Kinematics-Based Estimation of Contact Constraints Using Only Proprioception

Citation for published version:
Ortenzi, V, Lin, H-C, Azad, M, Stolkin, R, Jeffrey, K & Mistry, M 2016, Kinematics-Based Estimation of Contact Constraints Using Only Proprioception. in Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on. IEEE, Cancun, Mexico, pp. 1304-1311, 2016 IEEE-RAS 16th International Conference on Humanoid Robots, Cancun, Mexico, 15/11/16. DOI: 10.1109/HUMANOIDS.2016.7803438

Digital Object Identifier (DOI):
10.1109/HUMANOIDS.2016.7803438

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on

Kinematics-based estimation of contact constraints using only proprioception

Valerio Ortenzi[1] Hsiu-Chin Lin[2] Morteza Azad[2] Rustam Stolkin[1] Jeffrey A. Kuo[3] Michael Mistry[2]

***Abstract*— Robots are increasingly being required to perform tasks which involve contacts with the environment. This paper addresses the problem of estimating environmental constraints on the robot's motion. We present a method which estimates such constraints, by computing the null space of a set of velocity vectors which differ from commanded velocities during contacts. We further extend this method to handle unilateral constraints, for example when the robot touches a rigid surface. Unlike previous work, our method is based on kinematics analysis, using only proprioceptive joint encoders, thus there is no need for either expensive force-torque sensors or tactile sensors at the contact points or any use of vision. We first show results of experiments with a simulated robot in a variety of situations, and we analyse the effect of various levels of observation noise on the resulting contact estimates. Finally we evaluate the performance of our method on two sets of experiments using a KUKA LWR IV manipulator, tasked with exploring and estimating the constraints caused by a horizontal surface and an inclined surface.**

I. Introduction

The robotic revolution in manufacturing industries, over the past four decades, predominantly relied on robots which move payloads through unobstructed trajectories in free-space. In contrast, a new generation of robots is now needed which must cope with complex tasks, in uncertain environments, which frequently will involve forceful contacts between parts of the robot and surrounding objects or surfaces.

Many behaviours can be described as performing tasks under a set of contact constraints. For example, when a robot interacts with a horizontal surface (Fig 1), the end effector cannot penetrate the table. Some examples of tasks involving contacts includes: robotic grasping, active perception and manipulation; foot contacts in legged walking robots; grinding and polishing of cast parts in manufacturing; and tasks needed for nuclear decommissioning, such as "disruption" (cutting) or "scabbling" (grinding off the contaminated surface of a concrete room or "cave" to make it safe).

Our previous work [1], [2] showed how contacts can actually be exploited to enable robots to perform a desired motion task more efficiently, with reduced torques and energy consumption. This work was motivated by the ways in which humans exploit contacts. For example, an elderly person sweeping the floor with a broom, might lean on the broom handle to reduce loading on their lower back, while simultaneously performing the desired motion task of moving the broom across the floor surface. In [1] and [2] we used projected dynamics to decouple a motion task from force control in the null-space of the desired motion. We demonstrated this approach by tasking the robot with wiping a whiteboard, while also resting some of its weight on the whiteboard to reduce motor torques and energy consumption at the joints. We also showed how to control a desired contact force on the board, independently of controlling motion of the end-effector in the plane of the board. However, that work relied on prior knowledge of the position and orientation of the whiteboard surface relative to the robot's coordinate frame. In contrast, this paper explores the problem of how to detect and estimate the contact constraints by performing exploratory motions and using proprioception.

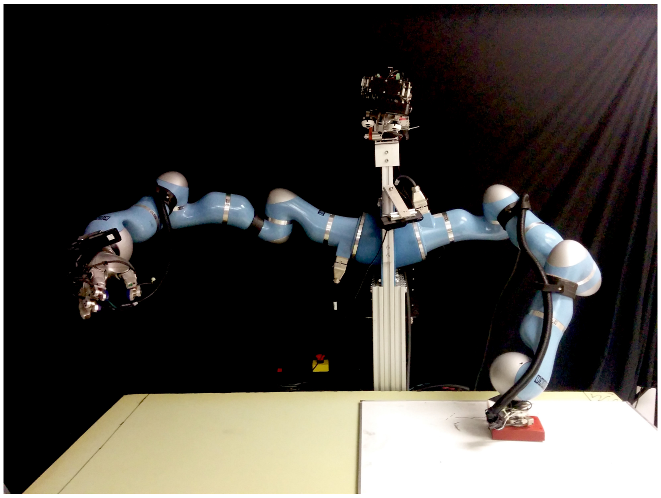


Fig. 1: Our bi-manual half-humanoid platform, Boris, with one of its end effectors in contact with a horizontal surface. This contact prevents the end effector from moving in the negative z direction of the task space.

Specifically, this paper presents a method for achieving reliable estimations of the constraints arising from contacts which limit the free motion space of the robot. We propose to use only kinematic observations derived from basic proprioception (rotation encoders at joints). Therefore, in contrast to much of the related literature, no added sensors are needed such as force-torque sensors or tactile sensors at the contact points, or feedback from a vision system.

Our proposed method would therefore be particularly useful for highly underactuated robotic arms, fingers or

Schools of Engineering[1] and Computer Science[2], University of Birmingham. [3]National Nuclear Laboratory Ltd. Contact: vxo344@bham.ac.uk

legs which contain passive (e.g. spring-loaded) joints, provided that basic position/rotation sensing is available at each joint. Such passive/underactuated robots are attracting increasing attention from the research community, e.g. [3], [4]. Additionally, in extreme environments such as nuclear decommissioning, beta and gamma radiation can destroy the delicate electronics needed for force, torque or tactile sensing. While intense gamma radiation can also destroy conventional proprioceptive rotation encoders (by causing the glass of optical encoders to become opaque), these can be replaced by electro-magnetic resolvers so that proprioception of each joint's rotation can still be reliably sensed in such environments. Our method could therefore be used to help such robots estimate the position and direction of constraint surfaces, e.g. for tasks such as scabbling (described above).

There is an extensive body of literature on the use of computer vision to infer object shapes and features, [5], [6], [7], [8]. There is also rapidly growing interest in estimation of object shapes and contacts using force-torque and tactile sensors, [9], [10], [11], [12], [13], [14], [15]. Recent work, [16], [17] has also begun exploring the fusion of visual and tactile data. In [18] a quadrupedal robot estimates the inclination of a planar surface on which it is trotting, by fusing data from IMU accelerometers with optical force sensors at each foot and the kinematics of each leg. In the context of hybrid motion/force control, [19] proposed a method for estimating the local shape of a constraint surface by combining position and end-effector force measurements. Another method for estimating constraints is proposed in [20], however this method learns the null space projector of an unknown task constraint from human demonstration.

In contrast to the above approaches, our method locally estimates the kinematic constraints due to contacts, without using any additional sensors (force, torque, tactile, vision) other than basic proprioception (rotation encoders at each joint). Specifically, we propose to perform a set of explorative actions and then estimate the kinematic constraints by observing the resulting motions. We present two variants, based either on a Cartesian space analysis or on a joint space analysis, and we also show how to discern unilateral constraints (*e.g.* contact with a rigid surface which only constrains motion in one direction). We first demonstrate our method with a simulated redundant 3 DOF planar robot, and show how it can detect and estimate various kinds of constraints. Next we analyse the effect of different levels of observation noise on the accuracy with which such constraints can be estimated. Finally we show the results of experiments carried out using a KUKA LWR IV robot arm, which is tasked with estimating the environmental constraints when contacting surfaces of different inclinations.

Our lab currently lacks passively compliant, underactuated robots. Therefore, for proof of principle we have instead used an actively-compliant KUKA LWR IV robot to demonstrate our method. We only make use of this robot's proprioceptive rotation encoders at each joint, and we do not explicitly make any use of joint torque information in our experiments. However, this robot does use torque sensing internally for low level control, to achieve compliant behaviours when in contact with environmental constraint surfaces.

The remainder of this paper is organised as follows. Sect. II introduces the notation used thereafter. Sect. III describes our method to estimate constraints. Sect. IV reports the results of experiments with both real and simulated redundant manipulators. Sect.V discusses the results and provides concluding remarks and suggestions for future work.

II. Fundamentals and notation

We assume a robot with n degrees of freedom operating in an m-dimensional workspace $\mathcal{W}$. Let $\boldsymbol{q}$ be the robot configuration taking values in an n_q-dimensional configuration space $\mathcal{C}$. A task is expressed in coordinates $\boldsymbol{p}$ and takes values in an m_y-dimensional space $\mathcal{Y}$. Task and robot configuration are related through the kinematic map:

$$\boldsymbol{p} = \boldsymbol{f}(\boldsymbol{q}) \tag{1}$$

The differential kinematics relations are:

$$\dot{\boldsymbol{p}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{2}$$

where $\boldsymbol{J}(\boldsymbol{q}) = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{q}}$ is the task Jacobian, which relates joint space velocities to task space velocities.

When the robot is in contact, motion is unfeasible in some subspace due to the constraints originated by the contact. We can describe the relationship between the constraint space and the joint space as:

$$\boldsymbol{J}_C(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{0} \tag{3}$$

where we denote the Jacobian of the constraints as $\boldsymbol{J}_C(\boldsymbol{q})$.

In the example of Fig 1, the movement in the z-axis toward the table surface is restricted since the table cannot be penetrated. In that particular example, the constraint Jacobian $\boldsymbol{J}_C(\boldsymbol{q})$ would be a matrix that relates the joint velocities to the end-effector velocities along the z-axis.

Using $\boldsymbol{J}_C(\boldsymbol{q})$, it is possible to define a projector $\boldsymbol{P}(\boldsymbol{q})$ which projects a vector into the null space of the constraints:

$$\boldsymbol{P}(\boldsymbol{q}) = \boldsymbol{I} - \boldsymbol{J}_C^{\dagger}(\boldsymbol{q})\boldsymbol{J}_C(\boldsymbol{q}) \tag{4}$$

In particular, we can also write that:

$$\boldsymbol{P}(\boldsymbol{q})\boldsymbol{J}_C^T(\boldsymbol{q}) = \boldsymbol{0} \tag{5}$$

and

$$(\boldsymbol{I} - \boldsymbol{P}(\boldsymbol{q}))\dot{\boldsymbol{q}} = \boldsymbol{0} \tag{6}$$

Equations 5 and 6 are the fundamental equations of projected dynamics [21] and represent respectively: the free motion subspace of the workspace, *i.e.* $\boldsymbol{P}(\boldsymbol{q})$ is a projector into the free motion subspace; and the force control subspace, *i.e.* $(\boldsymbol{I} - \boldsymbol{P}(\boldsymbol{q}))$ is a projector into the force control subspace, which is orthogonal to the free motion subspace. These two equations are particularly important when attempting to simultaneously control both the forces applied by the robot and its motion [1].

III. Method to Estimate the Constraints

First we propose a method to estimate kinematic constraints using exploration in the Cartesian space. Later we extend this reasoning to: explorations conceived in the joint space; and a method for handling unilateral constraints.

We consider that the contact Jacobian $\boldsymbol{J}_C(\boldsymbol{q})$ can be described as:

$$\boldsymbol{J}_C(\boldsymbol{q}) = \boldsymbol{\Lambda}\boldsymbol{J}(\boldsymbol{q}) \tag{7}$$

where $\boldsymbol{\Lambda}$ is a matrix that specifies which dimension(s) in the task-space is (are) constrained due to contacts. Substituting Eq. 7 into Eq. 3, we get:

$$\boldsymbol{J}_C(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{\Lambda}\boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} = \boldsymbol{0}$$

or equivalently:

$$\boldsymbol{\Lambda}\dot{\boldsymbol{p}} = \boldsymbol{0} \tag{8}$$

$\boldsymbol{\Lambda}$ is independent of the dimensionality of the configuration space of the robot and independent of the current configuration of the robot. However, the number of independent rows depends on the number of independent constraints.

In the example of Fig. 1, where the constraint is in the z direction of the end effector (approaching axis for the manipulator), $\boldsymbol{\Lambda}$ would have the form:

$$\boldsymbol{\Lambda} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \tag{9}$$

such that:

$$\boldsymbol{\Lambda}\dot{\boldsymbol{p}} = \boldsymbol{\Lambda}\begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{y}} \\ \dot{\boldsymbol{z}} \end{bmatrix} = \dot{\boldsymbol{z}} = 0 \tag{10}$$

i.e. the end-effector velocity along the z-axis is null.

In real-world tasks involving contacts, in general it will be non-trivial to compute $\boldsymbol{\Lambda}$. However, we know that $\boldsymbol{\Lambda}\dot{\boldsymbol{p}} = \boldsymbol{0}$ and equivalently, $\dot{\boldsymbol{p}}^T\boldsymbol{\Lambda}^T = \boldsymbol{0}$, from Eq. 8. Thus $\boldsymbol{\Lambda}^T$ is the solution to the homogeneous system:

$$\dot{\boldsymbol{p}}^T\boldsymbol{\Lambda}^T = \boldsymbol{0} \tag{11}$$

Let $\boldsymbol{B}_{\dot{\boldsymbol{p}}^T}$ be a set of observed $\dot{\boldsymbol{p}}^T$, *i.e.*, the end-effector velocities where the end-effector is in contact with the constrained surface, then the solution set of the homogeneous system in Eq. 11 can be found by computing the right null space of $\boldsymbol{B}_{\dot{\boldsymbol{p}}^T}$, using singular value decomposition:

$$\boldsymbol{B}_{\dot{\boldsymbol{p}}^T} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T \tag{12}$$

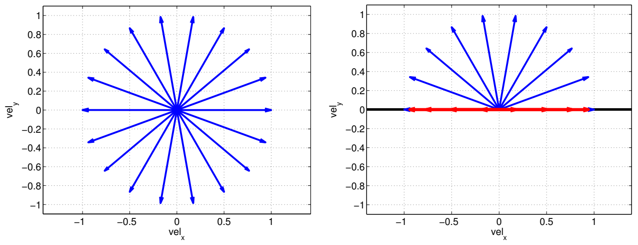
where $\boldsymbol{U}$ is the matrix of left singular vectors, $\boldsymbol{S}$ is a diagonal matrix such that $\boldsymbol{S}_{i,i}$ is the i^{th} largest singular value, and $\boldsymbol{V}$ is a matrix of the right singular vectors. $\boldsymbol{\Lambda}^T$ can then be computed by taking the columns of $\boldsymbol{V}$ with corresponding singular values smaller than a threshold value ϵ.

A. Exploration

To compose this $\boldsymbol{B}_{\dot{\boldsymbol{p}}^T}$, we propose to perform an exploration when the robot is in contact with the environment. When the robot is in contact, the observed task-space velocities $\dot{\boldsymbol{p}}^{obs}$ will be different from the expected $\dot{\boldsymbol{p}}^{exp}$ since the constraint restricts some subspace of the task space. Hence, there are two possibilities for the expected $\dot{\boldsymbol{p}}^{exp}$ and the observed $\dot{\boldsymbol{p}}^{obs}$:

1) in the free motion subspace, *i.e.* when the motion is not in the direction of any contact: $\dot{\boldsymbol{p}}^{exp} = \dot{\boldsymbol{p}}^{obs}$;
2) in the constrained motion subspace, *i.e.* when the motion is in the direction of any contact: $\dot{\boldsymbol{p}}^{exp} \neq \dot{\boldsymbol{p}}^{obs}$.

We collect a set $\dot{\boldsymbol{p}}^{obs}$ into $\boldsymbol{B}_{\dot{\boldsymbol{p}}^T}$ from the latter case and use the method based on Eq. 12 to estimate the selection matrix $\boldsymbol{\Lambda}$. Fig. 2a shows an example of exploration with 18 directions sampling velocities in the 2D Cartesian space. When in contact with a horizontal surface (black line), some Cartesian velocities cannot be performed. Fig. 2b shows the set $\dot{\boldsymbol{p}}^{exp} \neq \dot{\boldsymbol{p}}^{obs}$ in red, which go into $\boldsymbol{B}_{\dot{\boldsymbol{p}}^T}$, and the set of $\dot{\boldsymbol{p}}^{exp} = \dot{\boldsymbol{p}}^{obs}$ in blue.



(a) (b)

Fig. 2: An example of exploration with 18 directions. Fig. 2a shows commanded velocities $\dot{\boldsymbol{p}}^{exp}$, and Fig. 2b shows the resulting velocities $\dot{\boldsymbol{p}}^{obs}$, which are observed when a horizontal surface (shown in black) constrains the robot's motion. In Fig. 2b, blue vectors denote observed velocities which match the expected velocities, while red vectors denote observed velocities which are different from the expected velocities. In this case $\boldsymbol{B}_{\dot{\boldsymbol{p}}^T}$ is composed from the red vectors.

B. Unilateral Constraints

Projected dynamics [21] typically assume bilateral constraints, *i.e.* motion is blocked in both positive and negative directions of the constraint vector. However unilateral constraints are very common in real applications, *i.e.* constraints that limit motion in only one of these directions. For example, the constraint in Fig. 1 limits the motion of the end effector into the table, but does not constrain motion away from the table. Aghili and Su [22] recently proposed an extension to projected dynamics to handle unilateral constraints and friction. Here, we also extend our formulation to unilateral constraints. We model unilateral constraints as:

$$\boldsymbol{\Lambda}\dot{\boldsymbol{p}} \geq \boldsymbol{0} \tag{13}$$

To find $\boldsymbol{\Lambda}$ for unilateral constraints, we follow the same method as in the previous subsection, using the set of $\dot{\boldsymbol{p}}^{obs}$ which do not match $\dot{\boldsymbol{p}}^{exp}$. However, we introduce an additional check on the remaining observed motions which are equal to those commanded, *i.e.* $\dot{\boldsymbol{p}}^{exp} = \dot{\boldsymbol{p}}^{obs}$. In particular, we have to enforce:

$$\boldsymbol{\Lambda}\dot{\boldsymbol{p}} \geq \mathbf{0}, \quad \forall \dot{\boldsymbol{p}} \;\mid\; \dot{\boldsymbol{p}}^{\mathbf{exp}} = \dot{\boldsymbol{p}}^{\mathbf{obs}} \tag{14}$$

If this test holds false, then the sign of $\boldsymbol{\Lambda}$ is changed. This is particularly useful once the exploration phase is over and the robot can resume performing some other desired task. Specifically, for each Cartesian command $\dot{\boldsymbol{p}}_{cmd}$, if $\boldsymbol{\Lambda}\dot{\boldsymbol{p}}_{cmd} \geq \mathbf{0}$, then the command lies in the free motion subspace and can be executed as-is. On the other hand, if $\boldsymbol{\Lambda}\dot{\boldsymbol{p}}_{cmd} < \mathbf{0}$, then the command sent to the robot has to be projected using $\boldsymbol{P}(\boldsymbol{q})$ as computed in Eq. 4.

C. Exploration in the Joint Space

The previous analyses were carried out in the Cartesian space, however it is possible to derive equivalent equations in the joint space. In particular, we define the constraints as:

$$\boldsymbol{\Lambda}^{q}(\boldsymbol{q})\dot{\boldsymbol{q}} = \mathbf{0} \tag{15}$$

In this case $\boldsymbol{\Lambda}^{q}(\boldsymbol{q})$ can be regarded not only as a selection matrix, but also a constraint Jacobian. Moreover, $\boldsymbol{\Lambda}^{q}(\boldsymbol{q})$ is dependent on the configuration of the robot, in contrast to Eq. 8. Explorative movements can also be defined directly in the joint space. Similar to the Cartesian case, there are two possibilities for the expected $\dot{\boldsymbol{q}}^{exp}$ and the observed $\dot{\boldsymbol{q}}^{obs}$:

1) in the free motion subspace: $\dot{\boldsymbol{q}}^{exp} = \dot{\boldsymbol{q}}^{obs}$;
2) in the constrained motion subspace: $\dot{\boldsymbol{q}}^{exp} \neq \dot{\boldsymbol{q}}^{obs}$.

We collect a set of $\dot{\boldsymbol{q}}^{obs}$ from the latter case, and use singular value decomposition to estimate the selection matrix $\boldsymbol{\Lambda}^{q}(\boldsymbol{q})$.

IV. Results

This section presents results from a number of experiments with both real and simulated robots, and demonstrates how our proposed method is able to successfully estimate a variety of different kinds of constraint. First, we show examples of Cartesian space exploration, and evaluate how performance changes with various amounts of observation noise. Understanding the degree of robustness to noise is extremely important, because such observations are likely to be noisy in real applications. We next show examples of joint-space exploration, with an example task of using such exploration to detect the unilateral constraints imposed by joint limits. Later, we show another example of a robot tasked with following a circular trajectory. The robot detects a surface that blocks the trajectory, but is able to continue following the trajectory after exploring and estimating that constraint surface. Finally, we report on two experiments conducted on our bi-manual platform Boris, Fig. 1, using one of Boris' KUKA LWR IV arms to detect and estimate a horizontal surface and an inclined surface.

Exp.	Ang. Error	Real $\boldsymbol{\Lambda}$	Estimated $\boldsymbol{\Lambda}$
1	0°	$[-0.6870, -0.7267]$	$[-0.6870, -0.7267]$
2	0°	$[0.0840, -0.9965]$	$[0.0840, -0.9965]$
3	0°	$[0.5194, 0.8545]$	$[0.5194, 0.8545]$
4	0°	$[0.7787, -0.6274]$	$[0.7787, -0.6274]$
5	0°	$[0.9758, 0.2185]$	$[0.9758, 0.2185]$
6	0°	$[0.8388, 0.5444]$	$[-0.8388, -0.5444]$
7	0°	$[0.9894, -0.1454]$	$[0.9894, -0.1454]$
8	0°	$[-0.1445, 0.9895]$	$[-0.1445, 0.9895]$
9	0°	$[0.6178, 0.7864]$	$[-0.6178, -0.7864]$
10	0°	$[-0.4631, -0.8863]$	$[0.4631, 0.8863]$

TABLE I: Results of simulations on 10 different constraint surfaces in a 2D environment. For each experiment, the table reports true and estimated $\boldsymbol{\Lambda}$ and the error (in degrees) of the estimated constraint surface inclination angle. Opposite signs in the estimated constraints are highlighted in blue.

A. Cartesian exploration without observation noise

We begin with a simplified example, in order to clearly illustrate to the reader how our basic method works. We ran 10 simulations in a 2D environment, assuming frictionless interaction between the end-effector and constraining surface, and assuming perfect proprioception with zero observation noise. Each simulation corresponded to a setup similar to Fig. 2a, but with the constraint surface tilted at a different angle in each of the ten trials. Table I shows that the estimated constraint directions perfectly match the true constraint directions in every trial, however sometimes the sign is reversed (experiments 6, 9 and 10). In these cases, despite the sign difference, the estimated and true constraints have identical null space, thus the estimated projector and the true projector $\boldsymbol{P}(\boldsymbol{q})$ are also identical. Information on the sign can be recovered using Eq. 14. Nevertheless, the information about the sign is lost in $\boldsymbol{P}(\boldsymbol{q})$, due to its definition.

B. Robustness of Cartesian exploration to observation noise

In this section we explore how the accuracy of constraint estimation degrades with increasing amounts of observation noise during proprioceptive sensing. To analyse this performance degradation, we added normally distributed noise, of various magnitudes, to the observed velocity vectors used for constraint estimation. We conducted 100 experiments for each level of noise, with nine different (progressively larger) normally distributed noise levels, ranging from σ = 0.1 up to σ = 0.9 in terms of observed velocity noise magnitude (as compared to true/commanded velocity magnitude). In all experiments, the simulated 2D robot was tasked with exploring the same 2D constraint vector $[0, 1]$.

Fig. 3 plots how constraint estimation errors (and error spread) increase with respect to increases in proprioceptive noise magnitude. These results suggest that, given a fixed number of exploratory movements, constraint estimation errors increase linearly with proprioceptive noise magnitude. However note that, even with very high levels of noise, a correspondingly high number of exploratory robot movements should still be able to recover an accurate estimate of the constraint surface. Additional experiments, to explore how errors decrease with increased amounts of exploration, will be a subject of ongoing and future research.

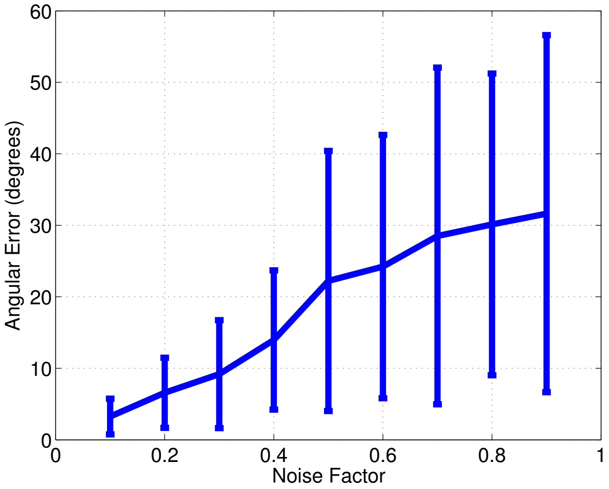


Fig. 3: Plot showing how errors in the estimation of the constraint surface (angle of surface inclination) increase with magnitude of observation noise (erroneous observed velocity vectors). Results suggest that constraint estimation errors increase linearly with observation errors, as expected.

C. Joint-space exploration to detect joint limits

As well as the Cartesian task-space method evaluated in section IV-B above, section III-C showed how exploration and estimation of constraints can also be performed in the configuration (joint angle) space of the robot. Unlike the Cartesian approach, the joint space approach estimates a configuration-dependent constraint, and is therefore particularly well suited to explorative estimation of the robot's joint limits. Fig. 4 illustrates a simulation experiment, in which a 3DOF planar arm is tasked with performing a circular trajectory. When a joint limit is hit (detected by observed joint angles deviating from commanded angles), an exploration in the joint space is triggered and our approach successfully understands which joint has hit its limit. In this example, our method returns an estimated constraint in the form $[1, 0, 0]$, indicating that joint 1 has hit its upper bound.

D. Adapting a task to overcome a detected constraint surface

Estimated constraints can be used to project the task motion onto the free motion subspace. After completing exploration and constraint estimation, if $\boldsymbol{\Lambda}\dot{\boldsymbol{p}}_{cmd} \leq \mathbf{0}$, velocity commands $\dot{\boldsymbol{q}}_{cmd}$ are projected using $\boldsymbol{P}(\boldsymbol{q})$, thus sending modified commands $\boldsymbol{P}(\boldsymbol{q})\dot{\boldsymbol{q}}_{cmd}$ to the robot's motors. Fig. 5 shows an example of such behaviour. The robot begins performing a circular trajectory task, but then collides with a horizontal surface. This constraint is then estimated through exploration. Once the constraint surface has been estimated, the robot resumes performing the commanded circular trajectory. During contact situations, the robot checks whether the commands lie in the free motion space or not. If not, such commands are projected using $\boldsymbol{P}(\boldsymbol{q})$. This modifies the trajectory to one which is as close as possible to the commanded trajectory, given the environmental constraint. Fig. 5 plots the resulting motion.

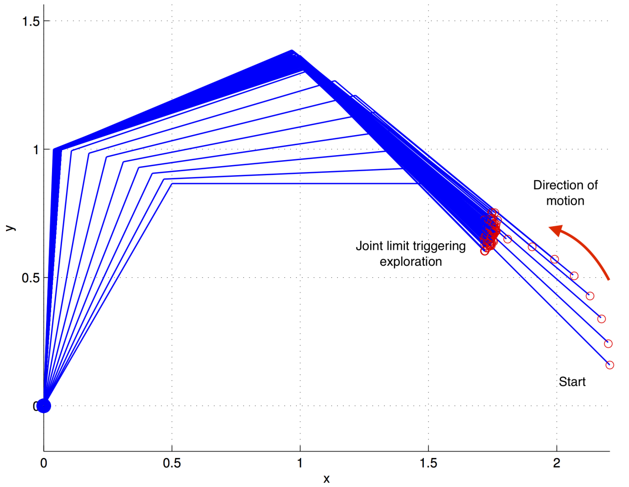


Fig. 4: Estimation of a joint limit by performing exploratory motions in the joint space. The robot starts executing a circular motion task, and then a deviation from the commanded trajectory is detected when the robot hits a limit on its base joint. The robot then quits the commanded task, and performs a series of exploratory motions defined in the robot's joint space. Finally the constraint is correctly estimated. In this example, the estimated constraint is $\boldsymbol{\Lambda}^q(\boldsymbol{q}) = [1, 0, 0]$, which means that joint 1 has hit its upper bound.

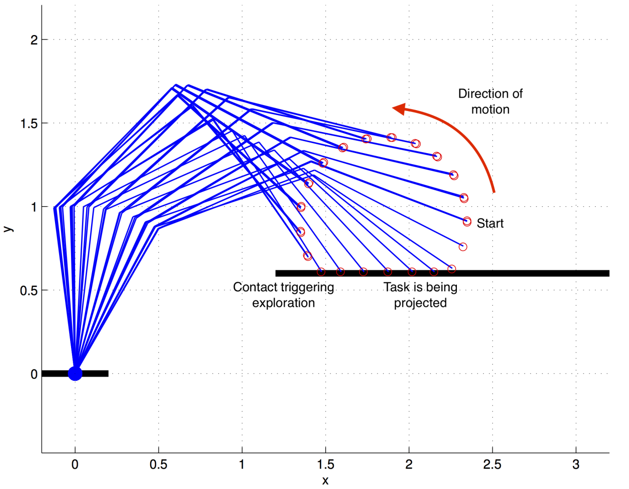


Fig. 5: Example of adapting a commanded trajectory to overcome detected constraints. The robot modifies a commanded circular trajectory, by projecting commanded velocities via $\boldsymbol{P}(\boldsymbol{q})$, to comply as closely as possible with the commanded trajectory, while respecting the detected horizontal surface.

E. Experiments with a real compliant robot arm

This section reports results of experiments using our bi-manual half-humanoid robot Boris. Boris is tasked with detecting and characterising constraints caused by a white-board, firstly when the board lies flat in the horizontal plane, Fig 1, and secondly when it is inclined at an angle of 47 degrees with respect to horizontal, Fig 6. In both cases, the whiteboard and the materials that support it are significantly flexible. This flexibility is unmodelled, thereby presenting significant noise in the observed end-effector velocities. In particular, this flexibility allows a significant amount of perpendicular end-effector motion into the board surface.

In each experiment, Boris performed 64 exploratory motions in the 3D space. Each exploratory motion comprised a 5 cm movement of the end-effector, *i.e.* the sphere of exploration had a 5 cm radius, and the azimuthal and altitudinal angles of rotation (wrt the tool-space) were sampled 8 times each. A contact situation was detected whenever observed motions differed from commanded motions, and relative end-effector velocity vectors were selected for the estimation. Note, for proof of principle the computation of surfaces from the selected observations was performed offline. The video which accompanies this paper shows Boris' explorative motions on both surfaces.

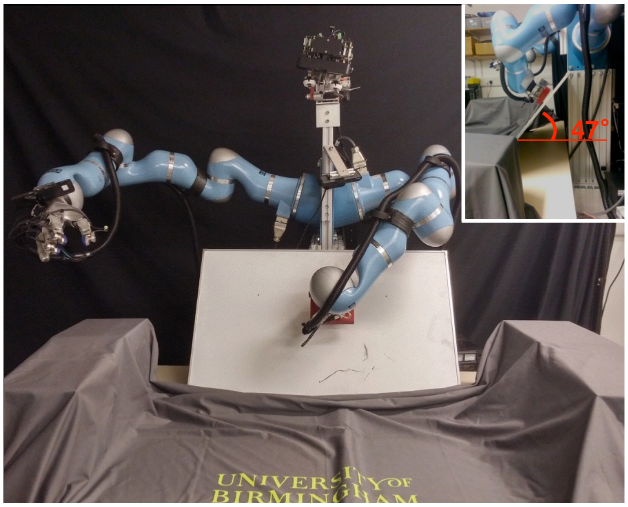


Fig. 6: Boris uses proprioception to explore a whiteboard surface inclined at an angle of 47 degrees to the horizontal. Top right inset, side view showing the angle of inclination of the constraining surface.

For each surface, we performed 10 experiments (each comprising 64 exploratory motions). The surface inclination angles were estimated with mean errors of $2.4484°$ ($\sigma = 1.0921°$) for the horizontal surface, and $4.1772°$ ($\sigma = 2.0684°$) for the $47°$ inclined surface. For the horizontal and inclined surfaces respectively: Tables II and III report angular errors and estimated $\mathbf{\Lambda}$ for all 10 experiments on each surface; Figs. 7 and 9 illustrate the estimated $\mathbf{\Lambda}$ vectors for all ten trials as red arrows, and the true $\mathbf{\Lambda}$ as a blue arrow; Fig. 8 and Fig. 10 show an example set of observed end-effector velocities for one of the ten experiments for each surface. Note that, for both experiments, the largest components of these velocity vectors lie in the planes of the contact surfaces (as expected), however these observed velocity vectors also have a small but significant component in the direction orthogonal to the contact surface. This is due to unmodelled flexibility of the whiteboard and its supporting materials, which allow the robot to move slightly in the constrained direction.

Experiment	Angular Error	Estimated $\mathbf{\Lambda}$
1	$3.0960°$	$[-0.0458, 0.0286, 0.9985]$
2	$2.1548°$	$[-0.0349, 0.0140, 0.9993]$
3	$2.6676°$	$[-0.0464, 0.0031, 0.9989]$
4	$2.8165°$	$[-0.0485, -0.0081, 0.9988]$
5	$2.8139°$	$[-0.0487, 0.0062, 0.9988]$
6	$2.6441°$	$[0.0453, 0.0087, -0.9989]$
7	$0.8245°$	$[-0.0117, -0.0084, 0.9999]$
8	$1.6986°$	$[-0.0296, -0.0005, 0.9996]$
9	$1.1103°$	$[-0.0112, -0.0158, 0.9998]$
10	$4.6582°$	$[-0.0795, 0.0168, 0.9967]$

TABLE II: Estimated $\mathbf{\Lambda}$ and error in estimated surface inclination angle (in degrees) for a horizontal contact surface with true $\mathbf{\Lambda} = [0, 0, 1]$.

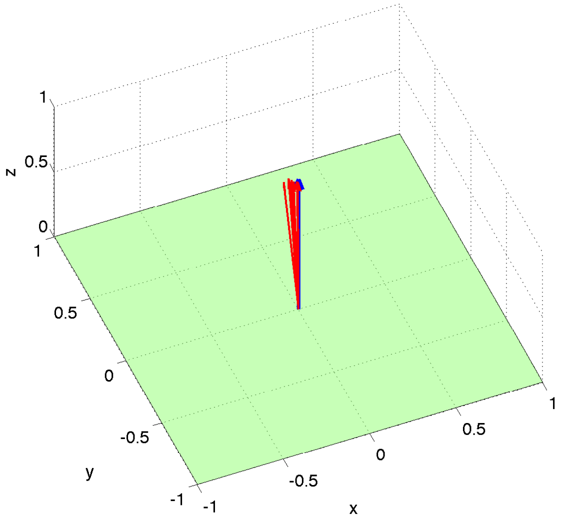


Fig. 7: Visualisation of results for the horizontal contact surface. Blue arrow represents the true $\mathbf{\Lambda}$ vector, and the green plane is the contact surface. Estimated $\mathbf{\Lambda}$, for each of ten trials, are represented by red arrows.

F. Discussion of results

We deliberately designed the experiments with the real robot to test our method in non-ideal conditions, *i.e.* a flexible, deforming contact surface. The results suggest that our method performs robustly in these circumstances, with mean errors of $2.4°$ for the horizontal surface, and $4.2°$ for the inclined surface. Note that, due to the highly deformable supporting structure and materials, as well as flexibility of the whiteboard itself, the true angles of inclination of these surfaces would not have been the same as those measured prior to the robot making contact (horizontal in the first

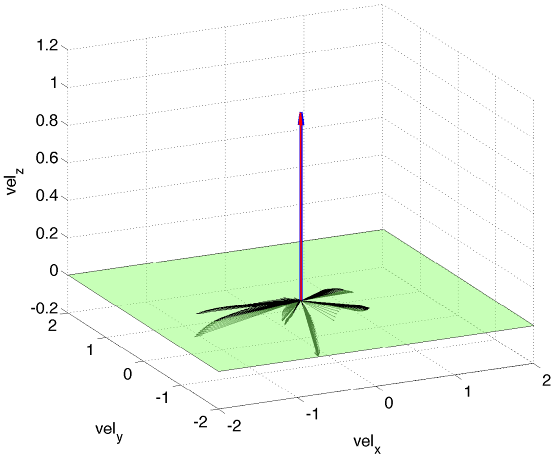


Fig. 8: Observed velocity vectors (black arrows), estimated $\mathbf{\Lambda}$ (red arrow), and true $\mathbf{\Lambda}$ (blue arrow) from experiment 7 on the horizontal contact surface. Note that some velocity vectors have components in the negative z direction, due to the flexibility of the whiteboard which allows the end effector of the robot to move slightly in the negative z direction.

Experiment	Angular Error	Estimated $\mathbf{\Lambda}$
1	$3.1708°$	$[-0.0509, -0.7155, 0.6968]$
2	$5.8918°$	$[0.0726, -0.7770, 0.6253]$
3	$5.5162°$	$[-0.0941, -0.7413, 0.6645]$
4	$5.3110°$	$[0.0441, -0.7837, 0.6196]$
5	$6.4421°$	$[0.0839, -0.7776, 0.6232]$
6	$5.9987°$	$[-0.0986, -0.7038, 0.7035]$
7	$4.8269°$	$[0.0324, -0.7817, 0.6228]$
8	$2.2875°$	$[-0.0129, -0.7565, 0.6538]$
9	$0.5057°$	$[0.0082, -0.7335, 0.6797]$
10	$1.8208°$	$[0.0175, 0.7490, -0.6623]$

TABLE III: Estimated $\mathbf{\Lambda}$ and error in estimated surface inclination angle (in degrees) for an inclined ($47°$) contact surface with true $\mathbf{\Lambda} = [0, -0.7314, 0.6820]$.

experiment, and inclined at $47°$ in the second experiment). It is therefore possible that the estimation errors with respect to the real surfaces, deformed by the robot pushing on them during the experiments, may actually be substantially less than the figures reported above.

In comparison, [18] reports smaller errors of less than $1°$ in the angles of inclination estimated by their quadrupedal trotting robot. However: their robot trots on a completely rigid flat surface, as compared to our flexible and deformable whiteboard structure; and they make use of optical force sensors on all four feet to detect contacts, which are not available in our case. Additionally note that [18] exploit multiple contacts on the surface. In contrast, our method can work with as little as one contact trajectory in the 2D case (similar to the trotting robot's surface that has only a pitch angle with respect to the ground plane, with no roll or yaw angles). In the case of a truly 3D surface of arbitrary orientation, our method can work with as little as two (non-parallel) observed contact trajectories.

The quadrupedal robot in [18] is assumed to be trotting on a uniformly flat surface with constant slope, and all four feet must be trotting on the same slope. In contrast, our method could be extended to more complex surfaces, where the constraints can be modeled as:

$$\mathbf{\Lambda}(\boldsymbol{p})\dot{\boldsymbol{p}} = \mathbf{0} \tag{16}$$

where the dependency of the constraints on the position in space is explicit. In such case, $\mathbf{\Lambda}(\boldsymbol{p}_\mathbf{i})$ could be estimated by performing a local exploration around the point $\boldsymbol{p}_i$.

For the planar surfaces addressed in this paper, we chose the vector of the $\boldsymbol{V}$ matrix associated to the smallest singular value. However, if each contact provides constraints limiting

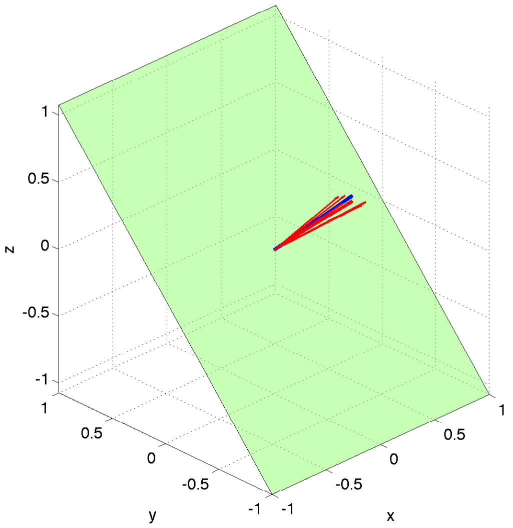


Fig. 9: Visualisation of results for exploring the inclined surface. Blue arrow represents the true $\mathbf{\Lambda}$ vector, and the green plane is the contact surface. Estimated $\mathbf{\Lambda}$, for each of ten trials, are represented by red arrows.

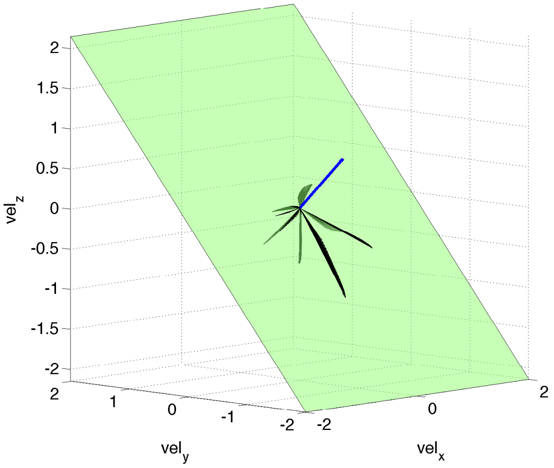


Fig. 10: Observed velocity vectors (black arrows), estimated $\mathbf{\Lambda}$ (red arrow), and true $\mathbf{\Lambda}$ (blue arrow) from experiment 9 on the inclined contact surface. Note that some velocity vectors have components in the perpendicular direction penetrating the whiteboard surface. This is due to the flexibility of the whiteboard which allows the end effector of the robot to move slightly in the penetrating direction into the surface.

motion in more than one direction, *i.e.* contact with a non-planar surface, this can be detected by the fact that multiple singular values, in the decomposition of the set of observed velocities, will be very close to 0.

Finally, our method is not limited to end-effector contacts only. In principle, our method could be extended to include multiple contacts on different parts of the robot body, having one $\mathbf{\Lambda}$ per contact and devising explorative motions for different parts of the robot, *e.g.* end effector and elbow.

V. Conclusions and suggestions for future work

This paper presented a method to estimate kinematic constraints due to contacts, without the need for force-torque or tactile sensors at the contact points, or other sensing modalities such as vision. Our method is based only on kinematics, and works by computing the null space of a set of observed velocity vectors which differ from commanded velocities during explorative motions. Additionally, we showed how to extend this method to handle unilateral constraints, and showed how an equivalent formulation, defined in the joint space, is convenient for applications such as explorative estimation of a robot's joint limits.

We demonstrated the effectiveness of our method in a number of simulations, where we also explored how the accuracy of constraint estimations are affected by noisy observations of velocity vectors. We also conducted two sets of experiments with our bi-manual half-humanoid robot Boris, showing how it is possible to reliably detect and estimate parameters for constraining contact surfaces positioned at different angles of inclination with respect to horizontal. The constraints were successfully estimated, with mean errors of $2.4°$ for the horizontal surface, and $4.2°$ for the inclined surface, even though the surface itself was flexible and deformed significantly during contacts.

Future work will address the problem of defining quicker and smarter sets of explorative actions, where we will also reduce the extent of space that needs to be covered during exploration. We will also implement and demonstrate extensions of this method to handle non-planar surfaces. The questions of how to reason about, parameterise, characterise, and explore highly deformable or soft surfaces, remain open research problems.

VI. Acknowledgements

This work was supported by: UK Nuclear Decommissioning Authority bursary 8030110; H2020 RoMaNS 645582, H2020 CogIMon 644727, FP7 CoDyCo 600716; EPSRC UK-Korea Civil Nuclear Collaboration EP/M026477/1. We thank Mr. Joshua Smith for his help in the laboratory.

References

[1] V. Ortenzi, M. Adjigble, J. A. Kuo, R. Stolkin, and M. Mistry, "An experimental study of robot control during environmental contacts based on projected operational space dynamics," in *14th IEEE-RAS International Conference on Humanoid Robots, Humanoids 2014, Madrid, Spain, November 18-20, 2014*, 2014, pp. 407–412.

[2] V. Ortenzi, R. Stolkin, J. A. Kuo, and M. Mistry, "Projected inverse dynamics control and optimal control for robots in contact with the environment: A comparison," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[3] S. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, vol. 307, no. 5712, pp. 1082–1085, 2005.

[4] L. U. Odhner, L. P. Jentoft, M. R. Claffee, N. Corson, Y. Tenzer, R. R. Ma, M. Buehler, R. Kohout, R. D. Howe, and A. M. Dollar, "A compliant, underactuated hand for robust manipulation," *The International Journal of Robotics Research*, vol. 33, no. 5, pp. 736–752, 2014.

[5] P. J. Besl and R. C. Jain, "Invariant surface characteristics for 3d object recognition in range images," *Comput. Vision Graph. Image Process.*, vol. 33, no. 1, pp. 33–80, Jan. 1986.

[6] J. Biswas and M. Veloso , "Depth camera based indoor mobile robot localization and navigation," in *Proceedings of IEEE International Conference on Robotics and Automation*. IEEE, May 2012, pp. 1697–1702.

[7] D. Schiebener, J. Morimoto, T. Asfour, and A. Ude, "Integrating visual perception and manipulation for autonomous learning of object representations." *Adaptive Behaviour*, vol. 21, no. 5, pp. 328–345, 2013.

[8] K. Duncan, S. Sarkar, R. Alqasemi, and R. V. Dubey, "Multi-scale superquadric fitting for efficient shape and pose recovery of unknown objects," in *2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, May 6-10, 2013*, 2013, pp. 4238–4243.

[9] A. Bicchi, J. K. Salisbury, and D. L. Brock, "Contact sensing from force measurements," *International Journal of Robotics Research*, vol. 12, pp. 249–262, 1993.

[10] T. Tsujimura and T. Yabuta, "Object detection by tactile sensing method employing force/torque information," *IEEE Transactions on Robotics and Automation*, vol. 5, pp. 444 – 450, 1989.

[11] R. S. Fearing and T. O. Binford, "Using a cylindrical tactile sensor for determining curvature," *IEEE Transactions on Robotics and Automation*, vol. 7, pp. 806 – 817, 1991.

[12] R. D. Howe, "Tactile sensing and control of robotic manipulation," *Journal of Advanced robotics*, vol. 8, no. 3, pp. 245–261, 1994.

[13] J. D. Schutter, H. Bruyninckx, S. Dutré, J. D. Geeter, J. Katupitiya, S. Demey, and T. Lefebvre, "Estimating first-order geometric parameters and monitoring contact transitions during force-controlled compliant motion," *I. J. Robotic Res.*, vol. 18, no. 12, pp. 1161–1184, 1999.

[14] T. Lefebvre, H. Bruyninckx, and J. D. Schutter, "Polyhedral contact formation modeling and identification for autonomous compliant motion," *IEEE Trans. Robotics and Automation*, vol. 19, no. 1, pp. 26–41, 2003.

[15] N. Sommer, M. Li, and A. Billard, "Bimanual compliant tactile exploration for grasping unknown objects," in *2014 IEEE International Conference on Robotics and Automation, ICRA 2014, Hong Kong, China, May 31 - June 7, 2014*, 2014, pp. 6400–6407.

[16] M. Björkman, Y. Bekiroglu, V. Hogman, and D. Kragic, "Enhancing visual perception of shape through tactile glances," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, November 3-7, 2013*. IEEE, 2013, pp. 3180–3186.

[17] C. Zito, M. S. Kopicki, R. Stolkin, C. Borst, F. Schmidt, M. A. Roa, and J. L. Wyatt, "Sequential trajectory re-planning with tactile information gain for dexterous grasping under object-pose uncertainty." in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 4013–4020.

[18] C. Gehring, C. D. Bellicoso, S. Coros, M. Blösch, P. Fankhauser, M. Hutter, and R. Siegwart, "Dynamic trotting on slopes for quadrupedal robots." in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5129–5135.

[19] T. Yoshikawa and A. Sudou, "Dynamic Hybrid Position/Force Control of Robot Manipulators: On-Line Estimation of Unknown Constraint," *Robotics and Automation, IEEE Transactions on*, 1990.

[20] H.-C. Lin, M. Howard, and S. Vijayakumar, "Learning null space projections." in *ICRA*. IEEE, 2015, pp. 2613–2619.

[21] 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.

[22] F. Aghili and C. Su, "Control of constrained robots subject to unilateral contacts and friction cone constraints," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, May 2016, pp. 2347 – 2352.