Exploiting Distributed Tactile Sensors to Drive a Robot Arm Through Obstacles

Alessandro Albini, Francesco Grella, Perla Maiolino, *Member, IEEE*, and Giorgio Cannata

***Abstract*—Robots operating in unstructured environments must be capable of safely handling unexpected collisions with objects existing in the surrounding area, possibly without stopping the task execution. This letter proposes a tactile feedback control law allowing the robot to apply bounded contact forces in reaction to physical collisions while performing a task. As a use-case scenario, the problem of driving a robot arm through obstacles to reach a known target position in the space is considered. Intensities and locations of multiple and simultaneous contacts between the robot and the environment are detected using large area tactile sensors covering the robot body. It will be shown that the robot is capable of controlling the end-effector position and of reacting to unexpected collisions by regulating the interaction forces applied to the environment. The method has been validated on a Baxter robot partially covered with 3495 distributed tactile elements, operating in an unknown environment.**

***Index Terms*—Tactile sensors, robot control, motion control.**

I. Introduction

ROBOTS have always executed operations with high-accuracy in well-constrained manufacturing environments. However, when the environment is complex or cluttered, the problem of performing autonomous tasks becomes harder. This topic has been widely investigated for various classes of robots including: mobile robots [1], robot manipulators [2]–[4] and snake robots [5]. The majority of the proposed approaches requires to know a model of the environment or to build one (typically using vision) to avoid direct contacts with obstacles. Conversely, it has also been demonstrated that, for snake robots locomotion [5], obstacles can, in some cases, be exploited to accomplish the task. The availability of a model of the environment allows to plan the robot actions, but when it is not known a priori and vision cannot be used, a strategy based on planning is not possible and a collision would be hard to avoid. Therefore, robots must be capable of safely reacting to unpredictable contact events while performing tasks. Force/torque sensors feedback can be used to retrieve contact forces between the robot and the external world and to properly control the robot motions [6]–[10]. However, when the robot gets in contact with multiple obstacles, force/torque sensors cannot fully capture the complexities of the physical interaction. Indeed, information about multiple contacts and internal forces would be lost. A large area tactile system (i.e., *robot skin*) is more suitable in this case. These systems are composed of a large number of pressure transducers (i.e., *taxels*) distributed over the robot body [11]–[14]. Differently from force/torque sensors, they allow to retrieve the position and the intensity of each single force acting on the robot surface. Such information is essential to properly control the robot in complex unstructured scenarios.

The contribution of this letter is to present a tactile-based closed-loop control law that allows the robot to react to unexpected collisions and to apply safe contact forces while performing a task with the end-effector. Multiple contacts, sensed through the robot skin, are fed back to the robot controller to compute suitable joint velocities minimizing the squared sum of contact forces applied on the robot arm. With respect to the current literature (see Section II), the proposed approach is purely tactile-based and does not require: (i) joint level or any other form of force/torque feedback; (ii) the adoption of a joint level torque controller, since the control law operates at kinematic level; (iii) an accurate robot dynamic model describing the interactions with the environment. The reference task consists of moving the robot end-effector to a specific target position. Fig. 1 shows a test case scenario. What proposed in this letter is relevant to address any task that can be seen as an extension or a generalization of a reaching problem that must be performed by the robot in highly unstructured environments. Indeed, applications related to cooperative assembly, navigation, exploration, etc., require robots capable of reaching or following a particular target without damaging humans or objects. Example scenarios are related to assembling and inspection processes in automotive and aircraft manufacturing (e.g., fixing a wire harness to the body of a car in automotive plant or sealing) , robotic surgery (e.g., robot arm accesses internal organs, natural orifices and performs surgery) or decommissioning in nuclear environments (e.g., the robot needs to navigate into vessels and pipes).

The letter is organized as follows. Section II provides an overview of the literature related to the design of robot reactive control laws. Approaches based on joint level and/or tactile sensing are reviewed and the differences and our contribution

Manuscript received October 15, 2020; accepted February 18, 2021. Date of publication March 23, 2021; date of current version April 9, 2021. This letter was recommended for publication by Associate Editor Y. Mezouar and Editor E. Marchand upon evaluation of the reviewers' comments. This work was supported by the European Community's Framework Programme Horizon 2020 Grant Agreement 820767, Project CoLLaboratE. *(Corresponding author: Alessandro Albini.)*

Alessandro Albini and Perla Maiolino are with the Oxford Robotics Institute, University of Oxford, Oxford OX1 2JD, U.K. (e-mail: alessandro@robots.ox.ac.uk; perla@robots.ox.ac.uk).

Francesco Grella and Giorgio Cannata are with the Department of Informatics, Bioengineering, Robotics and Systems Engineering, University of Genoa, Genova, 16145 GE, Italy (e-mail: francesco.grella@edu.unige.it; giorgio.cannata@unige.it).

This letter has supplementary downloadable material available at https://doi.org/10.1109/LRA.2021.3068110, provided by the authors.

Digital Object Identifier 10.1109/LRA.2021.3068110

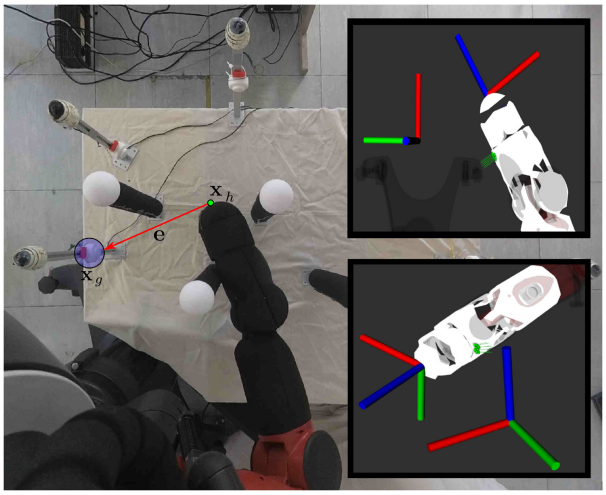


Fig. 1. A robot is trying to reach a target position $\mathbf{x}_g$ while controlling the interaction forces applied on the environment. The vector $\mathbf{e}$ represents the error at a given time instant between the target and the actual position of the end-effector $\mathbf{x}_h$. The RViz screenshots show the tactile sensors involved in the contact and their normals as well as the robot hand and goal frames from two different point of views.

with respect to the current literature are explained. Section III formally describes the problem and the assumptions that have been made. In Section IV, the control objectives are defined. The control law allowing the robot to interact with the environment is described in Section V. Section VI shows the experimental setup, while the results are discussed in Section VII. Conclusion follows.

II. State of the Art

The problem of detecting collisions and regulating the contact forces applied by the robot to the environment or humans has been mainly addressed by exploiting joint level measurements [6], [7], [15] or cameras [16]. In [7] and [6], torque feedback is used to recognize impacts with an obstacle or to discriminate between collisions with humans and purposive interactions. Both approaches switched the control law when the collision was correctly detected. In particular, [7] focused on regulating the interaction forces between the arm and the obstacle. The goal of [6] was instead to provide a safe human-robot interaction. Rather than switching the control action, different approaches dealt with the problem of preserving a task execution during a physical interaction [15], [16]. In these works, a redundant robot is controlled to react when a collision is detected. The remaining degrees of freedom are then used to execute a different task. Such a method leverages the null space projection to prioritize a robot safe behaviour and to execute (if possible) a secondary control objective. In [15], impacts are detected with joint torque sensors, whereas [16] uses depth cameras. Differently from [15] and [16], the letter presented in [17] considers more than two tasks and deals with the problem of dynamically changing the task hierarchy while the robot is in contact with the environment.

The work proposed in [18] presented a control architecture for collision detection and reaction requiring motor currents and joint positions feedback only. Furthermore, the robot is controlled by commanding joint velocity reference signals, making this approach particularly suitable for industrial robots.

Although force/torque sensors or more in general joint level measurements have been proved to be valid for a wide range of applications related to safety or collision reaction, they cannot be used to fully capture or describe complex contact events. As previously mentioned, they have two major limitations. Firstly, internal forces cannot be detected. Secondly, they cannot be used to precisely estimate the locations in multi-contact scenarios. In this respect, it is worth mentioning that [19] showed the possibility of estimating the positions of up to three simultaneous contact forces in simulation using joint torque feedback. However, the performance of the algorithm drops if noise is present or if the model of robot dynamics is not accurate enough.

Researchers started taking advantage of tactile sensors to sense the contact forces or to precisely retrieve their locations, thus proving the importance of tactile feedback when performing exploration tasks [20] or when designing control strategies for grasping and manipulation tasks [21], [22]. Differently from methods relying on proprioceptive sensing, the literature based on the use of robot skin feedback to design reactive or compliant control laws to address contacts occuring on the whole robot body is not so wide. Indeed, the majority of the works exploiting tactile sensors only considers contacts at the end-effector. This is due to the fact that robot skin technologies covering the entire robot body are mostly prototypes and therefore their usage is restricted to a small number of research groups in the world. However, the few works exploiting such technology to deal with physical contacts showed the advantages of using large-area tactile feedback. In these works [23]–[28], robot skin is used to retrieve contact information. Then, joint torques are commanded to properly control the interaction at the contact point or to make the robot compliant during a collision.

In [23], a contact force task is performed using the robot forearm. The proposed experiments showed that the contact location retrieved using robot skin is essential to improve the performance while controlling forces. In the work proposed in [24], the inverse dynamics of the iCub robot is learned by correlating the raw tactile sensor readings with joint torques. Then, a control torque is computed allowing the robot to compensate for an accidental contact with an object while following a trajectory with the end-effector. The parallel force/position control has been exploited in [25] together with robot skin. In particular, authors showed that using tactile feedback it is possible to execute motion trajectories at the contact point by precisely controlling both reference velocities and force profiles during the physical interaction. In [26] and [27], authors proved that tactile and joint torque sensing can be used together to move a robot in a cluttered environment. They proposed a control architecture consisting of a model predictive controller on top of an impedance controller ensuring robot compliance when in contact with obstacles. In [28], multi-contact interactions detected using a multi-modal robot skin are translated into robot motions to achieve a whole robot body compliance with an omnidirectional mobile manipulator.

In addition to robot skin, the aforementioned techniques require at least the availability of a very accurate model of the robot dynamics and the access to the low-level torque controller. These assumptions could not be always satisfied [18] (e.g., for industrial robots). Torque sensing or torque control capabilities can be hard to implement in already existing robots. Nevertheless, tactile sensing technologies can be integrated afterwards [29]. The method proposed in this letter only relies on robot skin. Indeed, differently from works where the robot is controlled to be compliant when a contact occurs, the idea here is to use the tactile feedback to compute a set of joint velocities that minimizes the squared sum of forces acting on the robot links. Using this approach, there is no need to model the robot dynamics and, since the robot joints are commanded at velocity level, the access to the low-level torque controller is not strictly required. Then, similarly to what proposed in [15] and [16], the robot redundancy is used to execute a secondary goal, which in this context corresponds to the reaching task.

III. Problem Description

The task considered in this letter consists of controlling the motion of the robot end-effector to reach a target point. The goal is located at a *known* position $\mathbf{x}_g \in \Re^3$ in a cluttered region of the space surrounding the robot. The position $\mathbf{x}_g$, as well as all the vectors defined in the rest of the letter, are expressed with respect to a common reference frame, which is assumed to be the robot base. The robot does not have any a priori knowledge of the position of the obstacles in the scene and does not use visual information. During the movement towards the target, it can get in contact with the obstacles with any part of its body (see Fig. 1).

It is assumed that the interaction forces can be controlled by using tactile feedback provided by a distributed tactile sensing system integrated all over the robot arm. Such a system is composed of taxels which are spatially [30] and force calibrated [31], allowing the robot to sense the intensities and locations of contact forces acting on its body. Each taxel provides a measurement describing a lumped force acting on it. Within the context of this letter, it is assumed that the robot surface in contact with the surrounding environment has a negligible curvature. Then, by considering a connected region of taxels involved in a contact (i.e., composed of a set of adjacent taxels jointly stimulated), the equivalent force applied on the region is computed as the resultant of the forces sensed by each taxel and applied to the centroid of the area. This operation can be performed with any robot skin system based on discrete tactile sensing, including the many technologies proposed in the literature (e.g., [11]–[14]). Therefore, the proposed approach can be assumed to be independent from the specific hardware implementation described in Section VI. Similarly to what proposed in [26], the following assumptions have been made: i) slow robot movements have been considered in order to neglect inertial effects; ii) friction effects are ignored; iii) the effects of a force applied on the contact centroid $\mathbf{x} \in \Re^3$ can be modeled as:

$$\dot{\mathbf{f}} = -k\left(\mathbf{n}\mathbf{n}^{\intercal}\right)\dot{\mathbf{x}} \tag{1}$$

where $\dot{\mathbf{x}} \in \Re^3$ is the velocity of the contact centroid, $\mathbf{n} \in \Re^3$ is the normal to the robot surface at the contact centroid and $k > 0$ is the elastic constant. Equation (1) represents the continuous time equivalent of the discrete time contact model proposed in [26] and [27].

IV. Control Objectives Definition

The problem of reaching a known target point $\mathbf{x}_g$ from a generic position $\mathbf{x}_h$ is usually tackled by applying a control law that minimizes a Lyapunov function of the form:

$$V_g = \frac{1}{2}\mathbf{e}^{\intercal}\mathbf{e} \tag{2}$$

where $\mathbf{e} = \mathbf{x}_h - \mathbf{x}_g$ is the position error.

Let us suppose the target is not moving and there are no obstacles in contact with the robot arm. The minimization of the previous Equation can be achieved by imposing the following control law [32]:

$$\dot{\mathbf{x}}_g = -\gamma_g \mathbf{e} \tag{3}$$

with $\gamma_g > 0$. The Cartesian command can be transformed in joint velocities for the robot with:

$$\dot{\mathbf{q}} = -\gamma_g \mathbf{J}_h^{\#}\mathbf{e} \tag{4}$$

where $\mathbf{J}_h$ is the Jacobian matrix associated to the point $\mathbf{x}_h$ and $(\cdot)^{\#}$ is the pseudo-inverse operation, possibly properly regularized if the robot is close to singular configuration.

In this specific context, the robot links can get in contact with the environment (see Fig. 1). Then, beyond the problem of controlling the motion of the point $\mathbf{x}_h$, the interaction forces arising on the robot arm must also be properly maintained within safety limits. Thus, it is proposed to modify the control law in Equation (3) by adding the goal of minimizing the interaction forces during the robot motion.

It is assumed that, at a given time instant, the robot arm is in contact with the environment with one or more links. For each connected region of tactile sensors involved in the contact, the following quantities can be computed exploiting the robot skin feedback: $\mathbf{f}_j$ representing the j-th force acting on the robot arm; $\mathbf{n}_j$ the normal to the robot surface at the contact centroid $\mathbf{x}_j$.

Then the minimization of the following function is considered:

$$V_f = \frac{1}{2}\sum_{j=1}^{M} f_j^2 \tag{5}$$

where $f_j = \mathbf{n}_j^{\intercal}\mathbf{f}_j \in \Re$ represents the component of the j-th force acting on the robot body, while M is the number of contacts applied at a generic time instant. The minimization of Equation (5) corresponds to the minimization of the squared sum of forces acting on the robot arm.

Equations (5) and (2) define two tasks, to be (possibly) executed concurrently. Robot redundancy can be used to satisfy multiple control objectives. Furthermore, it has already shown to be effective when exploited for problems of obstacle avoidance and collision reaction [15], [33].

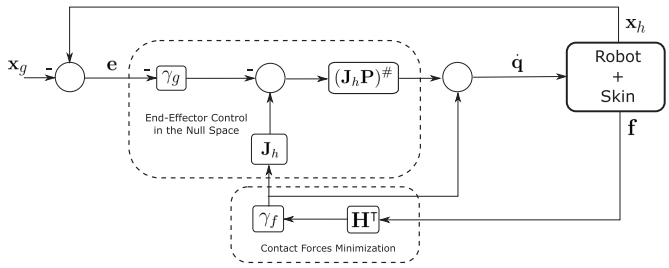


Fig. 2. Control scheme overview: robot skin data are processed to obtain a vector $\mathbf{f}$ containing information about the contact forces between the robot and the environment. The joint level command $\dot{\mathbf{q}}$ is computed by summing: (i) a term that minimizes the squared sum of forces; (ii) a term that moves the robot end-effector towards the target location $\mathbf{x}_g$. In the control scheme, γ_g and γ_f are the gains associated to the position and the force tasks. The robot task Jacobian is defined as $\mathbf{J}_h$, while $\mathbf{H}$ is the matrix defined in Equation (11) and $\mathbf{P}$ is its null space projector.

The contents of the next Section describe how to generate the joint level control signal that minimizes both Equations (5) and (2).

V. Control Law Design

This Section describes how to compute the joint level control action achieving the control tasks previously defined. First, the minimization of Equation (5) is addressed. Then, the task of moving the robot end-effector is considered. An overview of the control scheme is shown in Fig. 2.

A. Contact Forces Minimization

Given a set of forces acting on the robot, to achieve the minimization of Equation (5) the control action must impose the following condition at each time instant:

$$\dot{V}_f = \sum_{j=1}^{M} f_j \dot{f}_j < 0 \tag{6}$$

The time derivative of f_j can be computed as follows:

$$\dot{f}_j = \dot{\mathbf{n}}_j^\intercal \mathbf{f}_j + \mathbf{n}_j^\intercal \dot{\mathbf{f}}_j \tag{7}$$

From Equation (1) it can be seen that $\dot{\mathbf{n}}_j^\intercal \mathbf{f}_j = 0$, since the two vectors are orthogonal. Equation (1), representing the contact model, can be substituted in the expression of $\dot{f}_j$, obtaining:

$$\dot{f}_j = \mathbf{n}_j^\intercal \dot{\mathbf{f}}_j = -k_j \mathbf{n}_j^\intercal \left(\mathbf{n}_j \mathbf{n}_j^\intercal\right) \dot{\mathbf{x}}_j = -k_j \mathbf{n}_j^\intercal \dot{\mathbf{x}}_j \tag{8}$$

That allows to write $\dot{V}_f$ as:

$$\dot{V}_f = -\sum_{j=1}^{M} f_j k_j \mathbf{n}_j^\intercal \dot{\mathbf{x}}_j < 0 \tag{9}$$

The above expression can be written as a function of the joint velocity $\dot{\mathbf{q}}$, by defining $\mathbf{J}_j \in \Re^{3\times 6}$ as the linear jacobian of the contact point $\mathbf{x}_j$:

$$\dot{V}_f = -\sum_{j=1}^{M} f_j k_j \mathbf{n}_j^\intercal \mathbf{J}_j \dot{\mathbf{q}} < 0 \tag{10}$$

Finally, with the following definitions:

$$\mathbf{f} = \begin{bmatrix} f_1 & \dots & f_M \end{bmatrix}^\intercal \in \Re^M$$
$$\mathbf{H} = \begin{bmatrix} k_1 \mathbf{n}_1^\intercal \mathbf{J}_1 \\ \vdots \\ k_M \mathbf{n}_M^\intercal \mathbf{J}_M \end{bmatrix} \in \Re^{M\times 6} \tag{11}$$

it is possible to write $\dot{V}_f$ in a more compact form:

$$\dot{V}_f = -\mathbf{f}^\intercal \mathbf{H} \dot{\mathbf{q}} < 0 \tag{12}$$

The vector $\dot{\mathbf{q}}$ representing the joint velocities can be intended as the joint-level control action. In order to make $\dot{V}_f$ negative, $\dot{\mathbf{q}}$ can be chosen as:

$$\dot{\mathbf{q}} = \gamma_f \mathbf{H}^\intercal \mathbf{f} \tag{13}$$

where $\gamma_f > 0$ is a gain factor.

Equation (13) represents a force feedback closed loop control law minimizing the squared sum of the forces applied on the robot.

B. Motion Towards the Target

The control law in Equation (13) can be used to control the interaction between the robot and the environment. The minimization of Equation (2), representing the reaching control objective, is executed on the null space of the force task. The expression of $\dot{\mathbf{q}}$ becomes:

$$\dot{\mathbf{q}} = \gamma_f \mathbf{H}^\intercal \mathbf{f} + \mathbf{P}\dot{\mathbf{z}} \tag{14}$$

where $\mathbf{P} = [\mathbf{I} - \mathbf{H}^{\#}\mathbf{H}]$. Since $\mathbf{P}$ is an orthogonal projector for $\mathbf{H}$, the minimization of Equation (5) is not affected by the vector $\dot{\mathbf{z}}$, allowing to arbitrarily choose it. Therefore $\dot{\mathbf{z}}$ can be exploited to drive the robot end-effector towards the goal, while it is still in contact with the environment.

From Equation (3), we know that:

$$-\gamma_g \mathbf{e} = \mathbf{J}_h \dot{\mathbf{q}}. \tag{15}$$

Equation (14) can be substituted in the expression above, leading to:

$$-\gamma_g \mathbf{e} = \gamma_f \mathbf{J}_h \mathbf{H}^\intercal \mathbf{f} + \mathbf{J}_h \mathbf{P}\dot{\mathbf{z}} \tag{16}$$

Thus $\dot{\mathbf{z}}$ can be written as:

$$\dot{\mathbf{z}} = (\mathbf{J}_h\mathbf{P})^{\#} \left(-\gamma_g \mathbf{e} - \gamma_f \mathbf{J}_h \mathbf{H}^\intercal \mathbf{f}\right) \tag{17}$$

and then, since $\mathbf{P}(\mathbf{J}_h\mathbf{P})^{\#} = (\mathbf{J}_h\mathbf{P})^{\#}$ the final control law is the following:

$$\dot{\mathbf{q}} = \gamma_f \mathbf{H}^\intercal \mathbf{f} + (\mathbf{J}_h\mathbf{P})^{\#} \left(-\gamma_g \mathbf{e} - \gamma_f \mathbf{J}_h \mathbf{H}^\intercal \mathbf{f}\right) \tag{18}$$

C. Exploiting the End-Effector Orientation

As previously described, the error term $\mathbf{e}$, appearing in Equation (2), is defined in terms of position only. However, the orientation of the end-effector is a free parameter that can be exploited to help the robot get through obstacles.

To this aim, let us define a vector $\mathbf{w}$, passing through the center of the robot end-effector (see Fig. 3). An idea is to

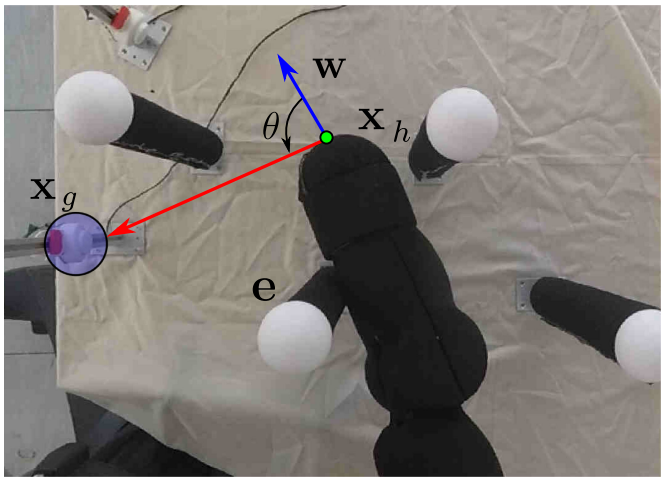


Fig. 3. The angle between $\mathbf{w}$ and $\mathbf{e}$ is minimized, allowing the robot end-effector to rotate around the obstacle.

minimize the angle θ between $\mathbf{w}$ and $\mathbf{e}$. In this way, $\mathbf{w}$ is controlled to point towards $\mathbf{x}_g$, rotating the end-effector around the obstacle. Therefore, the vector $\mathbf{e}$ can be redefined to consider the minimization of the angle θ:

$$\mathbf{e} = \begin{bmatrix} \mathbf{x}_g - \mathbf{x}_h \\ (\mathbf{w} \times \mathbf{e})\,\theta \end{bmatrix} \in \Re^6 \tag{19}$$

where $(\mathbf{w} \times \mathbf{e})$ is a vector representing the rotation axis of the end-effector.

Then Equation (18), representing the joint-level control action, can be rewritten as:

$$\dot{\mathbf{q}} = \gamma_f \mathbf{H}^\intercal \mathbf{f} + (\mathbf{J}_h \mathbf{P})^{\#} \left(-\Gamma_g \mathbf{e} - \gamma_f \mathbf{J}_h \mathbf{H}^\intercal \mathbf{f}\right) \tag{20}$$

where Γ_g is a gain matrix:

$$\Gamma_g = \mathrm{diag}(\gamma_l\ \gamma_l\ \gamma_l\ \gamma_a\ \gamma_a\ \gamma_a) \tag{21}$$

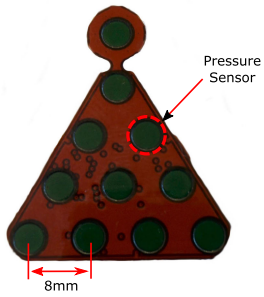
and $\gamma_l > 0$ and $\gamma_a > 0$ are the gains associated to the linear and the angular error respectively.

VI. Experimental Setup

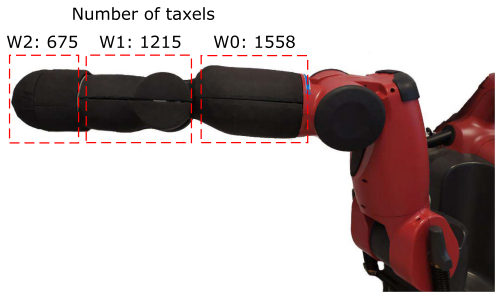
A. Robot Platform

The robot skin technology used to validate the approach is formed by a mesh of interconnected triangular modules [34]. The single module (see Fig. 4(a)) is composed of 11 capacitive transducers with a diameter of 3.5 mm and a pitch of 8 mm. Tactile readings are acquired using a 16-bits (with approximately 2-bits of noise) capacitance to digital converter (CDC) coupled with a microcontroller.

A robot skin mesh composed of 3495 taxels has been integrated on three links of a Baxter robot arm. Fig. 4(b) shows the final platform with more detailed information about distribution of the sensors. Taxels measurements are acquired at 20 Hz while the robot controller runs at 100 Hz. The raw response of the robot skin was calibrated using a technique similar to the one presented in [35], therefore, each taxel measurement can be converted into a normal force value. Furthermore, since the robot skin is spatially calibrated, the location of contact forces acting on the single tactile element is known with respect to the robot body base frame.



(a) Cyskin Module.



(b) Baxter robot used to validate the approach. W0, W1 and W2 are the robot links names.

Fig. 4. The robot skin technology and the robot platform.

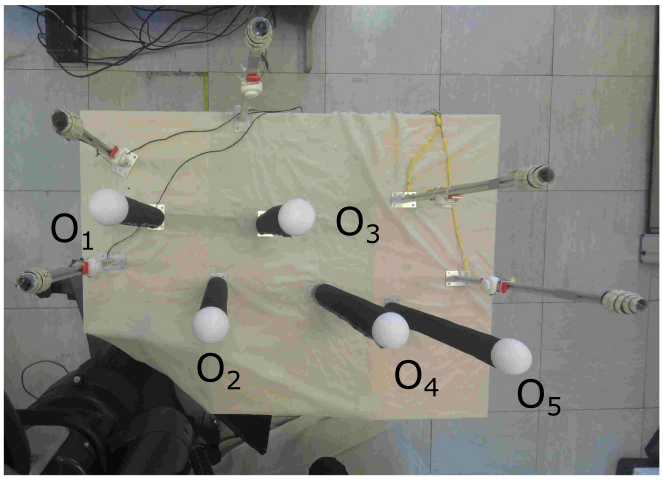
B. Environment Setup

The controller has been validated within the environment showed in Fig. 5. The robot is commanded to reach five target locations with the end-effector. To clearly see the positions of the goals in the images and in the attached video, five switches have been placed on the target locations. Each one of them is connected to a led strip, which will turn on if the switch is pressed.

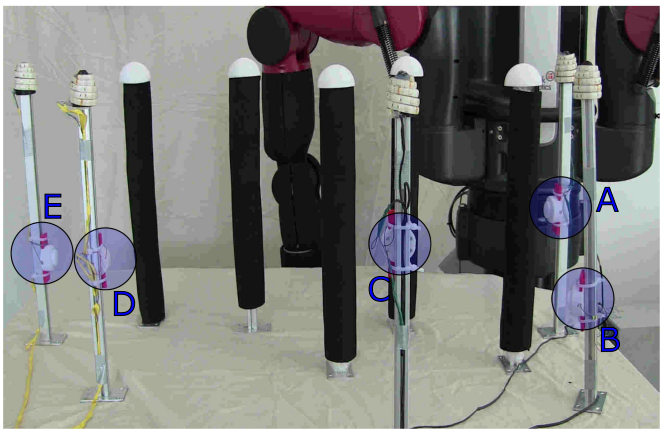
Five obstacles have been placed between the robot and the targets. The obstacles consist of aluminum extruded bars wrapped with an internal layer of pluriball and covered by an external layer of neoprene. Fig. 5 shows a picture of the experimental setup from two different points of view. Locations of obstacles and targets are highlighted in Figs. 5(a) and 5(b). There is no clear path between the robot end-effector and the targets. This means that the robot will collide with at least one obstacle during the motion. The placement of targets and obstacles has been chosen to test the approach on a set of representative contact configurations. Forces can act on a single link (even in opposite directions) or on different links, both on the same side of the robot arm or in opposite directions.

Examples of possible contact configurations are summarized in Fig. 6, where it can also be clearly seen the size of the robot arm compared with the distance among the obstacles.

For each target, the robot can collide in the following ways:



(a) Top view of the experimental setup.



(b) Front view of the experimental setup. Blue circles highlight the position of the targets.

Fig. 5. Experimental setup. There are 5 obstacles and 5 goal positions. Distances among adjacent obstacles: $\|\mathbf{O}_1 - \mathbf{O}_2\| = 30$ cm, $\|\mathbf{O}_1 - \mathbf{O}_3\| = 33$ cm, $\|\mathbf{O}_2 - \mathbf{O}_3\| = 23$ cm, $\|\mathbf{O}_2 - \mathbf{O}_4\| = 34$ cm, $\|\mathbf{O}_3 - \mathbf{O}_4\| = 27.5$ cm and $\|\mathbf{O}_4 - \mathbf{O}_5\| = 23.5$ cm. Diameters of the obstacles: $\mathbf{O}_1 = 5$ cm, $\mathbf{O}_2 = 5.5$ cm, $\mathbf{O}_3 = 6$ cm, $\mathbf{O}_5 = 6.5$ cm and $\mathbf{O}_5 = 5.5$ cm.

- Target **A**: single contact with $\mathbf{O}_2$ or multiple contacts between $\mathbf{O}_2$ and $\mathbf{O}_1$ (Fig. 6(a)).
- Target **B**: multiple contacts with $\mathbf{O}_1$ and $\mathbf{O}_2$ on different links and on the same side of the robot arm (see Fig. 6(b)).
- Target **C**: multiple contacts (both on the same or on different links) on opposite sides with $\mathbf{O}_2$ and $\mathbf{O}_3$ or with $\mathbf{O}_3$ and $\mathbf{O}_4$ (Figs. 6(c) and 6(d)).
- Target **D**: single contact with $\mathbf{O}_4$ or multiple opposite contacts on different links with $\mathbf{O}_5$ and $\mathbf{O}_4$ (Figs. 6(e) and 6(f)).
- Target **E**: single contact with $\mathbf{O}_5$ or multiple opposite contacts (both on the same or on different links) with $\mathbf{O}_4$ and $\mathbf{O}_5$ (Figs. 6(g) and 6(h)).

VII. Results

A. Experiments Description

Since the proposed method is reactive, the posture of the robot arm, as well as the contact configuration at a specific time instant, depends on the initial robot arm configuration. Therefore, the robot was commanded to reach each target location 10 times. At the beginning of each experiment, the robot arm configuration was changed. Thus, a total of 50 reaching attempts were made. A target is considered successfully reached if an absolute position error of 5 mm is achieved. Conversely, the controller was stopped if the end-effector did not proceed further due to a local minimum (a timeout of 5 seconds was set). Furthermore, it was considered to stop the robot if the maximum force, among those applied on the arm, was greater than a threshold F_{thresh} of 15 N. The value of F_{thresh} is selected to avoid damages to the robot or to the experimental setup.

There is only one requirement in the selection of the initial arm configuration: the robot must be able to physically reach the target location. To motivate this requirement, consider for example that the initial position is the one in Fig. 6(h) and the target location is **C** (see Fig. 5). Due to the structure of our controller, the goal **C** cannot be physically reached starting from that position of the end-effector. Indeed, the controller will move the arm towards **C** sliding on $\mathbf{O}_5$. However, the arm is not long enough to reach the target position and the robot will get stuck in a local minimum. To address this kind of local minima, there is the need of a high level planner that pulls out the arm and plans a different starting point for the end-effector. This problem is out of the scope of this letter and represents an argument for a future extension of the work. Here, only the reactive behaviour was evaluated.

In the experiments, obstacles were assumed to have the same stiffness. This is reasonable since they have been built using the same materials. With this assumption, the stiffness value can be extracted from the matrix $\mathbf{H}$ (see Equation (11)), becoming part of the controller gain γ_f defined in Equation (13). The gain was tuned by considering the following aspects. A high value allows to rapidly minimize Equation (5). However, this could lead to very fast and jerky robot motions. On the contrary, a low value leads to smooth but very slow movements, and the robot cannot properly react to contact events. The gain $\gamma_f = 0.05$ was selected to find a trade-off between smoothness and speed of the robot motions.

TABLE I
Summary of the 10 Trials Executed for Each Goal Location Starting From Different Robot Configurations

Goal	Success Rate	$\bar{F}$(N)	F_m(N)	F_M(N)
A	10 / 10	2.92	1.20	5.42
B	10 / 10	3.82	2.46	4.84
C	8 / 10	5.95	1.70	10.53
D	9 / 10	2.53	0.83	4.30
E	10 / 10	3.42	1.06	5.06

B. Results and Discussion

The results are shown in Table I. The success rate for each target location, along with a summary of the forces applied among the 10 trials, is provided. In Table I, $\bar{F}$ is the average of the maximum force applied during each trial, F_m is the smallest maximum force applied and F_M is the maximum force

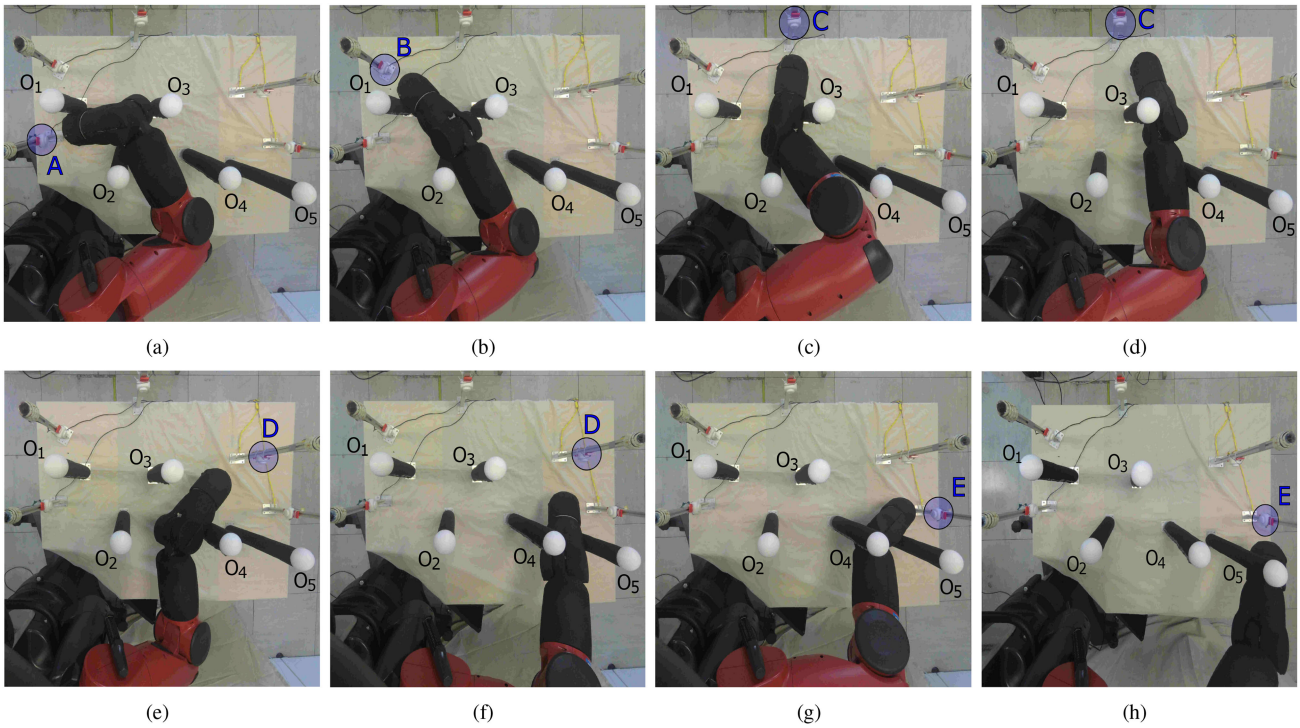


Fig. 6. The robot is moving with the goal of reaching several target positions represented by the blue circles in the images. This image shows a set of possible contact configurations.

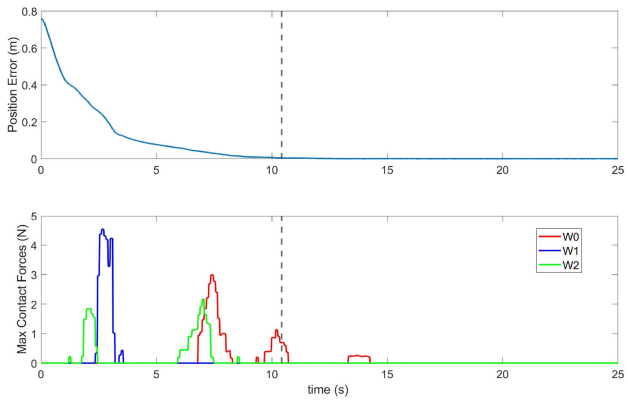


Fig. 7. Trends of the position error and forces while reaching the target **B** (see Fig. 6(b)). The second plot shows the maximum force applied at each time instant to each robot link. The robot links names are the same defined in Fig. 4(b). The vertical line represents the time instant where the position error is ≤ 5 mm. After the initial impacts, the controller manages to regulate the contact forces and to accomplish the reaching task. In this case, the robot is still in contact with one of its links when the desired accuracy is reached. The remaining part of the plot (after the vertical line) shows that the control law brings the forces to zero and additional contacts due to small robot motions are still managed while maintaining the desired reaching accuracy.

applied over the 10 trials. These statistics are computed for the experiments where the target was successfully reached. Then, an example showing the trends of the position error and forces during a single experiment is given in Fig. 7.

Results show that, with the proposed controller, the robot addresses the possible contact configurations described in Section VI-B and it can reach the targets in almost all the experiments performed. The safety threshold F_{thresh} was never violated and failed attempts occurred for local minima. It must be noted that in these local minima the robot stops its motion applying bounded contact forces. To clarify this point, let us consider the minimization of Equation (2) alone and the corresponding control law given by $\dot{\mathbf{q}} = -\gamma_g \mathbf{J}^{\#}\mathbf{e}$. Local minima of Equation (2) are due to the motion constraints imposed by the obstacles. Without taking into account the minimization of Equation (5), the robot could end up in a local minimum, pushing against the obstacles and applying an equilibrium force related to $\|\gamma_g \mathbf{e}\|$ which is not explicitly controllable and potentially dangerous. On the contrary, with the proposed solution exploiting the null space projection, contact forces are regulated by Equation (13), while Equation (2) is minimized at best. As a consequence, although the robot could end up on local minima where the reaching task is not satisfied, the proposed method still allows to properly control the interactions with the environment.

It can be seen from Table I that two of the failed attempts occurred while trying to reach the target **C**, where forces acted on opposite sides of the robot arm (see Fig. 6). There was also a failure while reaching the target **D**. In that specific case, the robot could not proceed any further since the joint shoulder reached its limit.

In summary, the proposed technique allows the robot to address different types of contacts. However, depending on the force displacement and on the arm configuration, there could be multi-contact scenarios that are hard to be addressed by purely relying on a reactive control law. A solution would be to design a planner on top of the proposed method that computes strategies to address local minima.

VIII. Conclusion

In this letter, a low-level control law allowing a robot manipulator to get through obstacles was presented. The interaction forces between the robot and the environment are controlled exploiting distributed tactile sensors placed over the robot arm. No model of the environment or an estimation of it is required. Furthermore, the method does not require an accurate model of the manipulator dynamics and not even joint torque sensing or the access to the robot low-level torque controller. The proposed technique could be useful in all the contexts where a purely reaction-based robot behaviour is required to prevent damage to the environment (or to the robot) or injuries to humans. The method was validated on a real robot partially covered with tactile sensors. In the experiments, the robot was commanded to perform 50 reaching attempts starting from different configurations of the arm. The obtained results showed that the proposed method allows to deal with complex contact conditions and therefore it can be used to control the robot in unstructured environments. Although there are still limitations related to local minima that are hard to be completely addressed with a purely reactive strategy, the proposed method allows to end up on local minima configurations where contact forces are properly controlled. The design of a high-level planning algorithm to be built on top of the reactive level is currently under development.

References

[1] M. Hoy, A. S. Matveev, and A. V. Savkin, "Algorithms for collision-free navigation of mobile robots in complex cluttered environments: A survey," *Robotica*, vol. 33, no. 3, pp. 463–497, 2015.

[2] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in proceedings," *Proc. IEEE Int. Conf. Robot. Automat.*, vol. 2, Mar. 1985, pp. 500–505.

[3] A. Hornung, M. Phillips, E. G. Jones, M. Bennewitz, M. Likhachev, and S. Chitta, "Navigation in three-dimensional cluttered environments for mobile manipulation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2012, pp. 423–429.

[4] S. Chitta, E. G. Jones, M. Ciocarlie, and K. Hsiao, "Perception, planning, and execution for mobile manipulation in unstructured environments," *IEEE Robot. Automat. Mag.*, vol. 19, no. 2, pp. 58–71, Jun. 2012.

[5] F. Sanfilippo, J. Azpiazu, G. Marafioti, A. A. Transeth, Ø. Stavdahl, and P. Liljebäck, "A review on perception-driven obstacle-aided locomotion for snake robots," in *Proc. 14th Int. Conf. Control, Automat., Robot. Vis.*, 2016, pp. 1–7.

[6] S. Haddadin, A. Albu-Schaffer, A. De Luca, and G. Hirzinger, "Collision detection and reaction: A contribution to safe physical human-robot interaction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 3356–3363.

[7] A. de Luca and R. Mattone, "Sensorless robot collision detection and hybrid force/motion control," in *Proc. Proc. IEEE Int. Conf. Robot. Automat.*, Apr. 2005, pp. 999–1004.

[8] T. Matsumoto and K. Kosuge, "Collision detection of manipulator based on adaptive control law," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, vol. 1, 2001, pp. 177–182.

[9] V. Perdereau and M. Drouin, "A New Scheme for Hybrid Force-Position Control," in *RoManSy* 9, A. Morecki, G. Bianchi, and K. Jaworek, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 150–159.

[10] F. Romano *et al.*, "The codyco project achievements and beyond: Toward human aware whole-body controllers for physical human robot interaction," *IEEE Robot. Automat. Lett.*, vol. 3, no. 1, pp. 516–523, Jan. 2018.

[11] T. Someya, T. Sekitani, S. Iba, Y. Kato, H. Kawaguchi, and T. Sakurai, "A large-area, flexible pressure sensor matrix with organic field-effect transistors for artificial skin applications," *Proc. Nat. Acad. Sci.*, vol. 101, no. 27, pp. 9966–9970, 2004.

[12] G. Cannata, M. Maggiali, G. Metta, and G. Sandini, "An embedded artificial skin for humanoid robots," in *Proc. IEEE Int. Conf. Multisensor Fusion Integration Intell. Syst.*, Aug. 2008, pp. 434–438.

[13] P. Mittendorfer and G. Cheng, "Humanoid multimodal tactile-sensing modules," *IEEE Trans. Robot.*, vol. 27, no. 3, pp. 401–410, Jun. 2011.

[14] Y. Ohmura, Y. Kuniyoshi, and A. Nagakubo, "Conformable and scalable tactile sensor skin for curved surfaces," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2006, pp. 1348–1353.

[15] A. De Luca and L. Ferrajoli, "Exploiting robot redundancy in collision detection and reaction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2008, pp. 3299–3305.

[16] E. Magrini and A. De Luca, "Human-robot coexistence and contact handling with redundant robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 4611–4617.

[17] M. Liu, R. Lober, and V. Paodois, "Whole-body hierarchical motion and force control for humanoid robots," *Auton Robot 40*, pp. 493–504, 2016, doi: 10.1007/s10514-015-9513-5.

[18] M. Geravand, F. Flacco, and A. De Luca, "Human-robot physical interaction and collaboration using an industrial robot with a closed control architecture," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2013, pp. 4000–4007.

[19] L. Manuelli and R. Tedrake, "Localizing external contact using proprioceptive sensors: The contact particle filter," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.,* vol. 10 2016, pp. 5062–5069.

[20] N. Sommer and A. Billard, "Multi-contact haptic exploration and grasping with tactile sensors," *Robot. Auton. Syst.*, vol. 85, pp. 48–61, 2016.

[21] Z. Kappassov, J.-A. Corrales, and V. Perdereau, "Touch driven controller and tactile features for physical interactions," *Robot. Auton. Syst.*, vol. 123, 2020, Art. no. 103332.

[22] Q. li, C. Schuermann, R. Haschke, and H. Ritter, "A control framework for tactile servoing," vol. 6, 2013, doi: 10.15607/RSS.2013.IX.045.

[23] A. Del Prete, F. Nori, G. Metta, and L. Natale, "Control of contact forces: The role of tactile feedback for contact localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 4048–4053.

[24] R. Calandra, S. Ivaldi, M. P. Deisenroth, and Peters, "Learning torque control in presence of contacts using tactile sensing from robot skin," in *Proc. IEEE-RAS 15th Int. Conf. Humanoid Robots*, Nov. 2015, pp. 690–695.

[25] S. Denei, F. Mastrogiovanni, and G. Cannata, "Towards the creation of tactile maps for robots and their use in robot contact motion control," *Robot. Auton. Syst.*, vol. 63, pp. 293–308, 2015.

[26] A. Jain, M. D. Killpack, A. Edsinger, and C. C. Kemp, "Reaching in clutter with whole-arm tactile sensing," *Int. J. Robot. Res.*, vol. 32, no. 4, pp. 458–482, 2013.

[27] M. D. Killpack, A. Kapusta, and C. C. Kemp, "Model predictive control for fast reaching in clutter," *Auton. Robots*, vol. 40, no. 3, pp. 537–560, 2015.

[28] Q. Leboutet, E. Dean-Leon, F. Bergner, and G. Cheng, "Tactile-based whole-body compliance with force propagation for mobile manipulators," *IEEE Trans. Robot.*, vol. 35, no. 2, pp. 330–342, Apr. 2019.

[29] A. Schmitz, P. Maiolino, M. Maggiali, L. Natale, G. Cannata, and G. Metta, "Methods and technologies for the implementation of large-scale robot tactile sensors," *IEEE Trans. Robot.*, vol. 27, no. 3, pp. 389–400, Jun. 2011.

[30] A. Albini, S. Denei, and G. Cannata, "Towards autonomous robotic skin spatial calibration: A framework based on vision and self-touch," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2017, pp. 153–159.

[31] P. Maiolino, M. Maggiali, G. Cannata, G. Metta, and L. Natale, "A flexible and robust large scale capacitive tactile system for robots," *IEEE Sensors J.*, vol. 13, no. 10, pp. 3910–3917, Oct. 2013.

[32] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*, 1st ed. Springer Publishing Company, Incorporated, 2008.

[33] A. Cherubini and F. Chaumette, "A redundancy-based approach for obstacle avoidance in mobile robot navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2010, pp. 5700–5705.

[34] " Cyskin," [Online]. Avilable: https://www.cyskin.com/

[35] J. Kangro, S. Traversaro, D. Pucci, and F. Nori, "Skin normal force calibration using vacuum bags," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2017, pp. 401–406.