Easing Reliance on Collision-free Planning with Contact-aware Control

Tao Pang and Russ Tedrake
{pangtao, russt}@csail.mit.edu

***Abstract*— We believe that the future of robot motion planning will look very different than how it looks today: instead of complex collision avoidance trajectories with a brittle dependence on sensing and estimation of the environment, motion plans should consist of smooth, simple trajectories and be executed by robots that are not afraid of making contact. Here we present a "contact-aware" controller which continues to execute a given trajectory despite unexpected collisions while keeping the contact force stable and small. We introduce a quadratic programming (QP) formulation, which minimizes a trajectory-tracking error subject to quasistatic dynamics and contact-force constraints. Compared with the classical null-space projection technique, the inequality constraint on contact forces in the proposed QP controller allows for more gentle release when the robot comes out of contact. In the quasistatic dynamics model, control actions consist only of commanded joint positions, allowing the QP controller to run on stiffness-controlled robots which do not have a straightforward torque-control interface nor accurate dynamic models. The effectiveness of the proposed QP controller is demonstrated on a KUKA iiwa arm. Project video: https://youtu.be/M-7JMQRkiPk.**

I. Introduction

Most robots today are programmed to move through the world as if they are afraid of making contact. Perhaps they should be: unexpected collisions while a robot is tracking a trajectory can create a large force at the point of contact, putting at risk both the robot and the environment with which it interacts (Fig. 1). Consequently, a significant amount of effort and care in robot motion planning is spent on avoiding collisions. For example, sampling-based planners need to perform numerous collision checks [1]; optimization-based planners need to constantly evaluate the signed distance functions and their gradients [2]. Moreover, high resolution collision geometries are usually needed to increase the chances of finding a collision-free path, which further increases the computational cost [3].

However, the complete and total avoidance of contacts is a severe limitation, even if we are willing to tolerate the computational cost. First, the effectiveness of collision-free planning is limited by the quality of the geometric models used for collision checks. Unless in structured environments where everything has been perfectly measured, models of the environment need to be reconstructed from range sensor (e.g. depth camera) measurements, which usually have a fair amount of uncertainty and can suffer from occlusion. Moreover, collision-free trajectories can be unnecessarily conservative [4]: a task achievable by making some contacts can be deemed infeasible by a collision-free planner.

In this work, we propose a QP controller which, given estimated contact positions and forces in unexpected contacts, tracks the reference trajectory as closely as possible while keeping contact forces below a user-defined upper bound. Compared with similar controllers based on null-space projection [5]–[9], the QP formulation shares the same underlying dynamics, but allows for more gentle separation when the robot breaks contact with the environment. The gentle separation happens naturally as a result of bounding the contact forces with inequality constraints, which are not supported by null-space projection.

This work is supported by Navy-ONR Award N00014-18-1-2210 and Lincoln Laboratory Award PO# 7000470769.

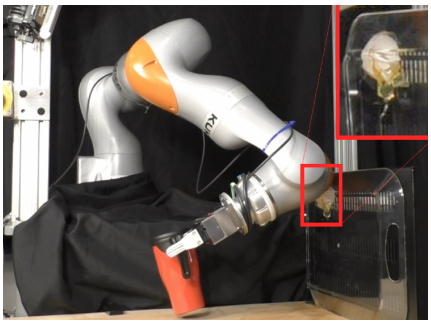

Fig. 1: As transparent obstacles are almost invisible to depth sensors, a collision-free motion planner, with the goal to pick up the mug and unaware of the transparent tray, plans a trajectory that crushes the egg along the way. The crushed egg is highlighted in the red box. Our controller is able to keep the egg intact even when the reference trajectory would crush it.

Instead of the usual second-order dynamics constraints used in robot locomotion [10], [11], the proposed QP controller utilizes a quasistatic dynamics model which predicts future equilibrium configurations and contact forces for a stiffness-controlled robot in response to position commands. By assuming bi-lateral, frictionless contacts, the quasistatic dynamics model can be expressed as equality constraints and does not require the estimation of friction coefficients. Moreover, as real-world contacts are uni-lateral and frictional, we also propose measures which both capitalize the simplicity of bi-lateral, frictionless contact models and mitigate the side effects of modeling real-world contacts as such.

II. Related Work

A. Interaction control

As the primary objective of the proposed QP controller is to bound unexpected contact forces while tracking a joint-space or end-effector trajectory, we review existing methods for combined motion and force control, which are also referred to as interaction control in the literature [12, Chapter 9]. Interaction control techniques can be broadly classified by whether the interaction force is controlled *directly* or *indirectly*.

Direct force control typically splits the task space into two orthogonal subspaces based on the robot's kinematic

constraints: one motion-controlled subspace along the tangents of the kinematic constraints, and one force-controlled subspace along the normals. Desired motion and force trajectories are specified in the motion and force controlled subspaces, and tracked independently using motor torque commands computed from the robot's second-order model [13]–[15].

However, the success of direct force control relies on accurate robot and environment models, which are not easily available in unstructured environments. Moreover, with few exceptions [16], most industrial robot arms, including the KUKA iiwa, do not have an interface for end-users to directly control motor torques [17]. Last but not least, by directly controlling motor torque, the interaction controller bypasses the robot's factory motion controller, and thus needs to run at high frequency in order to maintain stability.

On the other hand, a classical example of indirect force control is impedance control [18], which regulates the robot's response to external forces to that of a mass-spring-damper system (a mechanical impedance), thereby guaranteeing interaction stability by passivity. When the robot moves slowly, which is often the case in manipulation tasks, impedance control can be simplified to stiffness control [19], which can be interpreted as connecting the robot's end effector to a user-specified set-point by virtual springs. In the presence of contact, contact force can be controlled by commanding how much the set point penetrates the obstacle.

Compared with direct force control, indirect force control schemes are usually implemented as an outer-loop around the robot's factory motion controller, and therefore does not bear the responsibility of maintaining stability and can run at a much lower rate.

B. Quasistatic dynamics models in manipulation

Quasistatic dynamics has been used with great success to simplify the planning and control of simple tasks such as planar pushing [20], where modeling robots as prescribed motion trajectories is sufficient. However, for multi-contact tasks such as grasping [21], such simplifications can lead to non-unique contact forces or violation of non-penetration constraints [22], [23]. By modeling robots as impedances, our recent multibody quasistatic model [24] can faithfully reproduce the steady-state behavior of stiffness-controlled robot arms in multi-contact scenarios.

The quasistatic robot dynamics model proposed in this work, which predicts future joint angles in whole-arm contact scenarios, is an extension to classical indirect force control schemes that typically focus solely on the relationship between end effector pose and wrench [18], [19]. The proposed quasistatic dynamics is also a frictionless simplification of the Coulomb-friction-based quasistatic dynamics [24], which can be implemented on hardware with minimal contact sensing.

C. Null-space projection

Null-space projection is a classical and popular technique for executing a hierarchy of tasks defined by equality constraints [5]–[9]. The constraints imposed by higher-priority tasks are enforced by projecting the torque needed by low-priority tasks into the null space of the higher-priority tasks. The projections are defined over the ranges and null spaces of the task Jacobians and their *weighted* pseudo-inverses. It is noteworthy that the projections are generally not orthogonal, unless the weight matrix is identity [25].

More recently, controllers based on constrained optimizations such as quadratic programs (QP) have gained popularity in both locomotion [10], [11] and manipulation [26]. Compared with null-space projections, QP-based controllers can handle both equality and inequality constraints. In this work, we formulate the problem of trajectory tracking with bounded contact forces as a QP with a novel quasistatic dynamics constraint. We also show that a controller based on null-space projection implicitly enforces the same quasistatic dynamics constraint when the projection is *stiffness-consistent* [25].

III. Background and Notations

A. Constrained Inverse Dynamics Control

A popular controller in locomotion and manipulation is based on the following optimization-based formulation [10], [11], [27]:

$$\min_{x^{l+1},u^l} \; c_x(x^{l+1}) + c_u(u^l), \text{ s.t.} \tag{1a}$$

$$x^{l+1} = f(x^l, u^l), \tag{1b}$$

where x^l and u^l are the state and input at the current time step l, and x^{l+1} is the state at the next time step, $l+1$. The controller (1) picks an action u^l that minimizes the (usually LQR-style) state and action cost, $c_x(\cdot)$ and $c_u(\cdot)$, subject to the dynamics constraint (1b).

The most common choice for the dynamics constraint (1b) is the Newton's Second Law (N2L). For example, the DLR (German Aerospace Center) family of robots, including the KUKA iiwa and FRANKA panda, has the following closed-loop second-order dynamics after gravity compensation [28]:

$$M(q)\ddot{q} + (C(q,\dot{q}) + D_q)\,\dot{q} + K_q(q - q_{\text{cmd}}) = \tau_{\text{ext}}, \tag{2}$$

where $q \in \mathbb{R}^{n_q}$ is the joint angles of the robot, $C(q,\dot{q})\dot{q}$ is the Coriolis force, D_q is a diagonal damping matrix, K_q is a diagonal stiffness matrix, q_{cmd} is the commanded joint angles, and τ_{ext} is the torque by external contacts.

B. Contact and multibody notations

We consider rigid, point contacts in this work. The number of contacts the robot makes with the environment is denoted by n_c. Each contact point is denoted by C_i. The coordinates of the contact point relative to world frame, expressed in world frame is written as ${}^{W}p^{C_i} \in \mathbb{R}^3$; contact force at C_i expressed in world frame is represented by $f_W^{C_i} \in \mathbb{R}^3$.

Position Jacobian of the contact point C_i relative to frame W, expressed in frame W, is denoted by $J_q^{{}^{W}p^{C_i}}(q) : \mathbb{R}^{n_q} \to \mathbb{R}^{3\times n_q}$. It maps the robot's joint velocity $\dot{q}$ to the velocity of point C_i in world frame W:

$${}^{W}v^{C_i} = J_q^{{}^{W}p^{C_i}}(q)\dot{q}. \tag{3}$$

We further define

$$f_i := \left\| f_W^{C_i} \right\| \in \mathbb{R};\ u_i := f_W^{C_i}/f_i \in \mathbb{R}^3 \tag{4a}$$

$$J_{u_i} := u_i^\intercal J_q^{W\,p^{C_i}} \in \mathbb{R}^{1\times n_q} \tag{4b}$$

$$J_u := \left[J_{u_1}^\intercal, J_{u_2}^\intercal, \cdots J_{u_{n_c}}^\intercal\right]^\intercal \in \mathbb{R}^{n_c\times n_q}. \tag{4c}$$

The i-th row of J_u maps $\dot{q}$ to the Cartesian velocity of C_i along $\boldsymbol{u}_i$ in world frame. We assume that J_u is full-rank.

IV. Quasistatic Dynamics

A. Dynamics as transitions between equilibria

For robot arms with a joint-level stiffness controller, their steady-state equilibrium condition can be obtained by setting the derivative terms in the second-order dynamics (2) to 0:

$$K_q(q_{\text{cmd}} - q) + \tau_{\text{ext}} = 0. \tag{5}$$

For a stiffness-controlled robot, we can define its *quasi-static* dynamics, whose state consists only of the joint angles q, and input the commanded joint angles q_{cmd}. As shown in Fig. 2, the quasistatic dynamics predicts $x^{l+1} := q^{l+1}$, the equilibrium configuration at the next time step, from the current equilibrium configuration q^l and the next commanded configuration $u^l := q_{\text{cmd}}^{l+1}$.

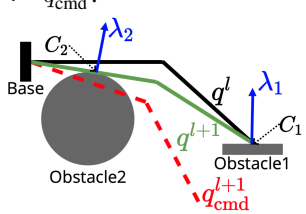

Fig. 2: Quasistatic dynamics of a 2D, 2-link robot arm. The arm starts at q^l (black) and is commanded to go to q_{cmd}^{l+1} (red). The virtual spring connecting q_{cmd}^{l+1} to q^l pulls the robot towards q_{cmd}^{l+1}, but the robot eventually stabilizes to q^{l+1} (green) due to contact constraints. At $l+1$, the arm makes two contact with two obstacles at C_1 and C_2 with contact forces λ_1 and λ_2.

The new equilibrium q^{l+1} can be solved for by minimizing the potential energy of the robot, subject to the contact constraints:

$$\min_{q^{l+1}} \frac{1}{2}(q_{\text{cmd}}^{l+1} - q^{l+1})^\intercal K_q (q_{\text{cmd}}^{l+1} - q^{l+1}) \text{ s.t.} \tag{6a}$$

$$J_u(q^l)(q^{l+1} - q^l) = 0. \tag{6b}$$

To derive the contact forces, we start with the Lagrangian of QP (6):

$$\begin{aligned} L(q^{l+1}, \lambda) =& \frac{1}{2}(q_{\text{cmd}}^{l+1} - q^{l+1})^\intercal K_q (q_{\text{cmd}}^{l+1} - q^{l+1}) \\ &- (\lambda^{l+1})^\intercal J_u (q^{l+1} - q^l) \end{aligned} \tag{7}$$

where $\lambda^{l+1} \in \mathbb{R}^{n_c}$ is the Lagrange multipliers of the contact constraint (6b), which can also be interpreted as the contact forces generated by (6b); the dependency of J_u on q^l is dropped for simplicity.

The KKT optimality condition of QP (6) is given by

$$\nabla_{q^{l+1}} L = K_q(q^{l+1} - q_{\text{cmd}}^{l+1}) - J_u^\intercal \lambda^{l+1} = 0, \tag{8a}$$

$$J_u\left(q^{l+1} - q^l\right) = 0. \tag{8b}$$

where (8a) is equivalent to the steady-state force balance condition (5), assuming that τ_{ext} is generated by the n_c point contacts, i.e. $\tau_{\text{ext}} = J_u^\intercal \lambda^{l+1}$.

Explicit expressions for λ^{l+1} and q^{l+1} can also be derived from the KKT conditions (8):

$$\lambda^{l+1} = -(J_u K_q^{-1} J_u^\intercal)^{-1} J_u (q_{\text{cmd}}^{l+1} - q^l), \tag{9a}$$

$$q^{l+1} = q^l + \left(I - K_q^{-1} J_u^\intercal (J_u K_q^{-1} J_u^\intercal)^{-1} J_u\right)(q_{\text{cmd}}^{l+1} - q^l). \tag{9b}$$

B. Relationship to null-space projection

In this sub-section, we show that when controlling stiffness-controlled robots using null-space projection, lower-priority tasks can be guaranteed to not interfere with higher-priority tasks during transients if the stiffness-consistent projection [25] is used. We also show that the underlying dynamics model of a controller based on stiffness-consistent projection is the same as the model proposed in Sec. IV-A.

Null-space projection technique revolves around two projections:

$$P_R(W) := J_u^\intercal \left(J_u^{W+}\right)^\intercal, \tag{10a}$$

$$P_N(W) := I - J_u^\intercal \left(J_u^{W+}\right)^\intercal, \tag{10b}$$

where $J_u^{W+} := W^{-1} J_u^\intercal (J_u W^{-1} J_u^\intercal)^{-1}$ is the pseudo-inverse weighted by a positive-definite W. The range and null space of $P_R(W)$ are respectively $R(J_u^\intercal)$ and $N\left(\left(J_u^{W+}\right)^\intercal\right)$, whereas the range and null space of $P_N(W)$ are reversed. Note that such projections can be defined for arbitrary task Jacobians, but we specialize to the contact Jacobian defined in (4c) without loss of generality.

For any choice of W and any joint torque τ_{m}, $P_R(W)\tau_{\text{m}} \in R(J_u^\intercal)$ generates contact forces, whereas $P_N(W)\tau_{\text{m}}$ generates no joint torque in $R(J_u^\intercal)$ after static equilibrium is reached. Choosing an appropriate W, however, can provide additional guarantees during the transient into this steady state [25]. For instance, the *dynamically-consistent* pseudo-inverse J_u^{M+} [29], which uses the robot's mass matrix M for W, ensures that

$$0 \equiv J_u M^{-1} P_N(M)\tau_{\text{m}},\ \forall \tau_{\text{m}}. \tag{11}$$

Assuming that the dominant effect of $P_N(M)\tau_{\text{m}}$ during the transient is to generate acceleration, property (11) guarantees that and the generated acceleration lies inside $N(J_u)$.

To determine the appropriate choice of W when the dominant effect of the $P_N(W)\tau_{\text{m}}$ during transient is to stretch/contract the virtual spring of a stiffness-controlled robot, we start at the instant l^+, immediately after sending the joint angle command $u^l = q_{\text{cmd}}^{l+1}$ at time step l. At l^+, the joint torque can be expressed as

$$\tau_{\text{m}}^{l^+} = K_q(q_{\text{cmd}}^{l+1} - q^l), \tag{12}$$

which can be decomposed as

$$\tau_{\text{m}}^{l^+} = \underbrace{P_R(W)\tau_{\text{m}}^{l^+}}_{\tau_R} + \underbrace{P_N(W)\tau_{\text{m}}^{l^+}}_{\tau_N}, \tag{13}$$

where $\tau_R \in R(J_u^\intercal)$ generates contact forces, and $\tau_N \in N\left(\left(J_u^{W+}\right)^\intercal\right)$ generates a motion that needs to be in $N(J_u)$.

Combining (12) and (13) yields

$$K_q(q_{\text{cmd}}^{l+1} - q^l) = \tau_R + \tau_N. \tag{14}$$

At time instant $(l+1)^-$, when the equilibrium at time step $l+1$ is reached but u^{l+1} has not been commanded, τ_N has

generated a displacement and been dissipated by damping, but τ_R, the generalized force due to contact, remains:

$$\tau_{\text{m}}^{(l+1)^-} = \tau_R. \tag{15}$$

Moreover, static equilibrium (5) at $(l+1)^-$ dictates that

$$\tau_{\text{m}}^{(l+1)^-} = K_q(q_{\text{cmd}}^{l+1} - q^{l+1}). \tag{16}$$

Combining (15) and (16) yields

$$K_q(q_{\text{cmd}}^{l+1} - q^{l+1}) = \tau_R. \tag{17}$$

Finally, subtracting (17) from (14) gives

$$K_q(q^{l+1} - q^l) = \tau_N = P_N(W)\tau_{\text{m}}^{l^+}. \tag{18}$$

As the motion from l to $l+1$ needs to stay in $N(J_u)$, we need $J_u(q^{l+1} - q^l) \equiv 0$, which implies through (18) that

$$0 \equiv J_u K_q^{-1} P_N(W)\tau_{\text{m}}^{l^+},\ \forall \tau_{\text{m}}^{l^+}, \tag{19}$$

which has the same form as dynamic-consistency defined in (11). Not surprisingly, choosing $W = K_q$ ensures the motion generated by τ_N stays in $N(J_u)$, and the resulting pseudo-inverse J^{K_q+} is called *stiffness-consistent* [25].

We can solve for q^{l+1} by plugging (12) into (18) and setting W to K_q, yielding

$$q^{l+1} = q^l + (I - J_u^{K_q+} J_u)(q_{\text{cmd}}^{l+1} - q^l). \tag{20}$$

Furthermore, as the contact force is the reaction to τ_R, we have $\tau_R = -J_u^\intercal \lambda^{l+1}$, combining this with the definition of τ_R in (13) gives

$$\lambda^{l+1} = -\left(J^{K_q+}\right)^\intercal K_q(q_{\text{cmd}}^{l+1} - q^l). \tag{21}$$

It can be shown that (20) and (21) are equivalent to (9b) and (9a), respectively. This equivalence implies that a controller based on stiffness-consistent projections and a controller (1) using QP (6) as its dynamics constraint share the same underlying dynamics model. Therefore, the QP formulation in Sec. V is preferred as it can handle inequality constraints.

V. QP Controller with Quasistatic Dynamics

A. Frictionless contacts

To track a reference trajectory $q_{\text{ref}}(t)$ as closely as possible while respecting dynamics constraints and upper bounds on contact forces, we can specialize the generic optimization-based controller (1) to the following QP:

$$\min_{q^{l+1}, q_{\text{cmd}}^{l+1}, \lambda^{l+1}} \left\|q^{l+1} - q_{\text{ref}}^{l+1}\right\|^2 + \epsilon \left\|q_{\text{cmd}}^{l+1} - q_{\text{ref}}^{l+1}\right\|^2, \text{s.t.} \tag{22a}$$

$$K_q(q^{l+1} - q_{\text{cmd}}^{l+1}) - J_u^\intercal \lambda^{l+1} = 0 \tag{22b}$$

$$J_u(q^{l+1} - q^l) = 0 \tag{22c}$$

$$\lambda^{l+1} \leq \lambda_{\max} \tag{22d}$$

$$\left|q_{\text{cmd}}^{l+1} - q_{\text{cmd}}^l\right| \leq \Delta q_{\max}. \tag{22e}$$

Here, the dynamics constraint (1b) consists of (22b) and (22c), which are the KKT conditions (8) of the quasistatic dynamics (6). Constraint (22d) places an upper bound on contact forces. The last constraint (22e) bounds how quickly q_{cmd} changes.

In the objective (22a), the first term penalizes deviation at $l+1$ from the reference trajectory. The second term, weighted by a small positive scalar ϵ, adds regularization without which the objective would become semi-definite. To see why, we re-write $\left(q^{l+1} - q_{\text{ref}}^{l+1}\right)$ by expressing q^{l+1} explicitly as a function of q_{cmd}^{l+1} using (20):

$$q^{l+1} - q_{\text{ref}}^{l+1} = (I - J_u^{K_q+} J_u)(q_{\text{cmd}}^{l+1} - q^l) + (q^l - q_{\text{ref}}^{l+1}), \tag{23}$$

where the second term is a constant, and the first term multiplies $(q_{\text{cmd}}^{l+1} - q^l)$ by a projection which has a non-zero null space.

In the quasistatic dynamics (6), expressing contact constraints as equality constraints (6b) and contact forces as the constraints' Lagrange multipliers implies that the contacts are bi-lateral and frictionless. In reality, however, contacts are uni-lateral and frictional.

The bi-lateralness of (6b) is less concerning. As contact sensors are inevitably noisy, only contact forces above a threshold are added to QP (22). In addition, by (9a), the change in contact force is bounded as long as $\left|q_{\text{cmd}}^{l+1} - q^l\right|$ is bounded, and the boundedness of $\left|q_{\text{cmd}}^{l+1} - q^l\right|$ is enforced by (22e). Therefore, as long as the bound $\Delta q_{\max}$ is sufficiently small, we do not need to worry about contact forces flipping sign in the middle of a control step.

On the other hand, naively ignoring friction will severely impact the performance of the controller, which motivates the mitigating measures detailed in the next subsection.

B. Frictional contacts

It is possible to model friction contact in quasistatic dynamics [24], but control through a frictional contact requires estimating the contact normal and the friction coefficient, in addition to estimating contact forces. This requires either more sophisticated whole-arm contact sensors, or making additional assumptions about the environments that make the control-estimation pipeline more brittle.

Therefore, we will retain the simpler frictionless contact model for controlling through a frictional contact, and mitigate the side effects of the wrong contact model by modifying the frictionless QP (22) to

$$\min_{q^{l+1}, q_{\text{cmd}}^{l+1}, \lambda^{l+1}} \underbrace{\left\|q_{\text{cmd}}^{l+1} - q_{\text{ref}}^{l+1}\right\|^2}_{\text{tracking}} + \underbrace{w^l \left\|q_{\text{cmd}}^{l+1} - q_{\text{cmd}}^l\right\|^2}_{\text{damping}} \tag{24a}$$

s.t. (22b), (22c), (22d) and (22e),

which has the same constraints as (22) but a different objective. In the rest of this section, we will elaborate on the reason for both terms in the objective (24a).

1) Tracking: When the reference trajectory q_{ref}^{l+1} leads the robot to make contact with a frictional surface at point C (Fig. 3a), the surface normal $n \in \mathbb{R}^3$ and the contact force direction $u \in \mathbb{R}^3$ can be different. However, the frictionless contact model (22b)-(22c) assumes that u is always the same as n. Therefore, it is possible for ${}^W v_{\text{cmd}}^C$, the commanded velocity of C, to have a negative component along u but a positive component along n, as shown in Fig. 3a. Such a ${}^W v_{\text{cmd}}^C$ would lead to the robot separating from the obstacle at $l+1$. When the frictionless QP (22) is constructed again at $l+1$, no contact force constraints are added but q_{ref}^{l+2} can still

lead the robot to contact with a large amount of penetration. Therefore, the robot could re-establish contact with a large contact force at $l+2$.

Although it is difficult to guarantee that ${}^{W}v^{C}_{\text{cmd}}$ has a negative component along n without knowing n, undesired contact jitters can be effectively reduced by making ${}^{W}v^{C}_{\text{cmd}}$ as close as possible to ${}^{W}v^{C}_{\text{ref}}$. In joint space, this translates to minimizing the distance between q^{l+1}_{cmd} and q^{l+1}_{ref}, which can be achieved by replacing the term $\left\|q^{l+1}-q^{l+1}_{\text{ref}}\right\|^2$ in (22a) by $\left\|q^{l+1}_{\text{cmd}}-q^{l+1}_{\text{ref}}\right\|^2$ in (24a).

To further illustrate the advantage of the new objective, we first re-write $q^{l+1}_{\text{cmd}}-q^{l+1}_{\text{ref}}$ using the relative quantities defined in Fig. 3b:

$$q^{l+1}_{\text{cmd}}-q^{l+1}_{\text{ref}}=\left(q^{l+1}_{\text{cmd}}-q^{l}\right)-\left(q^{l+1}_{\text{ref}}-q^{l}\right)=\Delta q^{l+1}_{\text{cmd}}-\Delta q^{l+1}_{\text{ref}}. \tag{25}$$

It is also easy to see from (23) that

$$\Delta q^{l+1}=(I-J_u^{K_q+}J_u)\Delta q^{l+1}_{\text{cmd}}. \tag{26}$$

The first term in the original objective (22a) thus becomes

$$\begin{aligned} q^{l+1}-q^{l+1}_{\text{ref}} =&\Delta q^{l+1}-\Delta q^{l+1}_{\text{ref}} \\ =&(I-J_u^{K_q+}J_u)\Delta q^{l+1}_{\text{cmd}}-\Delta q^{l+1}_{\text{ref}}, \end{aligned} \tag{27}$$

where $(I-J_u^{K_q+}J_u)$ is the projection into $N(J_u)$ along $R\left(J_u^{K_q+}\right)$, as shown in Fig. 3b.

As a result, minimizing $\left\|\Delta q^{l+1}_{\text{cmd}}-\Delta q^{l+1}_{\text{ref}}\right\|^2$ encourages $\Delta q^{l+1}_{\text{cmd}}$ to be close to $\Delta q^{l+1}_{\text{ref}}$ in the entire vector space. In contrast, when $\left\|\Delta q^{l+1}-\Delta q^{l+1}_{\text{ref}}\right\|^2$ is used as the cost, only the distance between $\Delta q^{l+1}_{\text{ref}}$ and the component of $\Delta q^{l+1}_{\text{cmd}}$ along $N(J_u)$ is minimized.

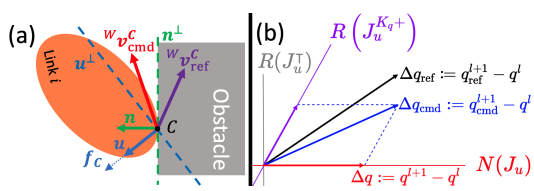

Fig. 3: (a): The reference trajectory brings the robot into contact at C. Due to friction, the contact normal n and the contact force direction u are different. Therefore, the commanded velocity at C can separate from the obstacle even when the angle between u and ${}^{W}v^{C}_{\text{cmd}}$ is greater than $\pi/2$. **(b)**: Definitions of Δq_{ref}, Δq_{cmd} and Δq. The range and null space of the projection $I-J_u^{K_q+}J_u$ are $N(J_u)$ and $R\left(J_u^{K_q+}\right)$, respectively.

2) Damping: The goal of this term is to command more conservative robot motions when we are less confident in the correctness of the frictionless contact model. As real-world contacts are almost always frictional, the contact force predicted by the frictionless model and the actual contact force measured by contact sensors are bound to be different. At every time step, this discrepancy can be quantified by

$$e^l_\lambda := 1-\exp\left(\left\|\lambda^l_{\text{pred}}-\lambda^l_{\text{est}}\right\|_\infty/a\right)\in[0,1], \tag{28}$$

where λ^l_{pred} is the contact forces predicted by the frictional QP (24) at time step $l-1$; λ^l_{est} is the measured contact forces at time step l; a is a positive constant that weights the force prediction error. The discrepancy e_λ is close to 1 when the force prediction error is large, and close to 0 when the error is small.

The weight of the second term of (24a), w^l, is the low-pass-filtered version of the discrepancy e^l_λ:

$$w^l=w_{\max}\left[\alpha e^l_\lambda+(1-\alpha)e^{l-1}_\lambda\right], \tag{29}$$

where $w_{\max}$ is the upper bound on w^l and α is the forget rate of the low-pass filter. A larger w^l encourages more conservative robot motions by more heavily penalizing the change in q_{cmd} from l to $l+1$.

VI. Experiments

In this section, we demonstrate the advantages of the proposed contact-aware controller through two tasks that involve unexpected contact with the environment, which are shown in Fig. 4. In both experiments, q_{cmd} is tracked using the iiwa's factory impedance controller. The factory controller's stiffness is set to $[800,600,600,600,400,200,200]\ \text{N}\cdot\text{m/rad}$, from base joint to wrist joint. The proposed contact-aware controller runs at 200Hz. The external contacts are estimated from iiwa's external torque measurements using the Contact Particle Filter [30], which runs at around 100Hz. QPs are constructed using Drake's `MathematicalProgram` interface [31] and solved by GUROBI [32].

In order to reduce sensitivity to measurement noise, only contact forces with norm $f_i\geq f_{\text{threshold}}$ are considered when constructing the contact Jacobian J_u (4c). We have chosen $f_{\text{threshold}}=5\text{N}$, and set the upper bound on contact forces in (22d) to be $\lambda_{\max}=15\text{N}$. Ignoring contacts with small contact forces can be justified by the passivity of the robot's internal controller [33], which ensures stability in the presence of external contacts.

A. Mug placement task (Fig. 4a)

This task is defined by an end-effector pose trajectory $\left({}^{W}R^{T_r}(t),{}^{W}p^{T_r}(t)\right)$, where T_r is the reference for the tool frame T, ${}^{W}R^{T_r}$ is the orientation of frame T_r w.r.t. the world frame W, and ${}^{W}p^{T_r}$ is the position of the origin of T_r in world frame. It is straightforward to modify the tracking term in the frictional QP objective (24a) to minimize the pose difference between frame T and its reference T_r, as described in [11] [34, Chapter 3].

The robot starts with a mug held in the gripper. It then **(i)** reaches down ($-z$ of world frame) by $0.22m$ in 4s, **(ii)** opens the gripper and drops the mug on the cart below the table in 2s, and **(iii)** moves back up to where it started in 4s. The orientation of the gripper is kept constant throughout the trajectory. As shown in Fig. 4a, the "wrist" (link 6) of the robot collides with the edge of the table as the gripper moves down and up.

1) Contact force: As shown in Fig. 5, except during the initial impact, our controller is able to keep the contact force norm $\left\|f^C_W\right\|$ close to $\lambda_{\max}$. In contrast, the baseline controller we compare against, which computes q_{cmd} by greedily minimizing tracking error without the dynamics and contact force constraints (22b)-(22e), incurs larger contact forces.

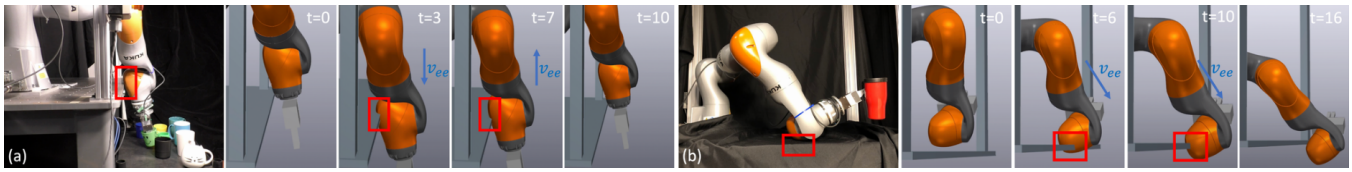

Fig. 4: (**a**): mug placement task. (**b**): mug moving task. Contacts are highlighted in red boxes. Blue arrows denote the direction of end effector velocity. For both tasks, the photographs on the left show where the real robot makes contact while executing the task. With collision geometry disabled, the simulation frames on the right show how much penetration would happen if the original trajectory were strictly followed. Videos of the real robot executing the tasks are included in the attachment.

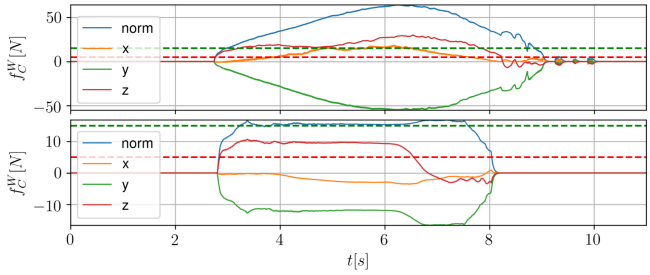

Fig. 5: mug placement task. x, y, z components and the norm of the contact force $\boldsymbol{f}_W^C$. **Top**: the baseline controller without contact force upper bounds is shown in the top plot. **Bottom**: the contact-aware controller (24) with a modified end effector tracking objective. In both plots, the horizontal red dashed line represents $f_{\text{threshold}}$ and the green dashed line $\lambda_{\max}$.

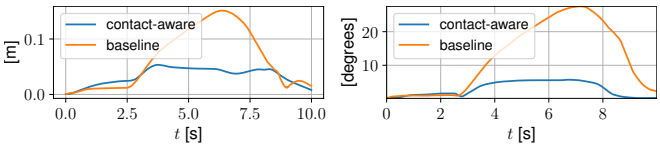

Fig. 6: mug placement task. **Left**: position tracking error. **Right**: orientation tracking error.

2) Tracking error: As shown in Fig. 6, compared with the baseline, the contact-aware controller produces significantly less tracking error in the presence of external contact.

B. Mug moving task (Fig. 4b)

This task is defined by a joint-space trajectory $q_{\text{ref}}(t)$, with the goal of moving the mug along a straight line while keeping the mug orientation constant. The 16s trajectory $q_{\text{ref}}(t)$ is an interpolation between joint-space knot points obtained by inverse kinematics. As shown in Fig. 4b, to move the mug to the desired destination, the robot needs to first establish a contact with the top face of the table, and then breaks contact with the side of the table.

The baseline we are comparing against is the class of controllers trying to achieve similar goals as ours but uses null-space projection, such as [9]. As null-space projection cannot handle inequality constraints, the contact force upper bound is usually enforced by an equality constraint which sets the contact force to $\lambda_{\max}$. When commanded to break contact by the reference trajectory, i.e.

$$J_u(q_{\text{ref}}^{l+1} - q^l) \leq 0, \tag{30}$$

the contact force constraint needs to be abruptly removed in order to continue to track the reference trajectory [9]. This can lead to large joint velocity during separation, as shown in Fig. 7.

In contrast, the proposed QP controller does not explicitly make the decision to break contact based on (30), instead the break of contact comes naturally as a consequence of solving QP (24) with the inequality constraints on contact forces (22d). As shown in Fig. 7, when the robot separates from the table, the drop in $\left\|f_W^C\right\|$ occurs gradually with our QP controller, but abruptly with a null-projection-based controller.

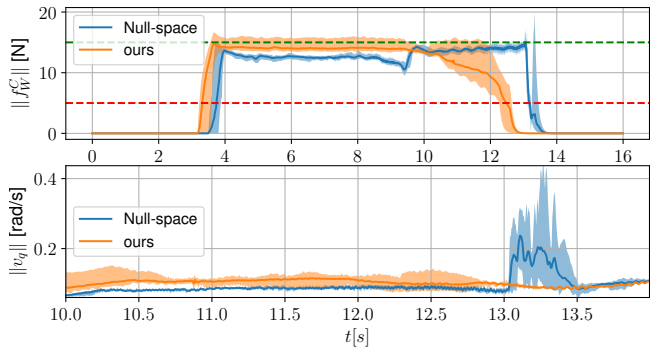

Fig. 7: Comparison between our QP controller and null-space projection-based control in the **mug moving task**. **Top**: contact force norm $\left\|f_W^C\right\|$. The red dashed line represents $f_{\text{threshold}}$ and the green dashed line $\lambda_{\max}$. Both controllers can keep $\left\|f_W^C\right\|$ near $\lambda_{\max}$ when the robot is in contact. **Bottom**: robot joint velocity norm $\|v_q\|$ during contact separation (from $t = 10s$ to $t = 14s$). Note the velocity spike in the null-space projection controller. In both plots, the solid lines are the mean of 10 runs; the shaded regions around the lines represent the maximum and minimum values of all runs.

VII. Conclusions

We have presented a contact-aware controller that reconciles trajectory tracking with safety in unexpected contacts. The proposed controller is formulated as a QP with a quadratic cost on tracking error, a quasi-static model of the robot dynamics as constraints, and upper bounds on contact forces.

The tasks for hardware experiments are designed based on our vision of future motion planners: they are comprised of smooth, simple trajectories defined by only a few knot points. We have shown that the proposed controller is able to keep both the tracking error and contact forces small if the robot makes an accidental contact. In addition, our controller outperforms controllers based on null-space projection when an established contact needs to be broken as the robot follows a reference trajectory.

It is difficult to reliably sense more than one contacts on the arm from only joint torque [35]. Nevertheless, with a more capable contact sensor such as [36], we believe the proposed contact-aware controller will greatly reduce robots' reliance on environment sensing/monitoring and collision-free motion planning.

REFERENCES

[1] S. M. LaValle, *Planning algorithms*. Cambridge university press, 2006.

[2] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "Chomp: Gradient optimization techniques for efficient motion planning," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 489–494.

[3] J. Pan, S. Chitta, and D. Manocha, "Fcl: A general purpose library for collision and proximity queries," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 3859–3866.

[4] M. T. Mason, "Toward robotic manipulation," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 1–28, 2018.

[5] Y. Nakamura, H. Hanafusa, and T. Yoshikawa, "Task-priority based redundancy control of robot manipulators," *The International Journal of Robotics Research*, vol. 6, no. 2, pp. 3–15, 1987.

[6] B. Siciliano and J.-J. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Fifth International Conference on Advanced Robotics' Robots in Unstructured Environments*. IEEE, 1991, pp. 1211–1216.

[7] F. Aghili, "A unified approach for inverse and direct dynamics of constrained multibody systems based on linear projection operator: applications to control and simulation," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 834–849, 2005.

[8] N. Dehio, J. Smith, D. L. Wigand, G. Xin, H.-C. Lin, J. J. Steil, and M. Mistry, "Modeling and control of multi-arm and multi-leg robots: Compensating for object dynamics during grasping," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 294–301.

[9] M. Jorda, E. G. Herrero, and O. Khatib, "Contact-driven posture behavior for safe and interactive robot operation," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 9243–9249.

[10] S. Kuindersma, F. Permenter, and R. Tedrake, "An efficiently solvable quadratic program for stabilizing dynamic locomotion," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 2589–2594.

[11] T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot atlas," *International Journal of Humanoid Robotics*, vol. 13, no. 01, p. 1650007, 2016.

[12] B. Siciliano, O. Khatib, and T. Kröger, *Springer handbook of robotics*. Springer, 2008, vol. 200.

[13] M. T. Mason, "Compliance and force control for computer controlled manipulators," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 11, no. 6, pp. 418–432, 1981.

[14] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal on Robotics and Automation*, vol. 3, no. 1, pp. 43–53, 1987.

[15] M. H. Raibert, J. J. Craig *et al.*, "Hybrid position/force control of manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 103, no. 2, pp. 126–133, 1981.

[16] Franka Emika GmbH, "Franka control interface documentation," 2019. [Online]. Available: https://frankaemika.github.io/docs/index.html

[17] KUKA Roboter GmbH, "Code based documentation of kuka fast robot interface c++ sdk," 2019.

[18] N. Hogan, "Impedance control: An approach to manipulation," in *1984 American control conference*. IEEE, 1984, pp. 304–313.

[19] J. K. Salisbury, "Active stiffness control of a manipulator in cartesian coordinates," in *1980 19th IEEE conference on decision and control including the symposium on adaptive processes*. IEEE, 1980, pp. 95–100.

[20] F. R. Hogan and A. Rodriguez, "Feedback control of the pusher-slider system: A story of hybrid and underactuated contact dynamics," in *Algorithmic Foundations of Robotics XII*. Springer, 2020, pp. 800–815.

[21] J.-S. Pang, J. C. Trinkle, and G. Lo, "A complementarity approach to a quasistatic multi-rigid-body contact problem," *Computational Optimization and Applications*, vol. 5, no. 2, pp. 139–154, 1996.

[22] M. Halm and M. Posa, "A quasi-static model and simulation approach for pushing, grasping, and jamming," in *International Workshop on the Algorithmic Foundations of Robotics*. Springer, 2018, pp. 491–507.

[23] T. Pang and R. Tedrake, "A robust time-stepping scheme for quasistatic rigid multibody systems," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 5640–5647.

[24] ——, "A convex quasistatic time-stepping scheme for rigid multibody systems with contact and friction," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.

[25] A. Dietrich, C. Ott, and A. Albu-Schäffer, "An overview of null space projections for redundant, torque-controlled robots," *The International Journal of Robotics Research*, vol. 34, no. 11, pp. 1385–1400, 2015.

[26] A. Jain, M. D. Killpack, A. Edsinger, and C. C. Kemp, "Manipulation in clutter with whole-arm tactile sensing," *arXiv preprint arXiv:1304.6146*, 2013.

[27] Y. Wang and A. Kheddar, "Impact-friendly robust control design with task-space quadratic optimization," in *Robotics: Science and Systems (RSS)*, 2019.

[28] C. Ott, A. Albu-Schaffer, A. Kugi, and G. Hirzinger, "On the passivity-based impedance control of flexible joint robots," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 416–429, 2008.

[29] R. Featherstone and O. Khatib, "Load independence of the dynamically consistent inverse of the jacobian matrix," *The International Journal of Robotics Research*, vol. 16, no. 2, pp. 168–170, 1997.

[30] L. Manuelli and R. Tedrake, "Localizing external contact using proprioceptive sensors: The contact particle filter," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 5062–5069.

[31] R. Tedrake and the Drake Development Team, "Drake: Model-based design and verification for robotics," 2019. [Online]. Available: https://drake.mit.edu

[32] Gurobi Optimization, LLC, "Gurobi optimizer reference manual," 2019. [Online]. Available: "http://www.gurobi.com"

[33] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A unified passivity-based control framework for position, torque and impedance control of flexible joint robots," *The international journal of robotics research*, vol. 26, no. 1, pp. 23–39, 2007.

[34] R. Tedrake, "Robot manipulation: Perception, planning, and control (course notes for mit 6.881)," 2021. [Online]. Available: http://manipulation.csail.mit.edu/

[35] T. Pang, J. Umenberger, and R. Tedrake, "Identifying external contacts from joint torque measurements on serial robotic arms and its limitations," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.

[36] Y. Luo, Y. Li, P. Sharma, W. Shou, K. Wu, M. Foshey, B. Li, T. Palacios, A. Torralba, and W. Matusik, "Learning human–environment interactions using conformal tactile textiles," *Nature Electronics*, vol. 4, no. 3, pp. 193–201, 2021.