Towards an Online Framework for Changing-Contact Robot Manipulation Tasks

Saif Sidhik[1], Mohan Sridharan[1] and Dirk Ruiken[2]

***Abstract*— We describe a framework for *changing-contact* robot manipulation tasks, which require the robot to make and break contacts with objects and surfaces. The discontinuous interaction dynamics of such tasks make it difficult to construct and use a single dynamics model or control strategy for such tasks. For any target motion trajectory, our framework incrementally improves its prediction of when contacts will occur. This prediction and a model relating approach velocity to impact force modify the velocity profile of the motion sequence such that it is C^∞ smooth, and help achieve a desired force on impact. We implement this framework by building on our hybrid force-motion variable impedance controller for continuous-contact tasks. We evaluate our framework in the illustrative context of a robot manipulator performing sliding tasks involving multiple contact changes with surfaces of different properties.**

I. Introduction

Consider a robot manipulator moving its end-effector along a desired pattern that involves making and breaking contacts, e.g., contact with the table's surface at "1" and with another object at "3" in Figure 1. The task's dynamics vary markedly before and after contact is made and broken, and based on the type of contact (e.g., surface, edge contact), surface friction, applied force, and other factors. We consider tasks involving changes in dynamics due to changes in the nature of contact as "*changing-contact*" tasks. Many core industrial assembly tasks, e.g., peg insertion and stacking, and human manipulation tasks, are changing-contact tasks whose discontinuous dynamics can result in poor transition-phase behavior or instability [1]. It is difficult to use a single dynamics model or control strategy for these tasks because the interaction dynamics are discontinuous when a contact is made or broken and continuous elsewhere [2].

Smooth motion along a desired trajectory can be achieved in a changing-contact manipulation task using an accurate analytical model of the transitions or a learned model that predicts the transition dynamics. Analytical models of the impact dynamics of a system of objects require comprehensive knowledge of the objects' physical and geometric attributes, and often impose assumptions not satisfied in practical domains [3], [4]. Methods that learn the attributes of the objects, build object classifiers based on these attributes, and/or learn sequences of parameters (e.g., joint angles) to achieve the desired trajectory, find it challenging to acquire sufficient examples of different objects, contacts, and attributes to learn generalizable models [5], [6].

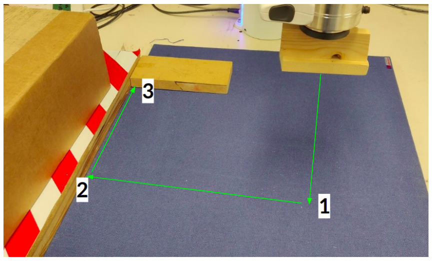


Fig. 1: A sliding task that involves making contact with the table's surface at "1" and with another object at "3".

Another approach is to use a *transition-phase* controller that lowers velocity and stiffness to reduce impact forces, vibration, and jerk on impact. Existing transition control strategies switch to a different controller once a contact is detected. This switch can cause substantial discontinuities in the interaction dynamics, damaging the robot or the objects [7], [8]. Instead, we seek to predict contacts and adapt the velocity and stiffness during the transition phase to minimize discontinuities, with the robot switching to a different controller after contact. To do so, we need to answer some key questions: (**Q1**) how best to predict contacts accurately? (**Q2**) when to activate the transition-phase controller? (**Q3**) how best to adapt the transition-phase controller's parameters to the task? and (**Q4**) what representation and strategy to use for reliable and efficient control with limited examples before, during, and after contact? Our framework builds on our prior work on a task-space, hybrid force-motion, variable impedance controller for continuous contact tasks [8], which partially addresses Q4 and simplifies the problem to enable the following contributions:

- A simple and efficient contact prediction method that incrementally improves its estimates.
- An adaptive strategy that uses predicted contacts to minimize time spent in the transition-phase; and
- A method that revises the transition phase velocity to achieve a C^∞ smooth velocity profile and a desired force on impact.

We evaluate our framework on a physical robot and in simulation, using the motivating example of sliding tasks that involve making and breaking contacts with objects and surfaces of different attributes. To thoroughly explore the control problems, we only consider sensor input from a force-torque sensor in the wrist. We review related work (Section II), and describe the framework (Sections III- V), results (Section VI), and conclusions (Section VII).

*This work was supported in part by Honda Research Institute Europe and the UK EPSRC grant EP/S032487/1.

[1] Intelligent Robotics Lab, School of Computer Science, University of Birmingham, UK [sxs1412,m.sridharan]@bham.ac.uk

[2] Honda Research Institute Europe GmbH, Offenbach am Main, Germany dirk.ruiken@honda-ri.de

II. Related Work

Control for changing-contact manipulation tasks can be achieved using analytical methods, learning methods, or transition-phase controllers.

Analytical methods that explore the relation between relative motion of two colliding objects and their impact dynamics typically formulate it as a linear complementarity problem (LCP) that considers the velocities and impulses [9], or the accelerations and forces [10], at contact points [3], [4], [11], [12]. Such a formulation and the associated methods can guarantee physical consistency between motion and impulses, but it is computationally expensive to solve the LCP at each time step, and difficult to provide the required prior knowledge of object attributes and/or accurate 3D object models in complex domains.

Methods developed to learn the physical attributes of objects, or to categorize objects based on these attributes, require the robot to perform the related task multiple times to obtain the training examples needed for building relevant models or for optimizing the models' parameters [6], [5]. Also, the learned models need to be retrained if the objects, tasks, or interaction dynamics change over time. Since changing-contact tasks are piece-wise continuous, some methods build probabilistic functions offline from collected data for predicting mode/contact changes [13], [14]. These methods assume the environment remains unchanged between training and testing, and require many training examples (including synthetic data) [14].

Methods that use a transition-phase controller for changing contact manipulation tasks focus on minimizing the discontinuities in the dynamics, i.e., on reducing the forces, vibration, and jerk on impact [7], [8]. However, many of these methods switch to a different controller only after a contact is detected, which can result in significant discontinuities when the switch is made, along with loss of energy and damage to the robot or the domain objects.

Since our approach includes a transition-phase controller, for smooth motion that that does not transition abruptly after contact detection, the transition velocity profile has to be continuous and the motion has to be at least C^4 smooth. Methods have been developed for kinematic time-optimal motion using trapezoidal velocity profiles [15], C^4-smooth trajectories using segments [16], and for minimum-jerk motion profiles [17], [18], [19]. These methods can be computationally expensive, particularly for paths with many points in each trajectory segment. Our approach instead focuses on smooth motion over any given trajectory by accurately predicting contact, and modifying the velocity using a C^∞ smooth profile to transition to (from) a transition-phase controller before (after) contact. Our main contribution is a framework that encodes representational choice to learn, in a few trials, to: (a) predict contact changes in a changing-contact manipulation task; and (b) adapt its control strategy such that the impact forces are reduced while deviating minimally from the target trajectory.

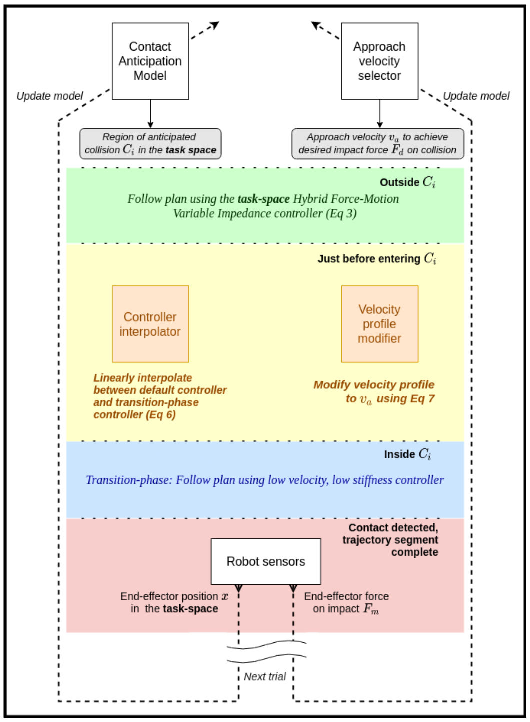

Contact Anticipation Model
Approach velocity selector
Update model
Update model
Region of anticipated collision C_i in the **task space**
Approach velocity v_a to achieve desired impact force F_d on collision
Outside C_i
Follow plan using the **task-space** Hybrid Force-Motion Variable Impedance controller (Eq 3)
Just before entering C_i
Controller interpolator
Velocity profile modifier
Linearly interpolate between default controller and transition-phase controller (Eq 6)
Modify velocity profile to v_a using Eq 7
Inside C_i
Transition-phase: Follow plan using low velocity, low stiffness controller
Contact detected, trajectory segment complete
Robot sensors
End-effector position x in the **task-space**
End-effector force on impact F_m
Next trial

Fig. 2: Overview of framework for smooth control of changing-contact manipulation tasks.

III. Framework Overview

Figure 2 presents an overview of our framework. The inputs are the desired motion trajectory, the force-torque sensor measurements, and the end-effector position. The default controller is the hybrid force-motion variable impedance controller that we developed for continuous-contact tasks [8]. This includes an incrementally learned forward (predictive) model of end-effector measurements; the error between the predicted and actual measurements automatically revise stiffness values in control laws to determine the control signal. We showed that operating in task (i.e., Cartesian) space allows this controller to use suitable abstractions to learn accurate forward models from very few examples, provide compliance along specific directions, and accurately track the desired trajectory, thus partially addressing Q4 in Section I. The framework in this paper builds on this default controller's representation, enabling a task-space contact anticipation model that incrementally updates its contact prediction using a Kalman filter (Q1). These predictions are used to minimize the time spent in the transition phase (Q2), and the velocity to be used in the transition phase is set adaptively to achieve a C^∞ smooth velocity profile and a desired impact force (Q3). Once the transition is completed at a suitable velocity and stiffness to minimize discontinuities, the robot moves to using another version of default controller and revises the parameter values suitably. We begin with a description of the contact prediction method.

IV. Contact Prediction

Anticipating contacts by predicting impact forces or time to collision is challenging because these parameters are influenced by robot dynamics and the controller's parameters, e.g., reducing the robot's velocity or the controller's stiffness reduces the impact force and increases time to contact. It is more reliable to predict *static* contact parameters such as end-effector position during impact and direction of contact force, which do not change significantly for repetitions of the task as long as we can make the reasonable assumption that the trajectory and the environmental attributes do not change significantly between repetitions.

The robot's belief about the position of each expected contact point in the assigned motion trajectory is modeled as a multivariate Gaussian, with the covariance ellipsoid denoting the uncertainty and the "region of anticipated contact" $\mathcal{C}$. Since the controller is in the task space, each contact location's representation is compact and is updated over very few trials of the task using a Kalman filter with the state update equation: $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + w$, where $\mathbf{x}$ is the contact position, $\mathbf{A}$ is defines the dynamics of motion of the contact point without the robot acting on it (I for positively activated objects, which is the case in this paper), $\mathbf{B}$ is the control matrix capturing the effect of the robot's action $\mathbf{u}$ on contact position, and w is the Gaussian model of the uncertainty in contact location. The sensor model uses the end-effector pose (given by forward kinematics from joint positions) as measurement when a contact is detected; noise in the sensor model depends on the joint encoder noise and forward kinematics. The corrected estimate of the contact point results in a reduced covariance ellipsoid for subsequent trials. The update equations are as follows:

$$\hat{\mu}_{k|k-1} = \mathbf{A}_{k-1}\hat{\mu}_{k-1|k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1} \tag{1a}$$

$$\mathbf{\Sigma}_{k|k-1} = \mathbf{A}_{k-1}\mathbf{\Sigma}_{k-1|k-1}\mathbf{A}_{k-1}^{\mathbf{T}} + \mathbf{Q}_{k-1} \tag{1b}$$

$$\mathbf{v}_k = \mathbf{y}_k - \mathbf{H}_k\hat{\mu}_{k|k-1} \tag{1c}$$

$$\mathbf{S}_k = \mathbf{H}_k\mathbf{\Sigma}_{k|k-1}\mathbf{H}_k^{\mathbf{T}} + \mathbf{R}_k \tag{1d}$$

$$\mathbf{K}_k = \mathbf{\Sigma}_{k|k-1}\mathbf{H}_k^{\mathbf{T}}\mathbf{S}_k^{-1} \tag{1e}$$

$$\hat{\mu}_{k|k} = \hat{\mu}_{k|k-1} + \mathbf{K}_k\mathbf{v}_k \tag{1f}$$

$$\mathbf{\Sigma}_{k|k} = \mathbf{\Sigma}_{k|k-1} - \mathbf{K}_k\mathbf{S}_k\mathbf{K}_k^{\mathbf{T}} \tag{1g}$$

where $\hat{\mu}_{i|i-1}$ and $\mathbf{\Sigma}_{i|i-1}$ are the predicted mean and covariance at step i, $\hat{\mu}_{i|i}$ and $\mathbf{\Sigma}_{i|i}$ are the corrected mean and covariance based on measurement $\mathbf{y}_i$ (of position on contact) at step i, $\mathbf{K}$ is the Kalman gain, and $\mathbf{Q}$ and $\mathbf{R}$ are noise matrices. Our representational choices enable us to develop such a linear anticipation model to estimate contact locations accurately from very few (noisy) repetitions of the task. Although this representation supports contact with movable objects, we assume in this paper that the end-effector only makes contact with stationary objects ($\mathbf{A} = \mathbf{I}$, $\mathbf{B} = \mathbf{0}$). Also, $\mathbf{H} = \mathbf{I}$ since state and measurements are in the same space. These simplifications result in Gaussian updates using the noisy measurements based on the robot's kinematics model (since sensor input is from an FT sensor) each time the robot experiences a contact change.

V. Controller Formulation

For any given task, the desired motion trajectory P is provided as a sequence of mappings from time to the corresponding end-effector pose and force (for force control); it is obtained through a single demonstration of the task by a human moving the robot manipulator. Defining the controller in the Cartesian-space provides an intuitive trajectory description. P is made up of segments, each of which is assumed to be smooth, continuous, and jerk-free; transition between segments is accompanied by a change in the direction of (force, motion) control, and P does not account for the collisions (i.e., contact points).

Our framework's default controller builds on the standard variable impedance control equation [20]:

$$\boldsymbol{h}_c = \mathbf{\Lambda}(\boldsymbol{q})\ddot{\boldsymbol{x}}_d + \mathbf{\Gamma}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{x}}_d + \boldsymbol{\eta}(\boldsymbol{q}) + \mathbf{K_p}\Delta\boldsymbol{x} + \mathbf{K_d}\Delta\dot{\boldsymbol{x}} \tag{2}$$

where $\boldsymbol{h}_c$ is the task-space control command, $\mathbf{\Lambda}(\boldsymbol{q}) = (\boldsymbol{J}\boldsymbol{M}(\boldsymbol{q})^{-1}\boldsymbol{J}^T)^{-1}$ is the 6×6 inertia matrix, $\mathbf{\Gamma}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{J}^{-T}\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\boldsymbol{J}^{-1} - \mathbf{\Lambda}(\boldsymbol{q})\dot{\boldsymbol{J}}\boldsymbol{J}^{-1}$ denotes the compensation wrenches including centrifugal and Coriolis effect, and $\boldsymbol{\eta}(\boldsymbol{q}) = \boldsymbol{J}^{-T}\boldsymbol{g}(\boldsymbol{q})$ is the gravitational wrench. $\boldsymbol{M}(\boldsymbol{q}), \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})$, and $\boldsymbol{g}(\boldsymbol{q})$ are the equivalent values defined in the joint space of the robot; $\Delta\boldsymbol{x}$ is the error in end-effector pose with respect to a desired pose $\boldsymbol{x}_d$; and $\mathbf{K_p}$ and $\mathbf{K_d}$ are 6×6 symmetric positive-definite matrices of desired impedance stiffness and damping. The joint-space control torque is computed as $u = \boldsymbol{J}^T h_c$.

In the absence of the external wrench $\boldsymbol{h}_e$, the control law provides asymptotic stability with equilibrium state $\dot{\boldsymbol{x}}_e = 0, \Delta\boldsymbol{x} = 0$ for a closed-loop system. With a non-zero $\boldsymbol{h}_e$, a non-null $\Delta\boldsymbol{x}$ will be present at equilibrium. For a fixed or non-stationary target $\boldsymbol{x}_d$, if the external force $\boldsymbol{h}_e$ is due to non-fixed resistance (e.g., friction when sliding on a surface), forces against the direction of motion can be canceled with a feed-forward term $\boldsymbol{h}_{ff}$ in the control law:

$$\boldsymbol{h}_c = \mathbf{\Lambda}(\boldsymbol{q})\ddot{\boldsymbol{x}}_d + \mathbf{\Gamma}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{x}}_d + \boldsymbol{\eta}(\boldsymbol{q}) + \mathbf{K_p}\Delta\boldsymbol{x} + \mathbf{K_d}\Delta\dot{\boldsymbol{x}} + \boldsymbol{h}_{ff} \tag{3}$$

When no contact change is expected, $\boldsymbol{h}_{ff}$, $\mathbf{K_p}$, and $\mathbf{K_d}$ are revised based on the difference between the predicted and observed values of forces and torques at the end-effector. The predictions are based on a task-specific (feed)forward model that is revised incrementally during task execution [8]. We build on this default controller to answer questions Q2-Q4.

A. Transition Controller Parameters

We first describe the transition controller and its parameters. Since the permitted impact force may differ based on the task, e.g., large forces can damage delicate objects, we imposed a limit on the maximum allowed impact force. Also, experimental analysis indicated that reducing the controller stiffness helps reduce the jerk in motion after impact by providing compliance, but has no significant effect on impact forces because the error and stiffness term in the feedback control loop come into effect only after contact is made. A safe controller should thus have lower stiffness for reducing vibrations. In addition, the approach velocity was observed to be directly proportional to the impact force, especially when the robot registers a contact while moving in free space. The design of our controller and the representational choices allow the use of a simple method (linear regression) to model the relationship between impact force and approach velocity between a pair of objects. This model is then used to compute the approach velocity for a desired impact force.

Since the robot may not initially have a model of the relationships discussed above, it starts with a safe low velocity during the first trial of any given task and target force on impact. It then uses the difference between the target and measured force on impact to revise the approach velocity for the next iteration of the task:

$$\Delta v_a = \beta(F_d - F_m) \tag{4}$$

where Δv_a is the change in approach velocity, F_d is the desired impact force along motion direction, F_m is the measured impact force, and β is a learning rate that is ideally less than or equal to the slope of the plot relating impact force to approach velocity. Over time, this method enables the robot to learn a task-specific approach velocity for a desired impact force; the learned linear model can also be reused for other target impact forces. Next, we consider when to start using the transition-phase controller.

B. Switching to Transition-Phase Controller

Recall that a lower stiffness in the transition phase can reduce vibrations on contact, and a lower velocity reduces the impact forces. Since any such strategy will cause the robot to deviate from the desired trajectory, the robot should ideally switch to this control phase just before the contact is made, and switch out of it immediately after stable contact is established. Since this is not possible in practice, it is safer to switch to this control mode when it enters a region in the task space where the contact is highly likely to occur, and switch out of it once stable contact is achieved.

As stated earlier, we use the covariance of the multivariate Gaussian estimating the contact location to define the region of anticipated contact ($\mathcal{C}$) in the task-space. Activating the transition-phase controller just before or after it enters $\mathcal{C}$ ensures that the transition-phase is only active when a contact is anticipated. The part of the target motion trajectory P within $\mathcal{C}$ can be found by checking if the points in $P(t)$ satisfy the relation:

$$(\mathbf{P}(t) - \mu)^T \mathbf{\Sigma}^{-1} (\mathbf{P}(t) - \mu) \leq \lambda \tag{5}$$

where μ is the mean of the Gaussian predicting contact position, $\mathbf{\Sigma}$ is the covariance, λ is a scaling factor modeled as the chi-squared percent point function of the desired confidence value. The first point in P to satisfy this condition is the boundary p_c of $\mathcal{C}$. When the robot does a task for the first time, the position uncertainty and volume of $\mathcal{C}$ are large, and the robot switches to the transition phase controller earlier than actual impact. Over time, as the covariance ellipsoid shrinks, the robot switches to the transition controller when it is about to make impact.

C. Smooth Transition between Controllers

Since the desired impact force is primarily achieved by revising the approach velocity, the transition-phase controller is set (by the designer) to use lower fixed controller gains $(\mathbf{K_p}^*, \mathbf{K_d}^*)$ as the robot moves at a lower velocity, for reducing the negative effects of collision. To avoid discontinuities, the robot needs to smoothly transition from a normal (pre-contact) controller with output $\mathbf{u_1}$ to the transition-phase controller with output $\mathbf{u_2}$. We use linear interpolation of $\mathbf{u_1}$ and $\mathbf{u_2}$ over a time window $[0, T]$ such that the transition is completed by the time the robot reaches p_c:

$$\mathbf{u} = (1 - \alpha)\mathbf{u_1} + \alpha\mathbf{u_2}; \qquad \alpha = t/T \qquad t \in [0, T] \tag{6}$$

where T is the desired duration of the transition between the controllers. As long as the outputs from the two controllers ($\mathbf{u_1}$ and $\mathbf{u_2}$) are individually smooth, the combined output will also be smooth. In this work, controllers use the task-space representation described earlier, with $\mathbf{u_2}$ being the output of the fixed, low-gain, transition-phase controller as the arm approaches the contact point. A similar approach is used to smoothly transition from the transition-phase controller to a normal controller after contact is made.

D. Modifying Velocity Profile over Target Trajectory

Recall that transition-phase controllers use a lower velocity than that used in the original kinematic sequence P to reduce the force on impact. Also, the switch to this controller will take place at different points in the trajectory as the region $\mathcal{C}$ is revised over time. The trajectory's timeline thus has to be modified to account for the modified velocity profile. To achieve this objective, we enable the robot to create a new velocity profile and time-mapping by building on the existing work on trapezoidal velocity profiles; it can be viewed as the *lift-off* or *set-down* phase of a trapezoidal profile. Our formulation results in motion that is smooth and continuous at all orders, i.e., is C^∞ smooth.

Without loss of generality, assume that P is along one dimension with velocity v_1. Assuming that transition starts at time t_1 with v_1 and has to be completed at t_2 with velocity v_2 as the robot crosses boundary point p_c of $\mathcal{C}$, the velocity profile is defined as:

$$v(\tau) = \begin{cases} v_1 + \frac{(v_2 - v_1)e^{-1/\tau}}{e^{-1/\tau} + e^{-1/(1-\tau)}} & \text{if } 0 < \tau < 1, \\ v_1 & \text{if } \tau \leq 0 \\ v_2 & \text{if } \tau \geq 1 \end{cases} \tag{7}$$

where $\tau = t/T = t/(t_2 - t_1)$. For $\tau \in (0, 1)$, $e^{-1/\tau}$ has continuous derivatives at all orders at every point τ on the real line. Since $v(\tau)$ has a strictly positive denominator for all points in its domain and velocity limits are enforced $\forall \tau \notin [0, 1]$, this profile provides a smooth transition from v_1 to v_2 over $[t_1, t_2]$ and $v(\tau)$ is continuous despite its piecewise definition. Acceleration and jerk are computed as first- and second-order derivatives of $v(\tau)$ with respect to τ, and position trajectory is obtained by integrating the profile; all motion derivatives are continuous—see Figure 3.

The timeline of the new velocity profile $v(\tau)$ for any given contact can be used to modify the corresponding P such that the velocity transition is completed as the robot reaches p_c. As the duration for velocity transition (Equation 7) and

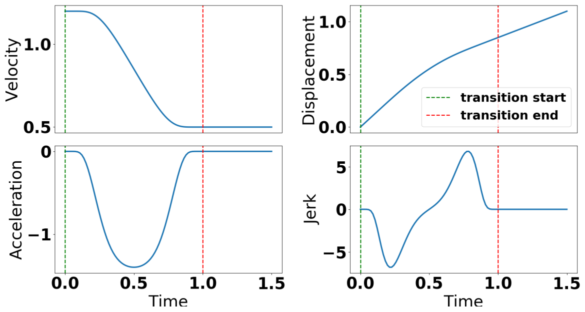


Fig. 3: Velocity plots with matched position, acceleration, and jerk plots. Velocity varies from 1.2 to 0.5 in unit time.

controller transition (Equation 6) is the same, the robot will become compliant and slow down just before it enters $\mathcal{C}$.

Note that only position is adapted using the C^∞-smooth profile; orientation is modified using SLERP, a linear interpolation of points in the spherical space of quaternions. This approach may cause position-orientation mismatch in manipulation tasks involving significant orientation changes in the transition regions.

VI. Experimental Analysis

We experimentally evaluated the following hypotheses:

H1: Our contact prediction approach incrementally improves the estimate of each contact's position over time, reducing task-completion delay and trajectory tracking error;

H2: The learned linear relationship between approach velocity and impact force provides an accurate estimate of the approach velocity for a desired impact force; and

H3: The overall framework produces smooth motion dynamics (i.e., velocity, acceleration etc) for manipulation tasks with multiple contact changes.

For experiments, we used a seven degrees of freedom (DoF) Franka Emika Panda robot operating on a tabletop (Figure 1) and its simulated version in PyBullet. Due to space constraints, we report results on the physical robot below; a video of the physical robot and simulation results are in the supplementary material. The performance measures include accuracy (e.g., position tracking, impact force), task completion time, and the time spent in the transition-phase.

A. Contact Anticipation

To evaluate the ability to incrementally improve the estimate of contact position (**H1**), we used a task-space trajectory that required the robot to approach a (static) table from above, move back up without making contact with the table, and move down and make contact with the table, resulting in a zig-zag trajectory along the z-axis. As described earlier, the robot was expected to move with a lower velocity when approaching a contact point, but spend as little time as possible in this low-velocity, low-stiffness transition phase to reduce tracking error and delay in task completion.

An initial estimate of each contact position (based on target trajectory) was provided manually to simulate input from an external planner or vision system. Each initial estimate had a large covariance (0.175 along each dimension, with distance measured in meters) to simulate the uncertainty associated with a visual sensor or planner. Due to the large covariance, the region of anticipated contact ($\mathcal{C}$) overlapped with points in the first 'valley' of the target (zig-zag) trajectory although there was no actual contact with the table's surface. We expected the robot to obtain an improved estimate of $\mathcal{C}$ over time and not switch to the transition-phase controller in the first valley; the switch was only expected when the robot approached the table the second time. Given the focus on contact prediction, we empirically chose safe values for the transition-phase control parameters (i.e., approach velocity and stiffness).

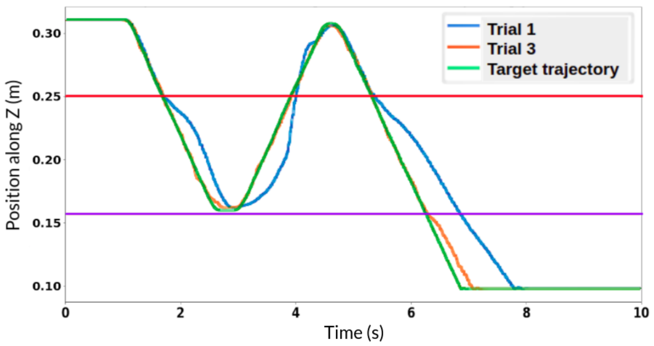


Fig. 4: Position of the end-effector (EE) during trial-1 and trial-3 of the contact prediction experiment. The red horizontal line is the edge of the covariance ellipsoid in trial 1; the violet line is the ellipsoid boundary in trial 3. Updated covariance in trial 3 enables the robot to avoid going to the transition-phase in the first dip of zig-zag trajectory and reduce the tracking error.

We observed a significant reduction in covariance, e.g., from 0.175 to 0.07 in just three successive trials in an experiment, as summarized in Figure 4, which enabled the robot to avoid going to the transition-phase in the first dip in the trajectory. Also, the average Euclidean tracking error (per time step) in the position of the end-effector (EE) reduced from $1.3\,cm$ in the first trial to $0.16\,cm$ in the third trial, and the task completion time reduced from $7.9\,s$ in the first trial to $7.2\,s$ in the third trial; the expected (ground truth) motion duration is $7\,s$. Similar results were obtained with other target trajectories, indicating support for **H1**, i.e., that the uncertainty in the contact position is reduced quickly, which reduced delays in task completion as well as errors in trajectory tracking. These results also indicate that using the transition-phase controller only when it is required reduces the deviation from the desired motion trajectory.

B. Approach Velocity and Impact Force

To test the relation between approach velocity and impact force on contact, the robot was given a target motion trajectory that required it to move in free space and make contact with the table; this is also shown in the supplementary video. The task was repeated with different velocities ranging from $0.02\,m/s$ to $0.16\,m/s$ in steps of 0.02, each repeated four times, and we measured the corresponding force on contact. We observed that a line whose parameters were estimated

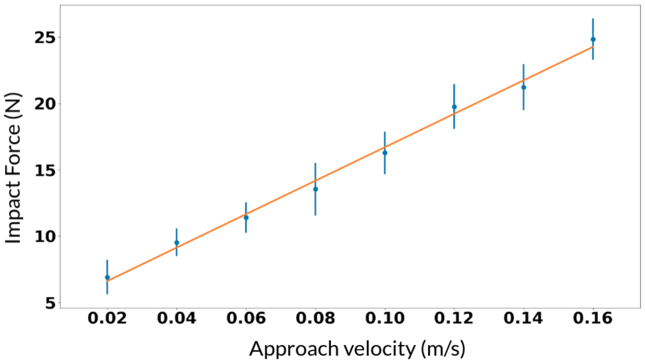


Fig. 5: Approach velocity vs force on impact. Orange line denotes the estimated linear relationship.

Target Force (N)	Estimated reqd velocity (m/s)	Measured force (N)	Error force (N)
10	0.047	7.4	2.6
12	0.063	15.1	3.1
15	0.086	15.3	0.3
18	0.11	16.7	1.3

TABLE I: Errors in contact force with the learned function specifying approach velocity as a linear function of the impact force; errors were higher at lower values of target force due to sensor noise.

by linear regression provided a reasonably good fit for the relationship between end-effector approach velocity and the end-effector force along the direction of motion, as shown in Figure 5. The variance in the fit can be attributed largely to the noise in the force-torque sensor, which can be large during discontinuities such as collisions.

Given such a learned relationship, the robot was asked to perform the same target trajectory (as above), but it had to now choose its approach velocity so as to achieve a desired impact force on contact. The measured contact force was compared with the desired impact force. Table I summarizes results for four trials for four of the 11 target force values we tested ($10-20$N at 1N increments). We observed that the robot was able to compute an approach velocity that resulted in an impact force similar to the desired value, with an error of $\sim 3\,$N. These errors were more pronounced at lower values of the target impact force, which can be attributed to sensor noise, i.e., the learned model was limited by the accuracy, sensitivity, and resolution of the force-torque sensor, joint encoders, and the robot's forward kinematics model.

Equation 4 was then used to incrementally update the approach velocity of the robot without providing the learned linear model. The initial value of the approach velocity was set to $0.1\,m/s$, the target impact force (F_d) was 10 N, and $\beta = 0.003$. Figure 6 shows the evolution of approach velocity over 10 successive trials. We observed that the approach velocity was $\approx 0.045\,m/s$ in the fifth trial, after which the noise in the force torque sensor measurements at impact made it difficult to converge to a single value of approach velocity. The error between the measured force and desired force reduced from $8.5\,N$ to $0.2\,N$ at the end of 10 trials. Similar results were obtained for other values of initial approach velocity and target impact force, indicating that in the absence of the learned linear model, it takes a greater number of trials to converge to a suitable approach velocity for a target impact force. These results thus support **H2**.

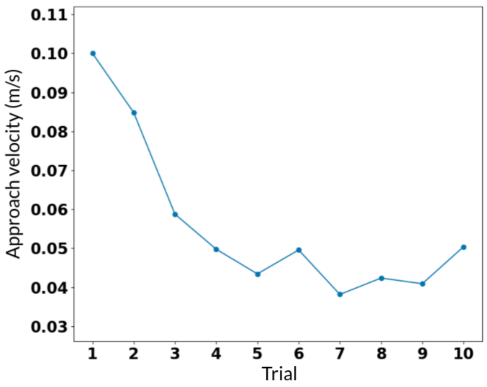
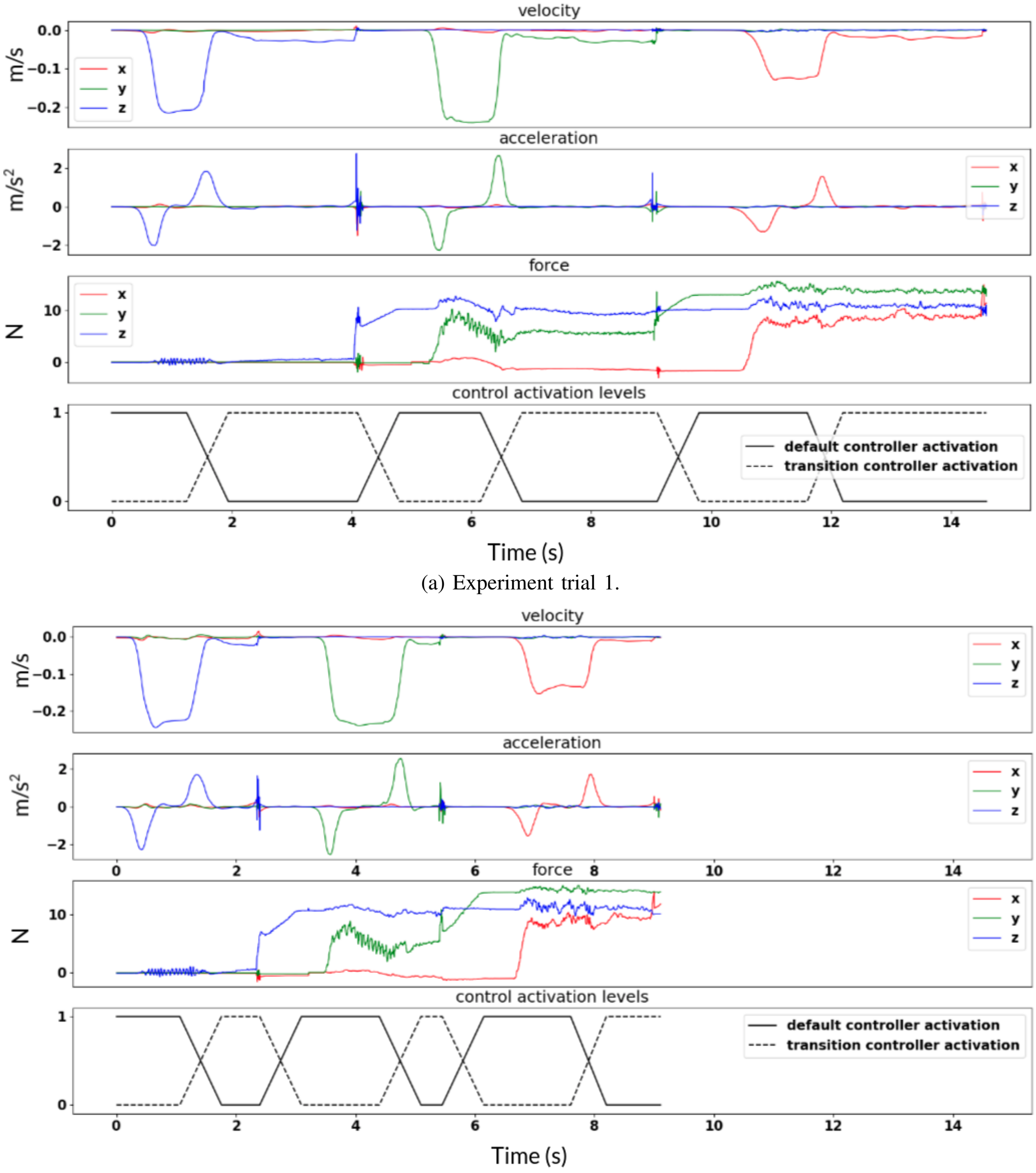


Fig. 6: Approach velocity over a sequence of trials to achieve a target impact force of 10 N.

C. *Smoothness of Motion*

The motion profiles (e.g., velocity, acceleration profiles) of a changing-contact manipulation task are expected to have large spikes in the absence of our framework that predicts the contact locations and adapts the velocity and stiffness during the approach to a contact position. This hypothesis (**H3**) was tested in a simulated environment. The corresponding results indicated that our framework significantly reduced the spikes in the overall motion profile of the robot in a changing-contact manipulation task, while also ensuring safe interaction during contact changes. The results from these simulation experiments are available in the supplementary material and are not described here due to space limitations.

To evaluate the overall framework and the resulting dynamics on a physical robot, the robot (with a wooden block attached to end-effector) was asked to move vertically down to the table (contact 1), slide along y-axis (the table's surface) to a wall (contact 2), and slide along the wall (while in contact with the table's surface) to another obstacle (contact 3), as shown in Figure 1. The robot was provided significantly incorrect initial guesses of the contact positions with substantial noise (see Table II). The robot had to repeat the task while reducing the deviation from the given motion pattern by improving its estimate of the contact positions. The robot also had to modify its approach velocity from the initial value of $0.05\,m/s$ to produce a desired impact force of 8 N. Since each contact in the task is in the presence of different environment dynamics (e.g., motion in free space, motion against surface friction), the velocity required to attain the desired impact force was expected to be different. The robot also had to incrementally update its approach velocity for each contact using gradient descent till the desired velocity for that environment was achieved. Furthermore, the robot had to perform all the trials with smooth overall motion dynamics with minimum spikes in the velocity or acceleration profiles.



(a) Experiment trial 1.

(b) Experiment trial 5.

Fig. 7: Velocity, acceleration, force, and controller activation levels in: (a) experimental trial 1; (b) experimental trial 5. Use of our framework reduces uncertainty in estimates of contact positions, reduces the time spent using the transition-phase controller, and reduces discontinuities.

Figure 7a shows the velocity, acceleration, and EE force in the first trial, and Figure 7b shows these values after five trials. The results in these figures and in Table II show that the uncertainty in the estimate of the contact positions is reduced, as indicated by a significant reduction in the size of the covariance ellipsoids, and the robot spends significantly less time using the transition-phase controller and the associated lower velocity. The last plot in Figure 7a and Figure 7b show the activation of the default controller and the transition-phase controller. The overall task was completed in $9.2\,s$ in the fifth trial as opposed to $14.4\,s$ in the first trial. The covariance ellipsoids converged in the first three trials of the task, but the task was repeated to evaluate the ability to compute and set the approach velocity for different transition-phase controllers.

With our framework, the robot converged to a suitable approach velocity for the first contact (from motion in free space) in five iterations. It was, however, difficult for the robot to adjust its approach velocities for contacts 2 and 3, which required the robot to use force control along one and two directions (respectively). Contact 3 was particularly challenging because it involved sliding along two different

Prediction Error (m)	Initial	Final (trial 5)
Contact 1 (Z-axis)	0.12 ± 0.3	0.016 ± 0.039
Contact 2 (Y-axis)	0.09 ± 0.2	0.011 ± 0.04
Contact 3 (X-axis)	0.1 ± 0.2	0.018 ± 0.036

TABLE II: Error in the estimated contact location along the most significant axis for the contact (in parenthesis) in the first and fifth trials of the task in Figure 1. The value along the diagonal of the corresponding covariance matrix is shown as the standard deviation (± term).

surfaces, resulting in very noisy readings from the force-torque sensor due to the different values of frictional resistance offered by the two surfaces. Since the impact force was along the same direction as friction, it was more difficult to isolate the impact force from the force due to surface friction.

VII. Discussion and Future Work

This paper described a framework for addressing the discontinuities in changing-contact manipulation tasks. The framework introduces a transition-phase controller in a hybrid force-motion variable impedance controller for continuous-contact tasks. Our representational choices enable us to simplify and address the associated challenges reliably and efficiently. Specifically, a Kalman filter formulation is adapted to incrementally improve the estimates of the contact positions. These estimates are used to minimize the time spent in the transition phase (with lower velocity and stiffness), and the velocity profile is modified automatically to achieve smooth motion and a desired impact force.

The framework opens up many directions of further research. First, we only focused on collisions due to translational motion, and did not address collisions due to rotations of the end-effector. This could be addressed by defining a region of anticipated collision in $\mathcal{SO}(3)$. Second, we observed that updating approach velocity for collisions when the robot is already in contact with another surface is more complicated. This is because of the difficulty in differentiating the sensor readings obtained due to reactive forces from the existing contact and the sensor readings obtained due to the impact force generated by the collision with another object. One possible way to address this issue is to learn a better forward model for the contact mode such that it can accurately predict the forces due to the first contact. Third, we only modified the velocity profile to achieve the desired smooth motion, and future work will explore the relationship between stiffness values and the impact forces. Initial experiments indicate that this is a challenging problem, as summarized in the supplementary material. Moreover, reducing the stiffness during approach (to a contact position) makes the motion more sensitive to inertia, e.g., the velocity drops almost to zero before settling on the target approach velocity at time 1.5s in Figure 7a. This behavior is due to the lag in tracking the target trajectory and the uncompensated end-effector mass, which are due to the lower value of the stiffness used as the robot approaches a contact position. The long-term objective of this research is to achieve smooth and reliable motion in different changing-contact manipulation tasks.

References

[1] R. P. Paul, "Problems and research issues associated with the hybrid control of force and displacement," in *IEEE International Conference on Robotics and Automation*, 1987, pp. 1966–1971.

[2] B. Brogliato and P. Orhant, "On the transition phase in robotics: impact models, dynamics and control," in *International Conference on Robotics and Automation*, 1994.

[3] K. Hunt and E. Crossley, "Coefficient of restitution interpreted as damping in vibroimpact," *Journal of Applied Mechanics*, 1975. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01333795

[4] A. Nakashima, Y. Ooka, and Y. Hayakawa, "Contact transition modelling on planar manipulation system with lugre friction model," in *International Workshop on Robot Motion and Control*, 2015, pp. 300–307.

[5] P. R. Barragán, L. P. Kaelbling, and T. Lozano-Pérez, "Interactive bayesian identification of kinematic mechanisms," in *International Conference on Robotics and Automation*, 2014.

[6] V. Högman, M. Björkman, A. Maki, and D. Kragic, "A sensorimotor learning framework for object categorization," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 8, no. 1, pp. 15–25, 2016.

[7] J. K. Mills and D. M. Lokhorst, "Control of robotic manipulators during general task execution: A discontinuous control approach," *International Journal of Robotics Research*, vol. 12, no. 2, pp. 146–163, 1993.

[8] S. Sidhik, M. Sridharan, and D. Ruiken, "Learning hybrid models for variable impedance control of changing-contact manipulation tasks," in *Annual Conference on Advances in Cognitive Systems*, 2020.

[9] F. Pfeiffer and C. Glocker, *Multibody dynamics with unilateral contacts*. John Wiley & Sons, 1996.

[10] M. Yashima and H. Yamaguchi, "Complementarity formulation for multi-fingered hand manipulation with rolling and sliding contacts," in *International Conference on Robotics and Automation*, vol. 2, 2003, pp. 2255–2261.

[11] M. Moore and J. Wilhelms, "Collision detection and response for computer animation," in *15th conference on Computer graphics and interactive techniques*, 1988, pp. 289–298.

[12] E. J. Routh *et al.*, *Dynamics of a system of rigid bodies*. Dover New York, 1955.

[13] S. A. Khader, H. Yin, P. Falco, and D. Kragic, "Data-efficient model learning and prediction for contact-rich manipulation tasks," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4321–4328, 2020.

[14] G. Lee, Z. Marinho, A. M. Johnson, G. J. Gordon, S. S. Srinivasa, and M. T. Mason, "Unsupervised learning for nonlinear piecewise smooth hybrid systems," *arXiv preprint arXiv:1710.00440*, 2017.

[15] R. Grassmann, L. Johannsmeier, and S. Haddadin, "Smooth point-to-point trajectory planning in $se(3)$ with self-collision and joint constraints avoidance," in *International Conference on Intelligent Robots and Systems*, 2018, pp. 1–9.

[16] S.-H. Nam and M.-Y. Yang, "A study on a generalized parametric interpolator with real-time jerk-limited acceleration," *Computer-Aided Design*, vol. 36, no. 1, pp. 27–36, 2004.

[17] A. Piazzi and A. Visioli, "Global minimum-jerk trajectory planning of robot manipulators," *IEEE transactions on industrial electronics*, vol. 47, no. 1, pp. 140–149, 2000.

[18] P. Freeman, "Minimum jerk trajectory planning for trajectory constrained redundant robots," Ph.D. dissertation, Washington University in St. Louis.

[19] P. Huang, Y. Xu, and B. Liang, "Global minimum-jerk trajectory planning of space manipulator," *International Journal of Control, Automation, and Systems*, vol. 4, no. 4, pp. 405–413, 2006.

[20] L. Villani and J. De Schutterm, "Force control," in *Handbook of Robotics*. Springer, 2008.