptive admittance controller for demonstration, our small-scale usability evaluation in

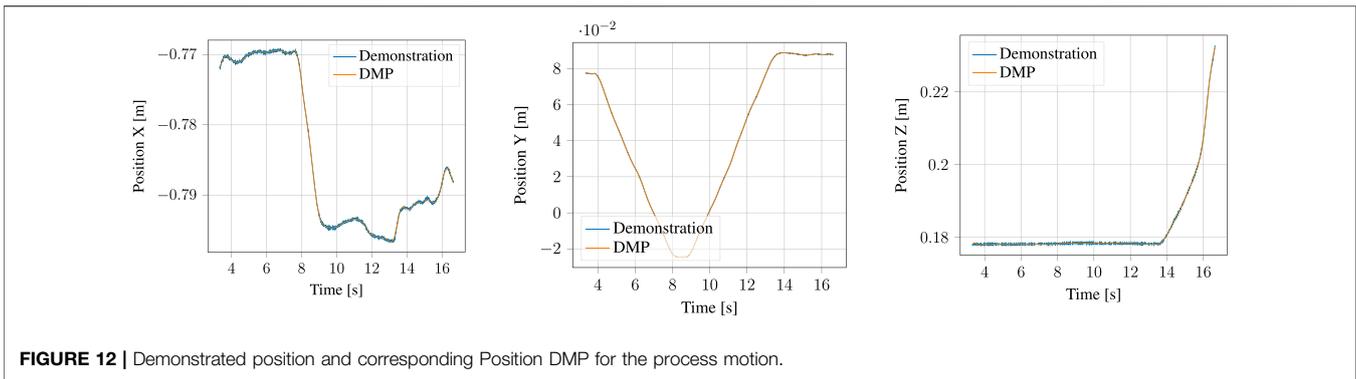


FIGURE 12 | Demonstrated position and corresponding Position DMP for the process motion.

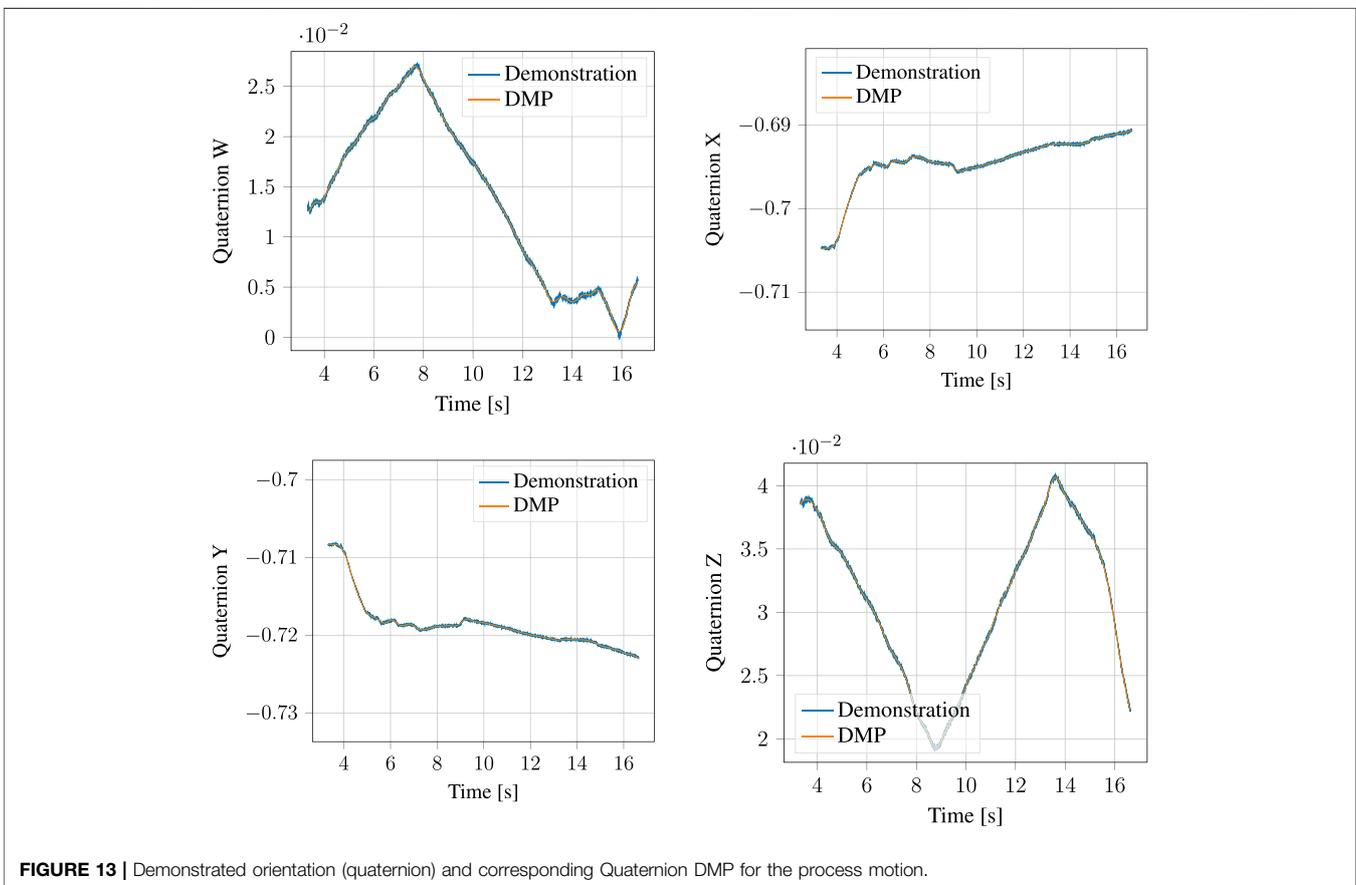


FIGURE 13 | Demonstrated orientation (quaternion) and corresponding Quaternion DMP for the process motion.

Section 5.1.2 shows a clear trend that our controller enables users to provide both more accurate and more efficient demonstrations of a path-constrained task when compared to Freedrive, as implemented on Universal Robots manipulators. This was verbally confirmed by the participants following the experiment. The subjects also reported that demonstrating was considerably more physically taxing using Freedrive, whereas our adaptive admittance scheme assisted in completing the task, making it easier to keep contact with the surface and to maintain a smooth continuous motion. As a full usability evaluation falls beyond the scope of this paper, backing this up with, e.g. the

NASA-TLX (Hart and Staveland, 1988) or SUS (Brooke et al., 1996) tests is left for future work.

In gluing and dispensing tasks, it is often imperative for an optimal process outcome that the feed-rate be kept constant throughout the surface of application. While approaches that are able to adjust the feed-rate of the gluing gun to compensate for changes in robot speed exist², by far the majority of applications use constant feed-rate gluing guns and therefore require constant robot movement speed. With our

²<https://aim-robotics.com/>.

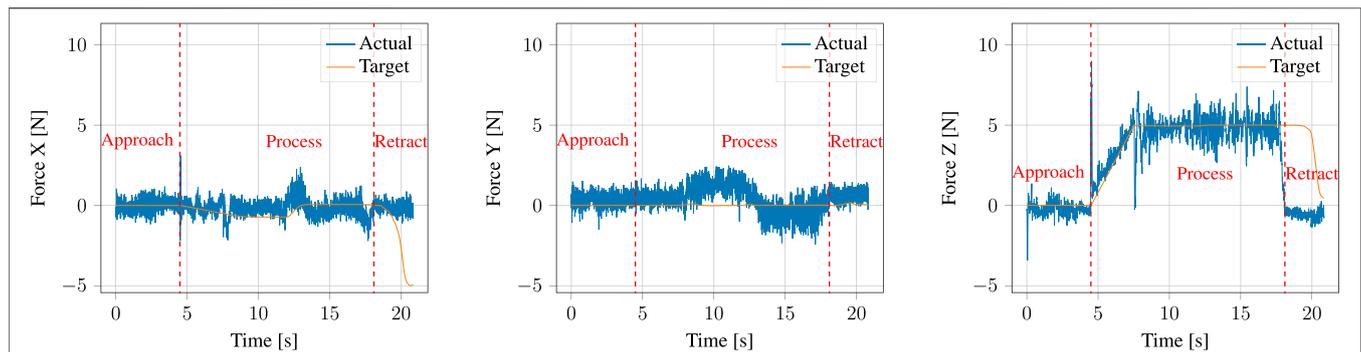


FIGURE 14 | Target forces input to the parallel position/force controller (labelled *target*) compared to the actual measured forces (labelled *actual*). The dotted vertical red line shows the point at which contact with the surface was detected, upon which the gains of the parallel position/force controller are adapted and the forces are ramped to the target process force magnitude.

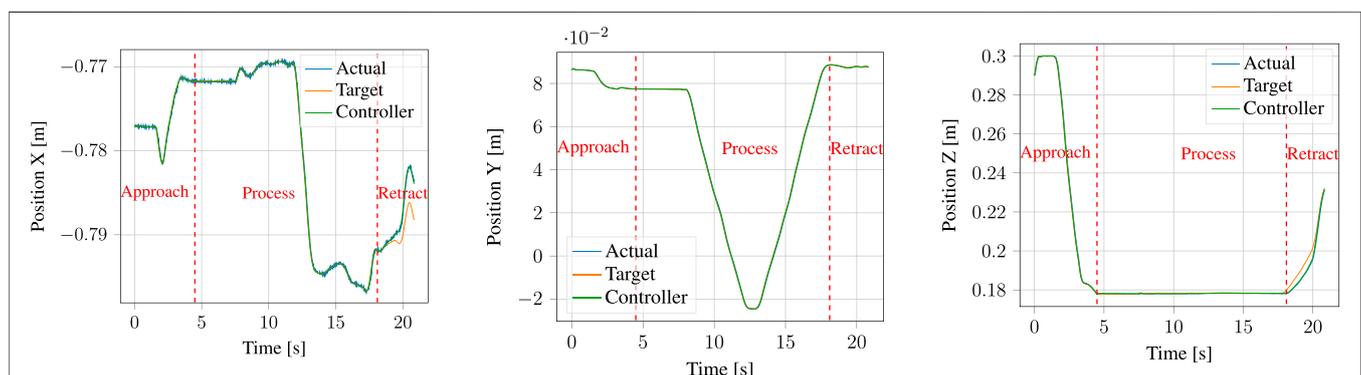


FIGURE 15 | Target positions input to the parallel position/force controller (labelled *target*) compared to the total controller output consisting of combined position and force contributions (labelled *controller output*) and to the actual measured positions (labelled *actual*).

approach, a skilled demonstrator can with some practice provide a near-constant speed demonstration, as is close to the case with the y -axis of **Figure 15**. However, it would be preferable to normalize execution speed in the encoding of the task to allow for imperfect demonstrations. Some approaches to DMP speed-scaling have been presented (Nemec et al., 2018a,b), typically through modulation of the time-constant, τ , with an additional term $\nu(x)$ that can be encoded as a mixture of RBFs. This method was considered for our system, but we were unable to achieve better performance than was demonstrated with it, as it did not result in constant speed. Alternative modulation schemes for DMP-speed will be investigated in future work.

The performance of the surface normal estimator is imperative for obtaining a good performance of the admittance controller used during the demonstration. It is of importance to both obtain fast convergence and little noise on the estimate. As seen in **Figure 9**, the convergence of the estimator might be improved; however, the parameters were chosen as a compromise between convergence rate and noise on the estimate. The surface normal estimation could potentially be improved by using a sliding model observer as presented in Hasan and Husain (2009). An analysis of

the stability of the interconnection between the adaptive controllers and the surface normal estimator was omitted, as the faster dynamics of the controller when compared to those of the gain adaptation according to the surface normal estimator preclude instability. Note that the performance of the surface normal estimator still enables tracking of curved surfaces, as shown in **Figure 8**, despite its dynamics being slower than those of the controller.

In applications relying on force control it is important to ensure that force limits are not surpassed. Especially when relying on programming by demonstration there is a high chance that the nominal behavior will not comply with force constraints. Thus, it is relevant to modify the position/force controller such that it guarantees compliance with force constraints. This can e.g. be ensured by set invariance control as shown in Polverini et al. (2017). High forces will likely occur in the transition between free and constrained motions if small positional uncertainties are present. Therefore, it could be beneficial to adapt the reference path according to the measured force as proposed in De Wit and Brogliato (1997) with an adaptive approach. This would be an alternative approach to the goal modulation proposed in **Section 4.4.1**.

As our system relies heavily on the performance of several parameter estimators, particularly the surface normal estimator, a possible improvement would be to allow for repetitive update of our parameter estimates, for instance through the use of Iterative Learning Control (ILC) (Wang et al., 2009). Such methods have already shown good performance in learning from demonstration systems for peg-in-hole tasks (Nemec et al., 2013; Abu-Dakka et al., 2015).

7 CONCLUSION

In this paper, we have presented a full system for learning from demonstration of gluing and dispensing tasks that encompasses the full pipeline of demonstration, encoding and execution.

During the demonstration, our system is designed to assist the user and provide a non-obtrusive kinesthetic teaching experience. By adapting the damping of an admittance controller according to a force-velocity law, we aid the user during acceleration or deceleration. Furthermore, when contact against the target surface is detected, a surface normal estimation algorithm is used to adapt the gain matrices of the controller to guarantee higher damping and preserve stability in the normal direction, while allowing lower damping in the other directions. Unlike previous approaches in the literature, we do not assume diagonal gain matrices. Rather, we allow the whole eigenstructure to change. This results in lower user forces and better adaptation to the user's intention.

During the encoding, we segment the task based on the initial contact into an approach phase and a process phase. Each of these are encoded as Dynamic Movement Primitives. Similarly, the surface normal estimate is encoded as Radial Basis Functions and synchronized with the process DMP phase system.

During the execution, a parallel position/force controller is used to meet both a position target and a force target, which is calculated based on a user-specified force magnitude applied on the encoded normal direction. The encoded normal is again used to vary the eigenstructure of the force controller gain matrices so as to apply force control in the contact direction and position control otherwise. Finally, we are able to reduce impact forces during the transition from approach to process by slowly modulating the goal of the approach DMP.

REFERENCES

- Abu-Dakka, F. J., Nemec, B., Jørgensen, J. A., Savarimuthu, T. R., Krüger, N., and Ude, A. (2015). Adaptation of Manipulation Skills in Physical Contact with the Environment to Reference Force Profiles. *Auton. Robot* 39, 199–217. doi:10.1007/s10514-015-9435-2
- Abu-Dakka, F. J., and Saveriano, M. (2020). Variable Impedance Control and Learning—A Review. *Front. Robotics AI* 7, 177. doi:10.3389/frobt.2020.590681
- Asada, H., and Izumi, H. (1989). Automatic Program Generation from Teaching Data for the Hybrid Control of Robots. *IEEE Trans. Robot. Automat.* 5, 166–173. doi:10.1109/70.88037
- Billard, A., Calinon, S., Dillmann, R., and Schaal, S. (2008). "Robot Programming by Demonstration," in *Springer Handbook of Robotics* (Berlin Heidelberg: Springer), 1371–1394. doi:10.1007/978-3-540-30301-5_60

We have evaluated and analyzed our system on a benchmark based on a real industrial PCB gluing application and verified the suitability of our proposed methods, showing robust execution performance capable of meeting both the kinematic and dynamic requirements of the task. We have also shown that the surface normal estimator works satisfactorily on curved geometries, and that the adaptive controller used for demonstration results in faster and more accurate executions compared to a standard controller implemented on a collaborative robot.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

ETHICS STATEMENT

Written informed consent was obtained from the individual(s) for the publication of any potentially identifiable images or data included in this article.

AUTHOR CONTRIBUTIONS

CS provided the initial suggestion for the system design and II, AK, and CS subsequently contributed to the conceptual development of the system. II implemented the core of the system, with AK and CS suggesting improvements. All three authors helped with the practical evaluation. II set up the initial article structure and first draft, and all three authors contributed to the writing and revision of the manuscript. II was mainly responsible for the figures.

FUNDING

The work leading to this publication is funded by the Innovation Fund Denmark (Innovationsfonden) as part of the Programming Ignition for Robotic Assembly Tasks (PIRAT) project, grant number 9069-00046B.

- Briat, C. (2014). *Linear Parameter-Varying and Time-Delay Systems: Analysis, Observation, Filtering & Control*. Berlin Heidelberg: Springer, 3, 5–7.
- Brooke, J. (1996). Sus-a Quick and Dirty Usability Scale. *Usability Eval. industry* 189, 4–7.
- Buchli, J., Stulp, F., Theodorou, E., and Schaal, S. (2010). Learning variable impedance control. *Int. J. Robot. Res.* 30, 820–833. doi:10.1177/0278364911402527
- Caccavale, F., Siciliano, B., and Villani, L. (1999). The Role of Euler Parameters in Robot Control. *Asian J. Control.* 1, 25–34. doi:10.1111/j.1934-6093.1999.tb00003.x
- Castelli, K., Zaki, A. M. A., Dmytryiev, Y., Carnevale, M., and Giberti, H. (2021). A Feasibility Study of a Robotic Approach for the Gluing Process in the Footwear Industry. *Robotics* 10, 6. doi:10.3390/robotics10010006
- Conkey, A., and Hermans, T. (2019). "Learning Task Constraints from Demonstration for Hybrid Force/position Control," in 2019 IEEE-RAS 19th International Conference on Humanoid Robots

- (Humanoids), Toronto, Canada, October 2019, 162–169. doi:10.1109/Humanoids43949.2019.9035013
- De Wit, C. C., and Brogliato, B. (1997). Direct Adaptive Impedance Control Including Transition Phases. *Automatica* 33, 643–649. doi:10.1016/s0005-1098(96)00190-2
- Deng, Z., Mi, J., Chen, Z., Einig, L., Zou, C., and Zhang, J. (2016). “Learning Human Compliant Behavior from Demonstration for Force-Based Robot Manipulation,” in 2016 IEEE International Conference on Robotics and Biomimetics (ROBIO), Qingdao, China, December 2016 (IEEE), 319–324. doi:10.1109/robio.2016.7866342
- Deniša, M., and Ude, A. (2015). Synthesis of New Dynamic Movement Primitives through Search in a Hierarchical Database of Example Movements. *Int. J. Adv. Robotic Syst.* 12, 137. doi:10.5772/61036
- Dimeas, F., and Aspragathos, N. (2014). “Fuzzy Learning Variable Admittance Control for Human-Robot Cooperation,” in 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, September 2014 (IEEE), 4770–4775. doi:10.1109/iros.2014.6943240
- Dimeas, F., and Aspragathos, N. (2015). “Reinforcement Learning of Variable Admittance Control for Human-Robot Co-manipulation,” in 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, September 2015 (IEEE), 1011–1016. doi:10.1109/iros.2015.7353494
- Ferraguti, F., Taligiani Landi, C., Sabatini, L., Bonfè, M., Fantuzzi, C., and Secchi, C. (2019). A Variable Admittance Control Strategy for Stable Physical Human-Robot Interaction. *Int. J. Robotics Res.* 38, 747–765. doi:10.1177/0278364919840415
- Ficuciello, F., Villani, L., and Siciliano, B. (2015). Variable Impedance Control of Redundant Manipulators for Intuitive Human-Robot Physical Interaction. *IEEE Trans. Robot.* 31, 850–863. doi:10.1109/TRO.2015.2430053
- Gams, A., Nemeč, B., Zlajpah, L., Wächter, M., Ijspeert, A., Asfour, T., et al. (2013). “Modulation of Motor Primitives Using Force Feedback: Interaction with the Environment and Bimanual Tasks,” in IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, November 2013 (IEEE), 5629–5635. doi:10.1109/iros.2013.6697172
- Gandy, A., and Lau, F. D.-H. (2013). Non-restarting Cumulative Sum Charts and Control of the False Discovery Rate. *Biometrika* 100, 261–268. doi:10.1093/biomet/ass066
- Gašpar, T., Deniša, M., Radanović, P., Ridge, B., Savarimuthu, T. R., Kramberger, A., et al. (2020). Smart Hardware Integration with Advanced Robot Programming Technologies for Efficient Reconfiguration of Robot Workcells. *Robotics and Computer-Integrated Manufacturing* 66, 101979. doi:10.1016/j.rcim.2020.101979
- Hart, S. G., and Staveland, L. E. (1988). “Development of Nasa-Tlx (Task Load Index): Results of Empirical and Theoretical Research,” in *Advances in Psychology* (Elsevier), 52, 139–183. doi:10.1016/s0166-4115(08)62386-9
- Hasan, S. M. N., and Husain, I. (2009). A Luenberger–Sliding Mode Observer for Online Parameter Estimation and Adaptation in High-Performance Induction Motor Drives. *IEEE Trans. Ind. Appl.* 45, 772–781. doi:10.1109/TIA.2009.2013602
- Ijspeert, A. J., Nakanishi, J., Hoffmann, H., Pastor, P., and Schaal, S. (2013). Dynamical Movement Primitives: Learning Attractor Models for Motor Behaviors. *Neural Comput.* 25, 328–373. doi:10.1162/neco_a_00393
- Karayiannidis, Y., Smith, C., Viña, F. E., and Kragic, D. (2014). “Online Contact point Estimation for Uncalibrated Tool Use,” in 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, May 31–June 5, 2014, 2488–2494. doi:10.1109/ICRA.2014.6907206
- Kober, J., Gienger, M., and Steil, J. J. (2015). “Learning Movement Primitives for Force Interaction Tasks,” in 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, May 2015, 3192–3199. doi:10.1109/ICRA.2015.7139639
- Kormushev, P., Calinon, S., and Caldwell, D. G. (2011). Imitation Learning of Positional and Force Skills Demonstrated via Kinesthetic Teaching and Haptic Input. *Adv. Robotics* 25, 581–603. doi:10.1163/016918611x558261
- Koropouli, V., Hirche, S., and Lee, D. (2012). “Learning and Generalizing Force Control Policies for Sculpting,” in IEEE International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal, October 2012 (Vilamoura: IEEE), 1493–1498. doi:10.1109/iros.2012.6385957
- Kronander, K., and Billard, A. (2014). Learning Compliant Manipulation through Kinesthetic and Tactile Human-Robot Interaction. *IEEE Trans. Haptics* 7, 367–380. doi:10.1109/toh.2013.54
- Kronander, K., and Billard, A. (2012). “Online Learning of Varying Stiffness through Physical Human-Robot Interaction,” in 2012 IEEE International Conference on Robotics and Automation, Saint Paul, MN, May 2012, 1842–1849. doi:10.1109/ICRA.2012.6224877
- Kumar, N., and Rani, M. (2021). A New Hybrid Force/position Control Approach for Time-Varying Constrained Reconfigurable Manipulators. *ISA Trans.* 110, 138–147. doi:10.1016/j.isatra.2020.10.046
- Lee, A. X., Gupta, A., Lu, H., Levine, S., and Abbeel, P. (2015). “Learning from Multiple Demonstrations Using Trajectory-Aware Non-rigid Registration with Applications to Deformable Object Manipulation,” in 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, September 2015, 5265–5272. doi:10.1109/IROS.2015.7354120
- Li, H.-Y., Dharmawan, A. G., Paranawithana, I., Yang, L., and Tan, U.-X. (2020). A Control Scheme for Physical Human-Robot Interaction Coupled with an Environment of Unknown Stiffness. *J. Intell. Robotic Syst.*, 1–18. doi:10.1007/s10846-020-01176-2
- Müller, F., Jäkel, J., Suchý, J., and Thomas, U. (2019). Stability of Nonlinear Time-Delay Systems Describing Human-Robot Interaction. *Ieee/asmе Trans. Mechatron.* 24, 2696–2705. doi:10.1109/TMECH.2019.2939907
- Müller, F., Janetzky, J., Behrnd, U., Jäkel, J., and Thomas, U. (2018). “User Force-dependent Variable Impedance Control in Human-Robot Interaction,” in 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE), Munich, Germany, August 2018, 1328–1335. doi:10.1109/COASE.2018.8560340
- Müller, M. (2007). “Dynamic Time Warping,” in *Information Retrieval for Music and Motion*. Berlin Heidelberg: Springer, 69–84.
- Nemeč, B., Abu-Dakka, F. J., Ridge, B., Ude, A., Jørgensen, J. A., Savarimuthu, T. R., et al. (2013). “Transfer of Assembly Operations to New Workpiece Poses by Adaptation to the Desired Force Profile,” in 2013 16th International Conference on Advanced Robotics (ICAR), Montevideo, Uruguay, November 2013 (IEEE), 1–7. doi:10.1109/icar.2013.6766568
- Nemeč, B., Likar, N., Gams, A., and Ude, A. (2018a). Human Robot Cooperation with Compliance Adaptation along the Motion Trajectory. *Auton. Robot* 42, 1023–1035. doi:10.1007/s10514-017-9676-3
- Nemeč, B., Žlajpah, L., Šlajpa, S., Piškur, J., and Ude, A. (2018b). “An Efficient Pbd Framework for Fast Deployment of Bi-manual Assembly Tasks,” in 2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids) (IEEE), 166–173. doi:10.1109/humanoids.2018.8625010
- Ortenzi, V., Stolkin, R., Kuo, J., and Mistry, M. (2017). Hybrid Motion/force Control: A Review. *Adv. Robotics* 31, 1102–1113. doi:10.1080/01691864.2017.1364168
- Pagano, S., Russo, R., and Savino, S. (2020). A Vision Guided Robotic System for Flexible Gluing Process in the Footwear Industry. *Robotics and Computer-Integrated Manufacturing* 65, 101965. doi:10.1016/j.rcim.2020.101965
- Pastor, P., Righetti, L., Kalakrishnan, M., and Schaal, S. (2011). “Online Movement Adaptation Based on Previous Sensor Experiences,” in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), San Francisco, September 2011 (San Francisco), 365–371. doi:10.1109/iros.2011.6095059
- Pérez-Ubeda, R., Zotovic-Stanisic, R., and Gutiérrez, S. C. (2020). Force Control Improvement in Collaborative Robots through Theory Analysis and Experimental Endorsement. *Appl. Sci.* 10, 4329. doi:10.3390/app10124329
- Peternel, L., Rozo, L., Caldwell, D., and Ajoudani, A. (2017). A Method for Derivation of Robot Task-Frame Control Authority from Repeated Sensory Observations. *IEEE Robot. Autom. Lett.* 2, 719–726. doi:10.1109/lra.2017.2651368
- Polverini, M. P., Nicolis, D., Zanchettin, A. M., and Rocco, P. (2017). “Robust Set Invariance for Implicit Robot Force Control in Presence of Contact Model Uncertainty,” in 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, Canada, September 2017, 6393–6399. doi:10.1109/IROS.2017.8206544
- Raibert, M. H., and Craig, J. J. (1981). Hybrid Position/force Control of Manipulators. *J. Dyn. Syst. Measur. Control Trans. ASME* 103, 126–133. doi:10.1115/1.3139652
- Roza, L., Jiménez, P., and Torras, C. (2013). A Robot Learning from Demonstration Framework to Perform Force-Based Manipulation Tasks. *Intel Serv. Robotics* 6, 33–51. doi:10.1007/s11370-012-0128-9
- Santos, L., and Cortesão, R. (2018). “Perceived Stiffness Estimation for Robot Force Control,” in 2018 IEEE International Conference on Robotics and Automation

- (ICRA), Brisbane, Australia, May 2018, 1667–1672. doi:10.1109/ICRA.2018.8460925
- Schaal, S. (2006). “Dynamic Movement Primitives-A Framework for Motor Control in Humans and Humanoid Robotics,” in *Adaptive Motion of Animals and Machines* (Tokyo: Springer), 261–280.
- Schindlbeck, C., and Haddadin, S. (2015). “Unified Passivity-Based Cartesian Force/impedance Control for Rigid and Flexible Joint Robots via Task-Energy Tanks,” in 2015 IEEE international conference on robotics and automation (ICRA), Seattle, May 2015 (IEEE), 440–447. doi:10.1109/icra.2015.7139036
- Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. (2009). *Robotics - Modelling, Planning and Control*. Springer.
- Sloth, C., and Iturrate, I. (2021). “Simultaneous Contact point and Surface normal Estimation during Soft finger Contact,” in Proceedings of the 20th International Conference on Advanced Robotics, Ljubljana, Slovenia, December 2021.
- Stramigioli, S. (2001). *Modeling and Ipc Control of Interactive Mechanical Systems—A Coordinate-free Approach*. New York: Springer-Verlag.
- Ude, A., Nemeč, B., Petrić, T., and Morimoto, J. (2014). “Orientation in Cartesian Space Dynamic Movement Primitives,” in 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, May 2014 (IEEE), 2997–3004. doi:10.1109/icra.2014.6907291
- Ureche, A. L. P., Umezawa, K., Nakamura, Y., and Billard, A. (2015). Task Parameterization Using Continuous Constraints Extracted from Human Demonstrations. *IEEE Trans. Robot.* 31, 1458–1471. doi:10.1109/tro.2015.2495003
- Wang, Y., Gao, F., and Doyle, F. J. (2009). Survey on Iterative Learning Control, Repetitive Control, and Run-To-Run Control. *J. Process Control.* 19, 1589–1600. doi:10.1016/j.jprocont.2009.09.006

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2021 Iturrate, Kramberger and Sloth. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.