

# Online Contact Point Estimation for Uncalibrated Tool Use

Yiannis Karayiannidis, Christian Smith, Francisco E. Viña, and Danica Kragic

**Abstract**—One of the big challenges for robots working outside of traditional industrial settings is the ability to robustly and flexibly grasp and manipulate tools for various tasks. When a tool is interacting with another object during task execution, several problems arise: a tool can be partially or completely occluded from the robot’s view, it can slip or shift in the robot’s hand - thus, the robot may lose the information about the exact position of the tool in the hand. Thus, there is a need for online calibration and/or recalibration of the tool. In this paper, we present a model-free online tool-tip calibration method that uses force/torque measurements and an adaptive estimation scheme to estimate the point of contact between a tool and the environment. An adaptive force control component guarantees that interaction forces are limited even before the contact point estimate has converged. We also show how to simultaneously estimate the location and normal direction of the surface being touched by the tool-tip as the contact point is estimated. The stability of the the overall scheme and the convergence of the estimated parameters are theoretically proven and the performance is evaluated in experiments on a real robot.

## I. INTRODUCTION

The ability to robustly grasp and manipulate tools intended for human use and employ these for various tasks (as in Fig. 1) remains one of the big challenges for robots working outside of traditional industrial settings. The application areas range from flexible industrial robots working with tools intended for human use to domestic service robots that perform household chores [1]. In order to provide the input for the control loop guiding the execution of a task, the knowledge of the position of the tool-tip is necessary.

In contrast to classical industrial or other robots with fixed and a priori known tools, it is not realistic to assume that service robots have precise beforehand calibrations of the positions of the tool-tips. Even if they did, the tool may slip and move relative to the gripper while it is being used. Therefore, there is a need for online calibration and/or recalibration of the position of the tip of the tool the robot is using. Current approaches mostly rely on vision-based methods for calibrating the pose and are therefore not applicable in scenarios where the tools or relevant parts of it are occluded.

In this paper, we present an online tool-tip calibration method that uses force/torque measurements and an adaptive estimation scheme to find the point of contact between a tool and the environment. This estimation can be carried out in real-time while the robot is using the tool to perform some



Fig. 1 : Robot manipulating a tool intended for human use.

arbitrary task, and does not require any predefined model of the shape, size or initial position of the tool being used.

An adaptive force control component guarantees that interaction forces are limited even before the contact point estimate has converged. We also show how to simultaneously estimate the location and normal direction of the surface being touched by the tool-tip as the contact point is estimated.

The paper has the following structure: Section II reviews the state of the art in related work, Section III formalizes the problem in terms of statics and kinematics, Section IV describes the proposed approach and motivates it theoretically, giving stability and convergence proofs, while Section V describes the experimental implementation of the method on a real robot and the experimental results. Finally, conclusions are presented in Section VI.

## II. RELATED WORK

The problem of resolving uncertainties in the end-effector or the tool positioning is well-studied and has been a relevant topic in robotics since the advent of the first manipulators. Early work focused on solving the problem of calibrating the position of the manipulators and end-effectors themselves [2], and this has been extended to also include objects grasped by the robot [3].

Some works treat the problem without explicitly modelling the position of a tool or its point of interaction with the

The authors are with the Computer Vision and Active Perception Lab., Centre for Autonomous Systems, School of Computer Science and Communication, Royal Institute of Technology (KTH), SE-100 44 Stockholm, Sweden. e-mail: {yiankar|ccs|fevb|dani}@kth.se

environment, but focus rather on robust performance of a well-defined task under positioning uncertainties. Examples of this includes work by Bruyninckx *et al.* that estimates the alignment error between the peg and the hole for an insertion task [4], work by Hovland and McCarragher that uses a neural network approach to estimate the contact states between two work pieces [5], and work by Koeppe and Hirzinger that learns the appropriate interaction forces for a peg-in-hole task [6].

Some work treats tool-tip position estimation as a calibration problem that can be performed offline prior to using the tool. Yang *et al.* use an iterative least squares approach to calibrate the relationship between tool-tip position and joint angles, using measurements from a known external reference [7]. Cheah *et al.* use adaptive control for tracking a kinematically uncertain manipulator chain, including a tool grasped by the end-effector, but only estimate the kinematics — the position of the end-effector and the tool are measured externally [3].

Others use a particle filter approach to fit a model to the object pose by collecting measurements from touching the object before grasping it [8], [9]. Hebert *et al.* use a fusion approach with vision, force/torque measurements and proprioception to estimate the position of an object with a known model, held in the end-effector [10]. Păiș *et al.* learn relations between a held object and an external tool it interacts with using gaussian mixture models [11].

Atkeson *et al.* have proposed a method for estimating inertial parameters of a grasped object based on force/torque measurements [12], and Kubus *et al.* have used sensor fusion combining measurements of acceleration, velocities, position, forces, and torques to estimate inertial parameters and principal moments of inertia of a grasped object, fitting parameters to estimate object pose [13]. Both these approaches require free motion in a prespecified trajectory, and can not be applied to estimate the contact point of a tool online as it is being used.

Muto and Shimokura have shown a method for estimating contact points given a known tool fixed in the end-effector, using force measurements [14]. Lei *et al.* propose a method that learns model parameters to estimate the position and external force load on a specific non-rigid grinding tool, using proprioception and force/torque measurements [15].

The literature on vision-based object pose estimation and tracking is far too vast to summarize here, but some examples of tool pose estimation based on computer vision methods include work by Kemp and Edsinger who use visual gradients to detect tool-tip positions [16], Krainin *et al.* who simultaneously build an object model and track it in the robot hand [17], and Beetz *et al.*, who use repeated visual template matching to find the location of a spatula in the robot hand [18].

In our previous work, we have shown how an adaptive estimator, integrating proprioception and force/torque measurements can be used to estimate the location of hinges on doors, or the direction of unconstrained motion for drawers that the robot is opening [19], and to estimate slopes on surfaces

the robot is in contact with [20]. We have also shown how to learn manipulation affordances and slippage behaviors of held objects by using combined measurements of wrist-based force/torque measurements and grasp forces [21].

In the present paper, we build on these ideas and propose an adaptive controller that estimates the point of contact of the tool with the environment, along with estimating the contact surface normal. This enables the robot to execute a task with the held tool while performing the estimation. The proposed controller limits the interaction forces, to avoid damage to the tool or workpiece while estimates are converging. The proposed method uses force/torque measurements from a wrist-mounted sensor, it is inherently online and fast enough to react to changes, and can thus track the contact point even if it moves relative to the robot hand while executing the task.

### III. KINETO-STATICS FORMULATION

Before describing the proposed method for estimating the contact point and the surface normal, we define notation and formalize the relevant first-order differential kinematics and the statics. We assume a system that consists of a robot which performs a task with its tool on a surface; the task requires motion control of the tool contact point and control of the contact forces of the tool on a surface.

#### A. Notation

First, we introduce the following notation and definitions that will be used throughout this paper:

Bold small letters denote vectors and bold capital letters denote matrices.  $\mathbf{I}_\nu$ ,  $\mathbf{O}_\nu \in \mathbb{R}^{\nu \times \nu}$  denote an identity and a square matrix of zeros respectively while  $\mathbf{0}_\nu \in \mathbb{R}^\nu$  denotes a column vector of zeros. Hat  $\hat{\cdot}$  and tilde  $\tilde{\cdot}$  denote estimates and the errors between control variables and their corresponding desired values/vectors respectively.  $\mathbf{R}_a$  denotes a rotation matrix that describes the orientation of a frame with respect to the global frame. A left superscript (e.g.  ${}^a\mathbf{b}$ ) denotes the frame (e.g.  $\{a\}$ ) in which a three-dimensional vector (e.g.  $\mathbf{b} \in \mathbb{R}^3$ ) is expressed, e.g.  ${}^a\mathbf{b} = \mathbf{R}_a^\top \mathbf{b}$ , and it is omitted in case of the global frame. The projection matrix on the orthogonal complement space of a unit three dimensional vector  $\mathbf{a}$  is denoted by  $\bar{\mathbf{P}}(\mathbf{a}) \triangleq \mathbf{I}_3 - \mathbf{a}\mathbf{a}^\top$ .  $\mathbf{S}(\mathbf{b})$  denotes the skew-symmetric matrix produced by  $\mathbf{b} \in \mathbb{R}^3$  to perform a cross product operation with any three-dimensional vector  $\mathbf{a} \in \mathbb{R}^3$  i.e.  $\mathbf{b} \times \mathbf{a} = \mathbf{S}(\mathbf{b})\mathbf{a}$ .

#### B. First Order Differential Kinematics

Consider a robot end-effector equipped with a force/torque sensor on its wrist that is grasping a tool which is in contact with a surface, as shown in Fig. 2. We denote with  $\{e\}$  a frame attached at a kinematically known position of the end-effector (e.g. center of force/torque sensor) denoted by  $\mathbf{p}_e$ . We assume that the contact between the tool and the surface is modeled as a contact point that can slide along the surface when the end-effector is moving. At the contact point position  $\mathbf{p}_c$ , we attach a frame  $\{c\}$  with orientation described by  $\mathbf{R}_c = [\mathbf{n}_c \ \mathbf{o}_c \ \mathbf{a}_c]$ , with  $\mathbf{n}_c$  being a unit vector

which is normal to the surface while  $\mathbf{o}_c$ ,  $\mathbf{a}_c$  being vectors that span the flat surface and can be arbitrarily chosen. Let  $\mathbf{r}$  be the relative position of  $\{e\}$  and  $\{c\}$  defined as follows:

$$\mathbf{r} = \mathbf{p}_c - \mathbf{p}_e \quad (1)$$

Note that  ${}^e\mathbf{r} \triangleq \mathbf{R}_e^\top \mathbf{r}$  is constant if we assume that the end-effector is rigidly grasping the tool. Assuming additionally that only sliding motion is performed i.e.  $\dot{\mathbf{p}}_c = \dot{\mathbf{p}}_e$  then  $\mathbf{r}$  is constant too and the velocity of the end-effector frame is constrained as follows:

$$\mathbf{n}_c^\top \dot{\mathbf{p}}_e = 0 \quad (2)$$

Note that the assumption of pure sliding motion that simplifies the estimation of the surface slope is not necessary for the main objective of this work which is the contact point estimation. Differentiating (1) implies that the contact point velocity is related to the end-effector velocity as follows:

$$\dot{\mathbf{p}}_c = \mathbf{J}_t \begin{bmatrix} \dot{\mathbf{p}}_e \\ \boldsymbol{\omega}_e \end{bmatrix}, \text{ with } \mathbf{J}_t \triangleq \begin{bmatrix} \mathbf{I}_3 & -\mathbf{S}(\mathbf{r}) \end{bmatrix} \in \mathbb{R}^{3 \times 6} \quad (3)$$

being the tool Jacobian matrix and  $\boldsymbol{\omega}_e$  being the end-effector rotational velocity i.e.  $\mathbf{S}(\boldsymbol{\omega}_e) = \mathbf{R}_e \mathbf{R}_e^\top$ . By commanding zero rotational velocity and assuming only sliding motion, we can omit the tool Jacobian when mapping the contact point velocities to the joint space. This means that the first order inverse kinematics are given by:

$$\dot{\mathbf{q}} = \mathbf{J}^+ \begin{bmatrix} \mathbf{u} \\ \mathbf{0}_3 \end{bmatrix} \quad (4)$$

where  $\mathbf{u}$  is a commanded end-effector or contact point velocity control law and  $\mathbf{J}^+$  is the right pseudo-inverse of the end-effector Jacobian with  $\mathbf{J}^+ \triangleq \mathbf{J}^\top (\mathbf{J}\mathbf{J}^\top)^{-1}$ .

### C. Statics

While the end-effector presses with the tool on the surface, the normal force arising (with magnitude denoted by  $f_n$ ) can – in case of rigid contact – be regarded as a Lagrange multiplier of the controlled system associated to the constraint (2). While the contact point is moving along the surface following the motion of the end-effector, tangential forces  $\mathbf{f}_t$  arise owing to dynamic friction components that depend on the sliding velocity of the contact point. The total contact force applied at the contact point is mapped to the end-effector as a wrench consisting of a force vector  $\mathbf{f}_c$  and a torque vector  $\boldsymbol{\tau}_c$ :

$$\mathbf{f}_c = \mathbf{n}_c f_n + \mathbf{f}_t, \quad (5)$$

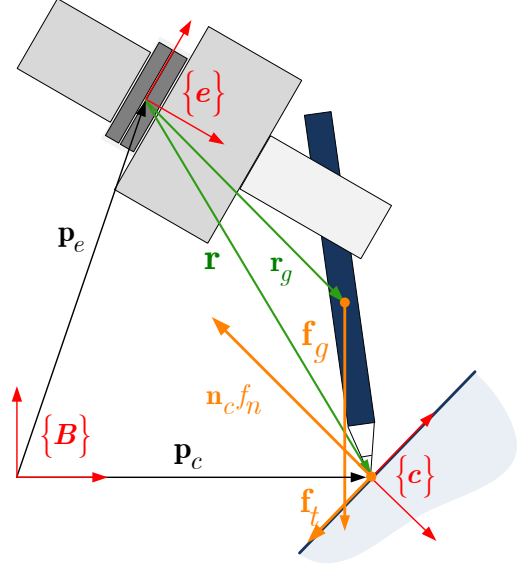
$$\boldsymbol{\tau}_c = \mathbf{r} \times \mathbf{f}_c. \quad (6)$$

The total force  $\mathbf{f}_m$  and torque  $\boldsymbol{\tau}_m$  measured by the force/torque sensor (assuming noise-free measurements, and no acceleration of the end-effector) is given by:

$$\mathbf{f}_m = \mathbf{f}_c + \mathbf{f}_g, \quad (7)$$

$$\boldsymbol{\tau}_m = \mathbf{r} \times \mathbf{f}_c + \mathbf{r}_g \times \mathbf{f}_g, \quad (8)$$

where  $\mathbf{f}_g$  is the gravity force acting at the center of mass of the object and  $\mathbf{r}_g$  is the position of the center of mass



**Fig. 2 :** A robot end-effector equipped with a force/torque sensor on its wrist that is grasping a tool which is in contact with the surface. Frames are illustrated with red lines. Forces are depicted with orange. Absolute position vectors with respect to the base frame  $\{B\}$  are depicted with black lines. Relative positions with respect to the end-effector (sensor) frame are depicted with green lines.

with regard to  $\mathbf{p}_e$ . Note that if  $\mathbf{f}_g$  and  $\mathbf{r}_g$  are known, gravity compensation can be performed by subtracting them from (7) and (8) to obtain  $\mathbf{f}_c$  and  $\boldsymbol{\tau}_c$ , and thus we can use (5) and (6) to identify  $\mathbf{r}$  as proposed in Section IV-A. This gravity compensation can be achieved either by assuming an object with known mass and position with respect to the end-effector, or by considering identification of the gravity effects  $\mathbf{f}_g$  and  $\mathbf{r}_g$  in a prior free-motion phase where the only force acting on the object is inertial, and  $\mathbf{f}_c = \mathbf{0}_3$ . In the latter case, proper rotation of the object by the end-effector can generate signals that can be used in the algorithm in Section IV-A to identify  $\mathbf{r}_g$  to allow compensation for the gravity during the main contact phase. In case of a lightweight tool, such that  $\mathbf{f}_g \ll \mathbf{f}_c$ , we can assume that the effects of gravity are within the limits of the measurement errors, and do not need to be compensated for.

## IV. METHODOLOGY

In this section we propose the adaptive laws for estimating the contact point and the surface orientation as well as the force/motion control which is based on these estimates. The overall control scheme effectiveness is theoretically justified and the formal proofs of the results are given in the Appendix.

### A. Contact Point Estimation

First we design the adaptive law for estimating  ${}^e\mathbf{r}$  assuming that  ${}^e\mathbf{r}$  is piecewise constant or slowly varying compared to the rate of the estimation. An example of an estimation

rate that can be achieved is demonstrated in Section V. The estimates can also be used to estimate  $\mathbf{r} \triangleq \mathbf{R}_e \mathbf{e}\mathbf{r}$  in the global frame. The proposed adaptive law utilizes measurements of forces and torques expressed in the end-effector frame, which is assumed to coincide with the force/torque sensor frame, as this can trivially be achieved through known transformations. The law is given by the following equations:

$${}^e\dot{\hat{\mathbf{r}}} = -\mathbf{\Gamma}_r [\mathbf{L}_r(t){}^e\hat{\mathbf{r}} - \mathbf{c}_r(t)] \quad (9)$$

with

$$\dot{\mathbf{L}}_r = -\beta_r \mathbf{L}_r - \mathbf{S}({}^e\mathbf{f})\mathbf{S}({}^e\boldsymbol{\tau}) \quad \text{with } \mathbf{L}_r(0) = \mathbf{O}_3 \quad (10)$$

$$\dot{\mathbf{c}}_r = -\beta_r \mathbf{c}_r + \mathbf{S}({}^e\mathbf{f}){}^e\boldsymbol{\tau} \quad \text{with } \mathbf{c}_r(0) = \mathbf{O}_3 \quad (11)$$

where  $\mathbf{\Gamma}_r$  is a positive definite matrix affecting the speed of convergence,  $\beta_r$  is a positive design constant acting as forgetting factor and  ${}^e\mathbf{f}$ ,  ${}^e\boldsymbol{\tau}$  are either the measured force and torque after gravity compensation used to estimate  ${}^e\mathbf{r}$  during the contact phase, or the measured force and torque owing to gravity used to estimate the center of mass  ${}^e\mathbf{r}_g$  in the free-motion phase.

*Proposition 1:* The adaptive estimation law (9)-(11) guarantees that:

- (i) the torque estimation error, the estimate  ${}^e\hat{\mathbf{r}}$  and its derivative are bounded i.e.  ${}^e\boldsymbol{\tau} - {}^e\hat{\boldsymbol{\tau}}$ ,  ${}^e\hat{\mathbf{r}}$ ,  ${}^e\dot{\hat{\mathbf{r}}} \in \mathcal{L}_\infty$ ,
- (ii) the torque estimation error and the estimation rate  ${}^e\hat{\mathbf{r}}$  are square integrable, i.e.  ${}^e\boldsymbol{\tau} - {}^e\hat{\boldsymbol{\tau}}$ ,  ${}^e\dot{\hat{\mathbf{r}}} \in \mathcal{L}_2$ ,
- (iii)  $\lim_{t \rightarrow \infty} {}^e\hat{\boldsymbol{\tau}} = {}^e\boldsymbol{\tau}$  and  $\lim_{t \rightarrow \infty} \|{}^e\dot{\hat{\mathbf{r}}}\| = 0$ , and
- (iv) if  $\mathbf{S}({}^e\mathbf{f})$  is persistently excited (PE)  ${}^e\hat{\mathbf{r}}$  converges exponentially to  ${}^e\mathbf{r}$ . By choosing  $\mathbf{\Gamma}_r = \gamma_r \mathbf{I}$ , with  $\gamma_r$  being positive constant, the speed of convergence can be arbitrarily increased by increasing  $\gamma_r$ .

The proposed estimator (9)-(11) is an integral adaptive control law since its design is based on the minimization of an integral cost function of the error between the actual and the estimated torque [22]; the proof of the Proposition 1 is following the proof of the integral adaptive control for identifying the parameters in a multiple inputs-single output parametric model and is based on the use of a quadratic Lyapunov function  $V({}^e\tilde{\mathbf{r}}) = \frac{1}{2} {}^e\tilde{\mathbf{r}}^\top \mathbf{\Gamma}_r^{-1} {}^e\tilde{\mathbf{r}}$ .

As is demonstrated in Section V, convergence to the actual parameters can be achieved by varying the direction of  $\mathbf{f}_e$  in order to span some surface in the Cartesian space, which is an identification condition arising from the problem formulation.

The contact point estimate  $\hat{\mathbf{p}}_c$  can be calculated using proprioception and the estimate  ${}^e\mathbf{r}$  produced by exploiting force/torque measurements:

$$\hat{\mathbf{p}}_c = \mathbf{p}_e + \mathbf{R}_e {}^e\hat{\mathbf{r}} \quad (12)$$

Note that it is also possible to use the contact point estimation together with an accurate model of the grasped tool to determine which of a possible set of points is in contact. In this case, we can additionally infer the orientation of the tool given that the grasping point is obtained through tactile sensing.

The adaptive law can also be used to identify the center of mass in case of free-space motion. The parameters are identified exponentially fast given that  ${}^e\mathbf{f} = \mathbf{R}_e^\top \mathbf{f}$  is PE. Note that  $\mathbf{f} = \mathbf{f}_g$  is constant and thus the identification is excited by the rotational motion of the object.

### B. Surface Normal Estimation

In order to estimate the surface normal direction we design the following adaptive law:

$$\dot{\hat{\mathbf{n}}}_c = -\gamma_n \bar{\mathbf{P}}(\hat{\mathbf{n}}_c) \mathbf{L}_n(t) \hat{\mathbf{n}}_c \quad (13)$$

$$\dot{\mathbf{L}}_n = -\beta_n \mathbf{L}_n + \frac{1}{1 + \|\hat{\mathbf{p}}_e\|^2} \hat{\mathbf{p}}_e \hat{\mathbf{p}}_e^\top \quad \text{with } \mathbf{L}_n(0) = \mathbf{O}_3 \quad (14)$$

where  $\gamma_n$  is a positive constant for tuning the speed of convergence and  $\beta_n$  is a positive forgetting factor.

*Proposition 2:* The adaptive law (13)-(14) guarantees that:

- (i) the norm of the estimate  $\hat{\mathbf{n}}_c(t)$  is invariant i.e. given that  $\|\hat{\mathbf{n}}_c(0)\| = 1$ ,  $\|\hat{\mathbf{n}}_c(t)\| = 1, \forall t > 0$ ,
- (ii) if  $\vartheta(0) \in (-\frac{\pi}{2}, \frac{\pi}{2})$  then  $\vartheta(t) \in (-\frac{\pi}{2}, \frac{\pi}{2}), \forall t > 0$  where  $\vartheta$  is the angle formed between  $\mathbf{n}_c$  and  $\hat{\mathbf{n}}_c$ ,
- (iii)  $\lim_{t \rightarrow \infty} \|\hat{\mathbf{n}}_c\| = 0$ , and
- (iv) if  $\hat{\mathbf{p}}_e$  is persistently excited (PE)  $\vartheta$  converges to zero exponentially which implies that  $\hat{\mathbf{n}}_c$  converges exponentially to  $\mathbf{n}_c$  with a rate that can be tuned by  $\gamma_n$ .

The proposed estimator (13)-(14) is an integral adaptive control – in contrast to those used in our previous work [19], [20]– with normalized input but here is modified in order to produce unit and well-defined estimates of the normal direction as the problem in hand requires. The proof of Proposition 2 can be found in the Appendix, and is based on defining the Lyapunov function in the domain of the *estimation error angle*  $\vartheta$  formed between  $\mathbf{n}_c$  and  $\hat{\mathbf{n}}_c$ .

Measurements of the contact force  $\mathbf{f}_c$  alone cannot in general be used together with (5) to identify the surface normal if the contribution from the tangential force  $\mathbf{f}_t$  due to friction is unknown. However, we can use the force measurements in order to initialize the proposed estimator when contact is detected i.e.  $\hat{\mathbf{n}}_c(0) = \mathbf{f}(0)/\|\mathbf{f}(0)\|$ . Given that the gravity is compensated in  $\mathbf{f}(0)$ , the initial angle error will be within the cone of friction which implies that  $|\vartheta(0)| < \pi/2$  and consequently that the estimator is properly initialized. If there is no rotational motion of the end-effector, the sliding velocity of the tool-tip on the surface is equal to the end-effector velocity, and thus the latter can be directly used to estimate the surface normal direction, independent of the accuracy of the contact point estimate.

### C. Force/motion Control

The control objective is to follow a velocity trajectory  $\mathbf{v}_d(t)$  and to press upon the surface with a desired force  $\mathbf{f}_d$ . In this way, we can perform a meaningful task and simultaneously generate signals  ${}^e\mathbf{f}$  and  $\hat{\mathbf{p}}_e$ . In particular, the motion along the surface not only generates  $\hat{\mathbf{p}}_e$  that span the orthogonal complement of the normal direction required in (13) but gives rise to tangential forces owing to dynamical friction that can be added to the normal interaction forces,

see (5), in order to generate an appropriate signal  ${}^e\mathbf{f}$  to excite (9) by spanning a surface in the Cartesian space.

The velocity control design is based on decomposing the motion and force control directions according to hybrid force/motion control methodology by using however the estimates  $\hat{\mathbf{n}}_c$  – like the kinematic loop of [20]. The proposed kinematic controller is given by the following equation:

$$\mathbf{u} = \bar{\mathbf{P}}(\hat{\mathbf{n}}_c)\mathbf{v}_d - \hat{\mathbf{n}}_c v_f \quad (15)$$

where  $v_f$  is a PI control loop of the estimated force error  $\tilde{f}_n = \hat{f}_n - f_d$ . Note that the estimated  $\hat{f}_n$  can be calculated based on force measurements and the online estimates  $\hat{\mathbf{n}}_c$ . In particular:

$$v_f = \alpha_I \int_0^t (\hat{f}_n - f_d) d\tau + \alpha_P (\hat{f}_n - f_d), \quad \hat{f}_n = \hat{\mathbf{n}}_c^\top \mathbf{f} \quad (16)$$

with  $\alpha_I$  and  $\alpha_P$  are positive control gains.

The velocity trajectory  $\mathbf{v}_d(t)$  can be either defined a priori in a feedforward fashion or designed appropriately by using feedback of control errors such as end-effector or contact point position errors. A simple way to define  $\mathbf{v}_d(t)$  is the following:

$$\mathbf{v}_d(t) = \dot{\mathbf{p}}_d - \alpha(\mathbf{p} - \mathbf{p}_d) \quad (17)$$

where  $\alpha$  is positive control gain and  $\mathbf{p}$  can be either the end-effector position or the contact point estimate depending on the definition of the desired position  $\mathbf{p}_d$ . Note that  $\mathbf{p}_d$  can be defined as follows:

- 1) directly and a priori in the robot workspace e.g. by using vision and mapping a desired trajectory from the image space to robot space. In this case the feedback control is designed using the contact point position which is however based on estimates obtained by (9) i.e.  $\mathbf{p} := \hat{\mathbf{p}}_c$ .
- 2) locally at the surface as  $\xi_d \in \mathbb{R}^2$  and then mapped online to the robot workspace through a transformation  $(\mathbf{p}_e(0), \hat{\mathbf{R}}_c)$ . Details on the motivation behind this selection and the construction of  $\hat{\mathbf{R}}_c$  can be found in [20]. In this case the design of  $\mathbf{v}_d(t)$  is based on  $\mathbf{p}_e$  instead of  $\hat{\mathbf{p}}_c$ . This can be explained by the following observation: the objective of drawing a circle with center around the initial contact point is equivalent of drawing a virtual circle around the end-effector's initial position.

In more complicated scenarios where both sliding and rolling motion of the tool take place the estimated contact point position must be used in  $\mathbf{v}_d(t)$  even when the target is defined based on local coordinates.

Analysis of the closed loop when  $\mathbf{u}$  (15)–(17) is applied (briefly sketched in the Appendix) yields to the following theorem:

*Theorem 1:* The control law (15)–(17) applied as a translational velocity controller to a robot firmly grasping a tool which is in contact with a flat surface, together with the adaptive laws (9) and (13) used for estimating the contact parameters such as contact point position and surface orientation ensure the boundedness of the contact force and

the velocity along the unconstrained directions as well as the convergence of the force/motion errors to zero and the identification of the uncertain parameters given that  $\mathbf{u}$  and  ${}^e\mathbf{f}$  are persistently excited.

## V. EXPERIMENTS

Our adaptive control framework for tool and surface calibration was evaluated with a robot setup consisting of a 7-DOF velocity controlled manipulator controlled at 130 Hz. The manipulator also includes a wrist mounted ATI Mini45 6-axis force-torque sensor used for the force feedback control and the contact point estimator.

For more details on the experimental setup, see e.g. [23]. The end-effector velocities used for the estimation of the surface normal were calculated using joint velocities filtered through joint position measurements.



**Fig. 3 :** Experimental setup used for evaluating our adaptive control scheme for contact point and surface normal estimation.

To perform our experiments we attached a tool rigidly to the robot's gripper as shown in Fig. 3. Attaching the tool rigidly to the end-effector allowed us to have a consistent ground truth with which to compare the controller's estimation of the contact point. Furthermore, we tested the controller over a flat table which was previously calibrated to obtain the ground truth of the surface normal. A circular trajectory of 4 cm radius and 5 second period was commanded to the manipulator during the experiment.

Fig. 4 shows the normal force error  $|\tilde{f}_n| = |f_n - f_d|$  which indicates that the adaptive controller manages to regulate the normal contact force.

Fig. 5 and 6 show the estimation errors of the contact point and the surface normal respectively. The contact point converges with an error of approximately 5 mm, which, given

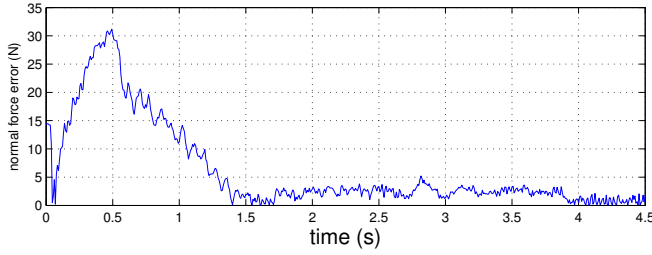


Fig. 4 : Normal force error  $|\tilde{f}_n|$ .

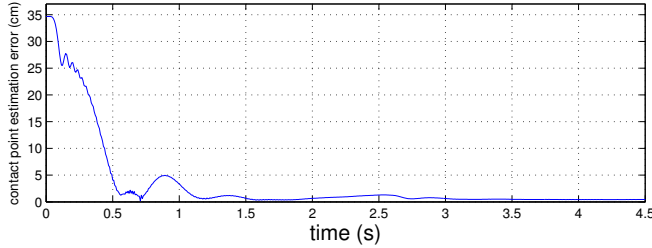


Fig. 5 : Contact point estimation error.

the 30.8 cm distance from the force-torque sensor to the tool-tip is within the error margins of the setup. Moreover, the surface normal estimate converges with an approximately 1.5 degree error with respect to the ground truth normal.

For a set of contact forces that share the same direction and only vary in magnitude, the contact point estimate  $\hat{\mathbf{p}}_c$  will converge to a point on a line passing through the actual contact point  $\mathbf{p}_c$ , parallel to the force direction. As the direction of contact forces vary,  $\hat{\mathbf{p}}_c$  will converge to the intersection point of a set of such lines, which will be  $\mathbf{p}_c$ . In the experimental convergence of  $\hat{\mathbf{p}}_c$ , as seen in Fig. 5, we see an initial convergence to a point on such a line after approx 0.1 s, and then, as the direction of the contact force starts to change as the tool-tip slides on the surface, we see convergence to the intersection point, or  $\mathbf{p}_c$ . For the setup in the experiment we see that we have good convergence for forces that spread over approximately 14 degrees with respect to the surface normal. For faster convergence with the same setup, we would need contact forces spanning that angle variation in shorter time.

## VI. CONCLUSIONS

In this paper, we have proposed a method for simultaneous online estimation of the point of contact of a tool held by a robot, and the normal of the surface it is interacting with. The method is based on adaptive estimation and a hybrid force/motion controller, and uses force and torque measurements from a wrist-mounted sensor.

The fast convergence of the contact point estimate makes it suitable for real time tracking of the endpoint of a tool that may slip and move in the robot's hand as it is being used for a task execution. The method also guarantees stable control of contact forces even before the estimates converge. For non-contact motion, the method can be used to estimate the center of mass of the end-effector and/or a grasped object.

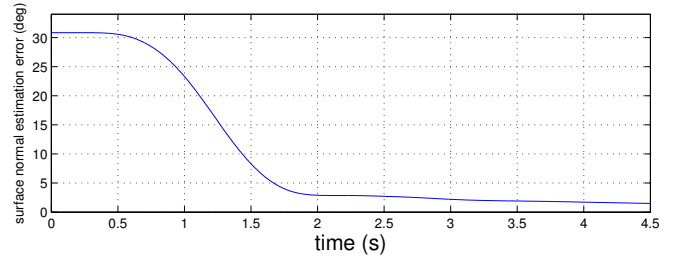


Fig. 6 : Surface normal estimation error.

This enables tool use with unmodelled and/or uncalibrated tools.

The strength of the method lies in the fact that it uses force and torque measurements and it is therefore complimentary to vision based approaches where occluded or bad lighting conditions make affect the estimation. An interesting future extension is to combine the proposed method with other methods for object tracking. One possibility is to combine it with model based vision methods or tactile sensors to use the tracked contact point to improve pose tracking of an object.

## APPENDIX

*Proof of Proposition 2:* (i) Note that  $\frac{d}{dt}(\|\hat{\mathbf{n}}_c\|^2) = -\gamma_n[\hat{\mathbf{P}}(\hat{\mathbf{n}}_c)\hat{\mathbf{n}}_c]^\top \mathbf{L}_n(t)\hat{\mathbf{n}}_c = 0$ ; thus the norm of the estimate is invariant and consequently bounded. (ii) Note that (14) and the constraint (2) implies:

$$\frac{d}{dt}(\mathbf{L}_n \mathbf{n}_c) = -\beta_n \mathbf{L}_n \mathbf{n}_c$$

which in turn, for  $\mathbf{L}_n(0) = \mathbf{O}_3$ , implies that  $\mathbf{n}_c$  belongs in the nullspace of  $\mathbf{L}_n$ . Note also that  $\mathbf{L}_n$  is positive semidefinite. Consider the following Lyapunov function:

$$V(\theta) = -\ln(\cos \vartheta), \quad V : (-\frac{\pi}{2}, \frac{\pi}{2}) \rightarrow \mathbb{R} \quad (18)$$

Its time derivative along the systems trajectories (13)-(14) is given by:

$$\dot{V}(\theta, t) = -\gamma_n \tilde{\mathbf{n}}_c^\top \mathbf{L}_n(t) \tilde{\mathbf{n}}_c \quad (19)$$

From (18) and (19) we conclude that  $V(\theta)$  is bounded which implies  $\vartheta(t) \in (-\frac{\pi}{2}, \frac{\pi}{2}), \forall t > 0$  for  $\vartheta(0) \in (-\frac{\pi}{2}, \frac{\pi}{2})$ . (iii) Clearly (i) and (ii) imply that  $\hat{\mathbf{n}}_c \in \mathcal{L}_\infty$ . Furthermore, since  $\frac{1}{1+\|\hat{\mathbf{p}}_e\|^2} \dot{\hat{\mathbf{p}}}_e \hat{\mathbf{p}}_e^\top$  is bounded by construction, (14) implies that  $\mathbf{L}_n$  and  $\dot{\mathbf{L}}_n$  are bounded too. By integrating both sides of (19) in  $t \in [0, \infty)$  and taking into account that  $V(\infty)$  is bounded from (ii), we get that  $\mathbf{L}_n^{1/2} \tilde{\mathbf{n}}_c \in \mathcal{L}_2$ . The update law (13) implies: (a)  $\|\dot{\hat{\mathbf{n}}}_c\| \leq \gamma_n \|(\mathbf{L}_n^\top)^{1/2}\| \|\mathbf{L}_n^{1/2} \tilde{\mathbf{n}}_c\|$  which implies that  $\dot{\hat{\mathbf{n}}}_c \in \mathcal{L}_\infty \cap \mathcal{L}_2$  as well as (b)  $\dot{\hat{\mathbf{n}}}_c$  given the boundedness of  $\hat{\mathbf{n}}_c$  and  $\dot{\mathbf{L}}_n$ . Clearly (a) and (b) yield  $\lim_{t \rightarrow \infty} \|\dot{\hat{\mathbf{n}}}_c\| = 0$ . (iv) The PE condition is satisfied given that there exists  $\alpha_0, T_0$  such that  $\int_t^{t+T_0} \hat{\mathbf{p}}_e \hat{\mathbf{p}}_e^\top d\tau \geq \alpha_0 \mathbf{I}_3$ . Using the integral expression of  $\mathbf{L}_n = \int_0^t \exp(-\beta_n(t-\tau)) \frac{1}{1+\|\hat{\mathbf{p}}_e\|^2} \hat{\mathbf{p}}_e \hat{\mathbf{p}}_e^\top d\tau$  implied by (14), it can be found that  $\mathbf{L}_n(t) \geq \lambda \exp(-\beta_n T_0) \mathbf{I}_3$  given that the PE condition is satisfied. Then we consider a quadratic Lyapunov function  $U = \frac{1}{2} \|\tilde{\mathbf{n}}_c\|^2$  which in our case depends

only on the estimation error angle i.e.  $U = 1 - \cos \vartheta$ . It can be easily proved that  $\dot{U} \leq -2\lambda\gamma_n \exp(-\beta_n T_0)U$

$$U(t) \leq \exp(-2\lambda\gamma_n e^{-\beta_n T_0} t) U(T_0) \quad (20)$$

Note also that  $\frac{4}{\pi^2} \|\vartheta\|^2 \leq U(\theta) \leq \frac{1}{2} \|\vartheta\|^2$  and thus (21) implies that the angle between the actual and the estimated vector  $\hat{\mathbf{n}}_c$ , converges exponential to zero as follows:

$$|\vartheta(t)| \leq \frac{\pi\sqrt{2}}{4} \exp(-\lambda\gamma_n e^{-\beta_n T_0} t) |\vartheta(T_0)| \quad (21)$$

*Proof of Theorem 1:* Let us consider the case of a bounded input  $\mathbf{v}_d$ . This assumption is valid even when  $\mathbf{v}_d$  is defined using feedback given that  $\mathbf{p}_d, \dot{\mathbf{p}}_d$  are bounded and by considering a bounded robot workspace. Saturation on the position error can be used in order to construct a bounded  $\mathbf{v}_d$ . Substituting the control law in to the system first order differential kinematics (4) we get:  $\dot{\mathbf{p}}_e = \mathbf{u}$ . By projecting the aforementioned equation along the surface normal we get:  $v_f = \frac{1}{\cos \vartheta} \mathbf{n}_c^\top \bar{\mathbf{P}}(\hat{\mathbf{n}}_c) \mathbf{v}_d$ . Since  $\hat{\mathbf{n}}_c$  is bounded and  $\vartheta \neq \pi/2$  from (i) and (ii) of Proposition 2,  $v_f$  is bounded. The boundedness of  $v_f$  implies  $\dot{\mathbf{p}}_e$  is bounded and additionally that  $\int_0^t (\hat{f}_n - f_d) d\tau, \hat{f}_n$  are bounded. Hence  ${}^e \mathbf{f}$  is bounded and thus the update law for  ${}^e \hat{\mathbf{r}}$  (9)-(11) is well-defined. The boundedness of  $\mathbf{p}_e$  can be proved by using the boundedness of the estimates  ${}^e \hat{\mathbf{r}}, \hat{\mathbf{n}}_c$  and their derivatives  ${}^e \dot{\hat{\mathbf{r}}}, \dot{\hat{\mathbf{n}}}_c$ . Ultimate bounds can also be found by exploiting  $\lim_{t \rightarrow \infty} {}^e \hat{\mathbf{r}} = \mathbf{0}_3, \lim_{t \rightarrow \infty} \dot{\hat{\mathbf{n}}}_c = \mathbf{0}_3$  (Proposition 1 and 2); analytic derivations are omitted. Given that  ${}^e \mathbf{f}, \dot{\mathbf{p}}_e$  (or  $\mathbf{v}_d$ ) satisfy the PE condition the estimation error converges to zero exponentially fast and thus  $v_f$  converges exponential fast to zero which implies  $\int_0^t (\hat{f}_n - f_d) d\tau \rightarrow 0$  and  $\hat{f}_n \rightarrow f_d$ . Furthermore,  $\dot{\mathbf{p}}_e \rightarrow \bar{\mathbf{P}}(\mathbf{n}_c) \mathbf{v}_d$  with implies that  $\bar{\mathbf{P}}(\mathbf{n}_c)(\mathbf{p} - \mathbf{p}_d) \rightarrow \mathbf{0}_3$ .

#### ACKNOWLEDGMENT

This work has been supported by the Swedish Research Council (VR), the European Union FP7 project Robo-How.Cog (FP7-ICT-288533), and the Swedish Foundation for Strategic Research. The authors gratefully acknowledge the support.

#### REFERENCES

- [1] C. Kemp, A. Edsinger, and E. Torres-Jara, "Challenges for robot manipulation in human environments [grand challenges of robotics]," *IEEE Robotics Automation Magazine*, vol. 14, no. 1, pp. 20–29, 2007.
- [2] Z. S. Roth, B. Mooring, and B. Ravani, "An overview of robot calibration," *IEEE Journal of Robotics and Automation*, vol. 3, no. 5, pp. 377–385, 1987.
- [3] C. Cheah, C. Liu, and J. Slotine, "Adaptive jacobian tracking control of robots with uncertainties in kinematic, dynamic and actuator models," *Automatic Control, IEEE Transactions on*, vol. 51, no. 6, pp. 1024–1029, 2006.
- [4] H. Bruyninckx, S. Dutre, and J. De Schutter, "Peg-on-hole: a model based solution to peg and hole alignment," in *IEEE International Conference on Robotics and Automation*, vol. 2, 1995, pp. 1919–1924.

- [5] G. Hovland and B. J. McCarragher, "Combining force and position measurements for the monitoring of robotic assembly," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, 1997, pp. 654–660.
- [6] R. Koeppel and G. Hirzinger, "Sensorimotor compliant motion from geometric perception," in *IEEE International Conference on Intelligent Robots and Systems*, vol. 2, 1999, pp. 805–811.
- [7] G. Yang, I.-M. Chen, S. H. Yeo, and W. K. Lim, "Simultaneous base and tool calibration for self-calibrated parallel robots," *Robotica*, vol. 20, pp. 367–374, 7 2002.
- [8] A. Petrovskaya, O. Khatib, S. Thrun, , and A. Y. Ng, "Touch based perception for object manipulation," in *Robotics Science and Systems Conference, Robot Manipulation Workshop*, Atlanta, GA, 2007.
- [9] C. Corcoran and R. Platt, "A measurement model for tracking hand-object state during dexterous manipulation," in *IEEE International Conference on Robotics and Automation*, 2010, pp. 4302–4308.
- [10] P. Hebert, N. Hudson, J. Ma, and J. Burdick, "Fusion of stereo vision, force-torque, and joint sensors for estimation of in-hand object location," in *IEEE International Conference on Robotics and Automation*, 2011, pp. 5935–5941.
- [11] L. Păiș, K. Umezawa, Y. Nakamura, and A. Billard, "Learning robot skills through motion segmentation and constraints extraction," *ACM/IEEE International Conference on Human-robot Interaction, Workshop on Collaborative Manipulation*, 2013.
- [12] C. Atkeson, C. An, and J. Hollerbach, "Rigid body load identification for manipulators," in *24th conf on Decision and Control*, Fort Lauderdale, FL, Dec 1985, pp. 996–1003.
- [13] D. Kubus, T. Krüger, and F. M. Wahl, "On-line rigid object recognition and pose estimation based on inertial parameters," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, 2007, pp. 1402–1408.
- [14] S. Muto and K. Shimokura, "Teaching and control of robot contour-tracking using contact point detection," in *IEEE International Conference on Robotics and Automation*, 1994, pp. 674–681.
- [15] Y. Lei and S. F. Miller, "Pose estimation and machining efficiency of an endoscopic grinding tool," *The International Journal of Advanced Manufacturing Technology*, pp. 1–11, 2013.
- [16] C. C. Kemp and A. Edsinger, "Robot manipulation of human tools: Autonomous detection and control of task relevant features," in *5th IEEE International Conference on Development and Learning (ICDL-06*, 2006.
- [17] M. Krainin, P. Henry, X. Ren, and D. Fox, "Manipulator and object tracking for in hand model acquisition," in *Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation at the Int. Conf. on Robotics & Automation (ICRA)*, Anchorage, Alaska, 2010.
- [18] M. Beetz, U. Klank, I. Kresse, A. Maldonado, L. Mosenlechner, D. Pangercic, T. Ruhr, and M. Tenorth, "Robotic roommates making pancakes," in *IEEE-RAS International Conference on Humanoid Robots*, 2011, pp. 529–536.
- [19] Y. Karayiannidis, C. Smith, F. Viña, P. Ögren, and D. Kragic, "Model-free robot manipulation of doors and drawers by means of fixed-grasps," in *IEEE International Conference on Robotics and Automation*, 2013, pp. 4470–4477.
- [20] Y. Karayiannidis and Z. Doulgeri, "Adaptive control of robot contact tasks with on-line learning of planar surfaces," *Automatica*, vol. 45, no. 10, pp. 2374–2382, 2009.
- [21] F. Viña, Y. Bekiroglu, C. Smith, Y. Karayiannidis, and D. Kragic, "Predicting slippage and learning manipulation affordances through gaussian process regression," in *IEEE-RAS International Conference on Humanoid Robots*, 2013.
- [22] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Upper Saddle River, NJ:Prentice Hall, 1996.
- [23] C. Smith and Y. Karayiannidis, "Optimal command ordering for serial link manipulators," in *IEEE-RAS International Conference on*