

# Robot force control without dynamic model: theory and experiments

Juan C. Rivera-Dueñas and Marco A. Arteaga-Pérez\*†

Universidad Nacional Autónoma de México, México, D. F., 04510, México

(Accepted March 16, 2012. First published online: April 20, 2012)

## SUMMARY

Among the many challenges to deal with, when a robot is interacting with its environment, friction at the contact surface and/or at the joints is one of the most important to be considered. In this paper we propose a control algorithm for the tracking of position and force (unconstrained orientation case only) of a manipulator end-effector that does not require the robot model for implementation. This characteristic has the advantage of making it capable to compensate friction effects without any previous estimation. Furthermore, no velocity measurements are needed, and the unit quaternion is employed for orientation control. Experimental and simulation results are provided.

KEYWORDS: Force control; Friction; Observer design.

## 1. Introduction

When a robot manipulator is in contact with the environment, it is necessary to control not only position but also the force exerted at the contact. For this aim, several approaches have been proposed in the literature (see refs. [1–3] for an overview). In general, there are two categories for interaction control strategies: direct and indirect. While in the first case the contact force can be set to a desired value by using an explicit force feedback loop, in the second one the force control is achieved exclusively via motion control. To this last category belong compliance and impedance control,<sup>4</sup> while to the first one belong some techniques like hybrid and parallel control.<sup>2</sup> For instance, in ref. [5] hybrid schemes are developed by using the orthogonal property of the velocity vector. This way, force and position terms are well defined and taken into account separately. On the other hand, note that most control algorithms make use of velocity and force measurements, which may not be available. Solutions to this problem are given in refs. [6, 7].

Another important issue in force control design and implementation is the possible lack of an accurate system model. Some solutions to cope with this problem are given in refs. [10, 15–16]. This is specially true when dealing with friction effects. In ref. [8] a model is proposed that considers many of the most important characteristics of friction, like arbitrary steady-state, hysteretic behavior because of

frictional lag, and spring-like behavior in stiction. It also gives varying break-away force depending on the rate of change of the applied force. Another phenomenon in robotic systems is meshing friction. This is caused by the inclusion of high-reduction gears in joint transmissions, and it is specially important at low velocities. In ref. [9], a new model is introduced to describe this kind of friction, and some experimental results are presented to validate it successfully.

In this paper we introduce some essential modifications to the force control algorithm given in ref. [10], yielding the following remarkable differences with the original work:

- The velocity observer has been modified to deal with curved surfaces. The algorithm in ref. [10] can *strictly* work with planes.
- The new algorithm allows not only the use of the analytical Jacobian but also of the geometrical Jacobian. This represents an essential advantage, since the later is easier to compute than the former.
- By taking advantage of the geometrical Jacobian, the orientation problem is further analyzed to employ the unit quaternion. It is important to point out that only the unconstrained orientation case is considered, but keeping an appropriate orientation of the end-effector improves force tracking.
- As explained in Remark 3.2, the usual closed loop orientation error dynamics for the unit quaternion given by<sup>2</sup>

$$\Delta \dot{\omega}_n + K_{D_0} \Delta \omega_n + K_{P_0} {}^0 R_d^d \epsilon_n = \mathbf{0}$$

is simplified to

$$\Delta \omega_n + k_\epsilon {}^0 R_d^d \epsilon_n = \mathbf{0}.$$

This represents an advantage, since the implementation is simpler.

- External perturbation terms are considered for analysis. Complex friction effects can be included here.
- A more complete set of experimental results is presented.
- Simulation results are presented for curved surfaces, and the complete model of a 6-degree of freedom (DOF) robot manipulator is written down.
- An alternative proof for the boundedness of the Lagrange multiplier  $\lambda$  is provided.

It is important to stress that despite the very similar structure of the control–observer scheme presented in ref. [10], the

\* Corresponding author. E-mail: arteaga@verona.fi-p.unam.mx

† Juan C. Rivera-Dueñas and M. A. Arteaga-Pérez are with the Departamento de Control y Robótica, DIE-FI.

complete set of experiments carried out in this work for planes are more difficult to implement than with the original algorithm, and this would simply not work for the experiment and the simulation presented for the robot in contact with a sphere.

The paper is organized as follows. The robot model for constrained movement and some properties are given in Section 2. The force/position controller with observer scheme is proposed in Section 3. Section 4 presents experimental results, while Section 5 shows a simulation for a very fast sampling time. The paper concludes in Section 6.

## 2. System Model and Properties

Consider an  $n$ -DOF rigid robot in contact with its environment, represented by an  $m$ -dimensional constraint. The dynamics of the system is then given by<sup>11</sup>

$$H(q)\ddot{q} + C(q, \dot{q})\dot{q} + D\dot{q} + g(q) = \tau - \tau_p - J_\varphi^T(q)\lambda, \tag{1}$$

where  $q \in \mathbb{R}^n$  is the vector of generalized joint coordinates,  $H(q) \in \mathbb{R}^{n \times n}$  is the symmetric positive definite inertia matrix,  $C(q, \dot{q}) \in \mathbb{R}^n$  is the vector of Coriolis and centrifugal torques,  $D \in \mathbb{R}^{n \times n}$  is a diagonal positive semi-definite matrix accounting for viscous friction,  $g(q) \in \mathbb{R}^n$  is the vector of gravitational torques,  $\tau \in \mathbb{R}^n$  is the vector of torques acting at the joints,  $\tau_p \in \mathbb{R}^n$  represents any bounded external perturbation whose first derivative is also assumed to be bounded, and  $\lambda \in \mathbb{R}^m$  is the vector of Lagrange multipliers (physically represents the force applied at the contact point).  $J_\varphi(q) = \nabla\varphi(q) \in \mathbb{R}^{m \times n}$  is assumed to be of full rank in this paper.<sup>1</sup>  $\nabla\varphi(q)$  denotes the gradient of the surface vector  $\varphi \in \mathbb{R}^m$ , which maps the vector onto the normal plane at the tangent plane that arises at the contact point described by  $\varphi(q) = \mathbf{0}$ .

**Remark 2.1.**  $\tau_p$  can include complex friction terms (modeled with smooth functions) like those described in refs. [8, 9]. △

Our goal is to design a force control law where the desired task is directly given in work-space coordinates. To do this, first of all consider the following well-known relationship:

$$\dot{x} = \begin{bmatrix} \dot{p}_n \\ \dot{\omega}_n \end{bmatrix} = J(q)\dot{q}, \tag{2}$$

where  $J(q) \in \mathbb{R}^{6 \times n}$  is the geometrical Jacobian of the manipulator,  $\dot{\omega}_n \in \mathbb{R}^v$  is the angular velocity of the end-effector, while  $\dot{p}_n \in \mathbb{R}^{(n-v)}$  is the end-effector position. We assume  $n = 6$  and  $v = 3$ . Whenever the robot is not in singularity, one also has the following relationship:

$$\dot{q} = J^{-1}(q)\dot{x}. \tag{3}$$

<sup>1</sup> Note that since  $J_\varphi(q)$  provides directions where it is possible to apply forces,  $\text{Rank}(J_\varphi(q)) < m$  means that the constraints are redundant because in that case at least two rows of  $J_\varphi(q)$  are dependent linearly.

**Assumption 2.1.** The robot doesn't reach any singularity. △

**Remark 2.2.** Notice that the robot will not reach any singularity as long as the desired path and a neighborhood around it do not pass through a singular configuration. The size of the neighborhood is given by the maximal possible error between actual and desired trajectories and can conveniently be expressed by the constant  $y_{\max}$  given in Appendix A. See also Remark 1 in ref. [12] △

Then  $x$  is given in this case by

$$x = \begin{bmatrix} p_n \\ \phi_n \end{bmatrix} = \begin{bmatrix} p_n \\ \int_0^t \omega_n d\vartheta \end{bmatrix}. \tag{4}$$

**Remark 2.3.** As explained in full detail in ref. [4],  $\phi_n$  in Eq. (4) does not have any physical meaning. Still there is no analytical reason at all not to employ it as long as only force and not torque control of the end-effector is pursued (*i.e.* only unconstrained orientation is considered). Note that a drawback of using a nonphysical variable is that it cannot be measured and it has to be computed as said in Section 4. △

We can further set<sup>2</sup>  $x = [x \ y \ z \ \phi_x \ \phi_y \ \phi_z]^T$ . Suppose now that we rewrite constraint  $\varphi(q) = \mathbf{0}$  in Cartesian coordinates, *i.e.*

$$\varphi(p_n) = \varphi(x) = \mathbf{0}, \tag{5}$$

where we make explicit that  $\varphi(x) = \mathbf{0}$  indeed depends only on the upper part of  $x$  so that the lack of physical meaning of the lower part of this vector is not relevant. Then we have

$$J_{\varphi x} = \frac{\partial \varphi(x)}{\partial x} \quad \text{and} \quad J_\varphi(q) = J_{\varphi x} J(q). \tag{6}$$

One can rewrite Eq. (1) as

$$H(q)\ddot{q} + C(q, \dot{q})\dot{q} + D\dot{q} + g(q) = \tau - \tau_p - J^T(q)J_{\varphi x}^T \lambda. \tag{7}$$

**Property 2.1.** The vector  $\dot{x}$  can be written as

$$\dot{x} = Q_x(x)\dot{x} + P_x(x)\dot{x} = Q_x(x)\dot{x}, \tag{8}$$

where  $Q_x(x) \triangleq (I_{n \times n} - P_x(x))$ ,  $P_x(x) \triangleq J_{\varphi x}^+ J_{\varphi x}$ , and  $J_{\varphi x}^+ \triangleq J_{\varphi x}^T (J_{\varphi x} J_{\varphi x}^T)^{-1} \in \mathbb{R}^{n \times m}$  stand for the Penrose's pseudoinverse and  $Q_x \in \mathbb{R}^{n \times n}$  satisfies  $\text{rank}(Q_x) = n - m$ . These two matrices are orthogonal, *i.e.*  $Q_x P_x = \mathbf{0}$  (and in fact  $Q_x J_{\varphi x}^T = \mathbf{0}$  and  $J_{\varphi x} Q_x = \mathbf{0}$ ). Note that the last equality in Eq. (8) is due to the fact that  $\dot{\varphi}(x) = J_{\varphi x} \dot{x} = \mathbf{0}$  in view of constraint (5). △

<sup>2</sup> Using  $x$  as vector and scalar (to designate the  $x$ -coordinate) should not cause any confusion.

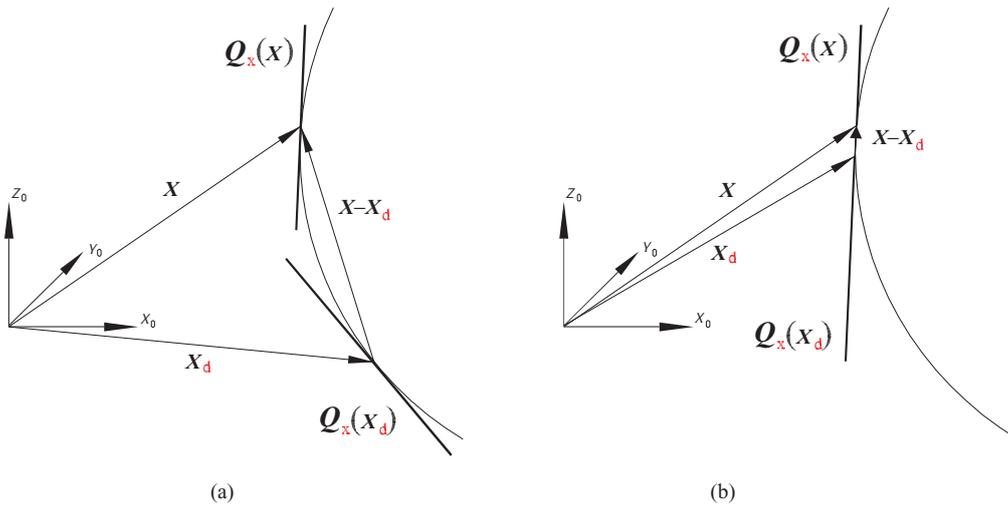


Fig. 1. (Colour online) Planes tangent to the surface at actual and desired positions. (a) Large tracking error; (b) small tracking error.

To further simplify the stability analysis, we make the following.

**Assumption 2.2.** The matrix  $J_{\varphi_x}$  is constant.  $\Delta$

**Remark 2.4.** A direct consequence of Assumption 2.2 is that both  $Q_x$  and  $P_x$  are constants, meaning that the surface is a plane. We do this assumption *only* for the sake of simplicity in designing and proving stability for our control–observer approach. In fact, the whole analysis of the next section is based on showing that the position and velocity errors, both for tracking and observation, lie exclusively in the space spanned by  $Q_x$  (see Eqs. (26)–(28)). When this matrix is not constant, the tracking error can be written as

$$\Delta x = x - x_d = Q_x(x)x + P_x(x)x - Q_x(x_d)x_d - P_x(x_d)x_d. \tag{9}$$

In Fig. 1 two cases are shown, namely when the tracking error is large, and when it is small. For the second case it can be appreciated that

$$\Delta x \approx Q_x(x)(x - x_d) = Q_x(x)\Delta x, \tag{10}$$

because it is tangent to the surface. This fact allows to carry out a *local* stability analysis for a small enough region around  $\Delta x = \mathbf{0}$ . Note, as shown in the figure, that how small the error must be, for relationship (10) to hold depends on how smooth is the surface. Also, after Property 2.1, it holds

$$\Delta \dot{x} \approx Q_x(x)(\dot{x} - \dot{x}_d) = Q_x(x)\Delta \dot{x}. \tag{11}$$

$\Delta$

### 3. Force Control–Observer Design

To design the control law, consider an auxiliary variable  $x_a$  and define the tracking error as

$$\Delta x \triangleq x - x_a. \tag{12}$$

It is assumed that  $x_a$  and its first two derivatives are bounded. One may think of  $x_a$  as the desired value for  $x$ . A possible definition is given in Eq. (44) to use the unit quaternion. On the other hand, suppose that velocity measurements are not available, then an estimate of  $x$  is given by  $\hat{x}$ , and the observation error is

$$z \triangleq x - \hat{x}. \tag{13}$$

Finally, as given in ref. [12], we define the auxiliary error variable as

$$\bar{x} \triangleq \hat{x} - x_a. \tag{14}$$

The force error is given by

$$\Delta \lambda \triangleq \lambda - \lambda_d, \tag{15}$$

where  $\lambda_d \in \mathbb{R}^m$  is the desired bounded force, with at least its first derivative bounded. Now, we define the sliding variable,

$$s \triangleq Q_x(\dot{\bar{x}} + \Lambda_x \bar{x}) + J_{\varphi_x}^+ \xi_2 \Delta F \triangleq s_p + s_f, \tag{16}$$

where  $\Lambda_x \in \mathbb{R}^{n \times n}$  and  $\xi_2 \in \mathbb{R}^{m \times m}$  are diagonal positive definite matrices and

$$\Delta F = \int_0^t \Delta \lambda d\vartheta. \tag{17}$$

Our design approach is based on making  $s$  tend to zero. First of all note that  $s_p$  and  $s_f$  are orthogonal vectors, and  $P_x J_{\varphi_x}^+ = J_{\varphi_x}^+$ . Thus, they both must become zero if  $s$  becomes zero. In this case, since  $J_{\varphi_x}^T$  is assumed to be of full rank, so is  $J_{\varphi_x}^+$ , and one has  $\Delta F = \mathbf{0}$  because  $\xi_2$  is a positive definite matrix. However, the fact that  $s_p$  becomes zero does not necessarily implies that  $\dot{\bar{x}}$  and  $\Lambda_x \bar{x}$  will be zero as well. To achieve this, the observer must be designed properly. First of all, recall that from Eq. (8) one has  $\dot{x} = Q_x \dot{x}$ . On the other

hand, Assumption 2.2 guarantees that

$$\mathbf{x} = \mathbf{Q}_x \mathbf{x}_a + \mathbf{P}_x \mathbf{x}, \tag{18}$$

with  $\mathbf{P}_x \mathbf{x} \neq \mathbf{0}$ . This means that the part of the vector  $\mathbf{x}$  lying in the space spanned by  $\mathbf{P}_x$  must be constant. Then  $\mathbf{x}_a$  must be chosen to satisfy

$$\mathbf{x}_a = \mathbf{Q}_x \mathbf{x}_a + \mathbf{P}_x \mathbf{x}_a \equiv \mathbf{Q}_x \mathbf{x}_a + \mathbf{P}_x \mathbf{x}, \tag{19}$$

$$\dot{\mathbf{x}}_a = \mathbf{Q}_x \dot{\mathbf{x}}_a. \tag{20}$$

For the estimated variable  $\hat{\mathbf{x}}$ , one too has

$$\hat{\mathbf{x}} = \mathbf{Q}_x \hat{\mathbf{x}} + \mathbf{P}_x \hat{\mathbf{x}}, \tag{21}$$

$$\dot{\hat{\mathbf{x}}} = \mathbf{Q}_x \dot{\hat{\mathbf{x}}} + \mathbf{P}_x \dot{\hat{\mathbf{x}}}. \tag{22}$$

The observer has to be designed to comply with  $\mathbf{P}_x \hat{\mathbf{x}} = \mathbf{P}_x \mathbf{x}$  and thus  $\mathbf{P}_x \dot{\hat{\mathbf{x}}} = \mathbf{0}$ . To achieve this goal, we propose

$$\hat{\mathbf{x}} = \mathbf{Q}_x \int_0^t \dot{\hat{\mathbf{x}}} d\vartheta + \mathbf{P}_x \mathbf{x}, \tag{23}$$

$$\dot{\hat{\mathbf{x}}} = \mathbf{Q}_x \left( \dot{\mathbf{x}}_a - \mathbf{A}_x \bar{\mathbf{x}} + k_d \mathbf{A}_z \int_0^t \mathbf{z}(\vartheta) d\vartheta + \mathbf{A}_z \mathbf{z} + k_d \mathbf{z} \right), \tag{24}$$

where  $\mathbf{A}_z \in \mathbb{R}^{n \times n}$  is a diagonal positive definite matrix and  $k_d$  is a positive constant. Obviously, one has  $\mathbf{P}_x \dot{\hat{\mathbf{x}}} = \mathbf{0}$  in view of Eq. (24). But, since  $\mathbf{Q}_x$  and  $\mathbf{P}_x$  are constants according to Assumption 2.2, we also have

$$\int_0^t \dot{\hat{\mathbf{x}}} d\vartheta = \int_0^t \mathbf{Q}_x \dot{\hat{\mathbf{x}}} d\vartheta = \mathbf{Q}_x \int_0^t \dot{\hat{\mathbf{x}}} d\vartheta. \tag{25}$$

To compute Eq. (25), we took advantage of the fact that  $\mathbf{Q}_x \mathbf{Q}_x = \mathbf{Q}_x$ . This means that the integral on the right-hand side of Eq. (23) lies only in the space spanned by  $\mathbf{Q}_x$ , implying that the part of the estimated vector  $\hat{\mathbf{x}}$  lying in the space spanned by  $\mathbf{P}_x$  is  $\mathbf{P}_x \mathbf{x}$ , which is what we were looking for. Note, however, that it has been multiplied by  $\mathbf{Q}_x$  to deal with curved surfaces as well. On the other hand, since we have  $\mathbf{P}_x \mathbf{x} = \mathbf{P}_x \hat{\mathbf{x}} = \mathbf{P}_x \mathbf{x}_a$ , it holds

$$\mathbf{z} = \mathbf{Q}_x \mathbf{z} \quad \bar{\mathbf{x}} = \mathbf{Q}_x \bar{\mathbf{x}}. \tag{26}$$

In other words,  $\mathbf{P}_x \mathbf{z} = \mathbf{P}_x \bar{\mathbf{x}} = \mathbf{0}$ . Since  $\mathbf{Q}_x \dot{\mathbf{x}}_a = \dot{\mathbf{x}}_a$ , then

$$\dot{\mathbf{z}} = \mathbf{Q}_x \dot{\mathbf{z}} \quad \dot{\bar{\mathbf{x}}} = \mathbf{Q}_x \dot{\bar{\mathbf{x}}}. \tag{27}$$

While Assumption 2.2 implies that only planes can be considered, we will show by means of experiments that the control-observer approach also works well with smooth curved surfaces (as can be expected after Remark 2.4). This is due to the inclusion of  $\mathbf{Q}_x$  in Eq. (24), contrary to the observer in ref. [10]. Now, suppose that we have  $\mathbf{Q}_x \mathbf{A}_x = \mathbf{A}_x \mathbf{Q}_x$  (for example by setting  $\mathbf{A}_x = k_x \mathbf{I}$ , with  $k_x > 0$ ). Then  $s_p$  in Eq. (16) can be rewritten as

$$s_p = \mathbf{Q}_x (\dot{\hat{\mathbf{x}}} + \mathbf{A}_x \bar{\mathbf{x}}) = \dot{\hat{\mathbf{x}}} + \mathbf{A}_x \bar{\mathbf{x}}. \tag{28}$$

From Eq. (28) one can conclude that both  $\bar{\mathbf{x}}$  and  $\dot{\hat{\mathbf{x}}}$  tend to zero if  $s_p$  tends to zero, i.e. if  $s$  does. But showing that  $\bar{\mathbf{x}}$  and  $\dot{\hat{\mathbf{x}}}$  tend to zero is not our main goal. Rather, this fact is used to prove that  $\Delta \mathbf{x}$ ,  $\Delta \dot{\mathbf{x}}$ ,  $\mathbf{z}$ , and  $\dot{\mathbf{z}}$  do tend to zero as well, with a proper design of the controller. Consider

$$\dot{\mathbf{x}}_r \triangleq \mathbf{Q}_x (\dot{\mathbf{x}}_a - \mathbf{A}_x \bar{\mathbf{x}}) - \mathbf{J}_{\varphi x}^+ \xi_2 \Delta \mathbf{F} + s_d - \mathbf{K}_\gamma \sigma, \tag{29}$$

where  $\mathbf{K}_\gamma \in \mathbb{R}^{n \times n}$  is a diagonal positive definite matrix and  $\sigma \in \mathbb{R}^n$ , with

$$s_d = s(0) e^{-k_1 t}, \tag{30}$$

$$\sigma = \int_0^t \{ \mathbf{K}_\beta s_1(\vartheta) + \text{sign}(s_1(\vartheta)) \} d\vartheta, \tag{31}$$

$$s_1 = s - s_d, \tag{32}$$

where  $\sigma(0) = \mathbf{0}$ , and  $\mathbf{K}_\beta \in \mathbb{R}^{n \times n}$  is a diagonal positive definite matrix,  $k_1$  is a positive constant, and  $\text{sign}(s_1) \triangleq [\text{sign}(s_{11}) \dots \text{sign}(s_{1n})]^T$ , with  $s_{1i}$  element of  $s_1$ ,  $i = 1, \dots, n$ .  $s_d \in \mathbb{R}^n$  could be set to zero without affecting the stability analysis, and it is only used to get a better transient performance by getting  $s_1(0) = \mathbf{0}$  (see ref. [12] for details). Then, consider

$$s_x \triangleq \dot{\mathbf{x}} - \dot{\mathbf{x}}_r \tag{33}$$

$$= \mathbf{Q}_x (\Delta \dot{\mathbf{x}} + \mathbf{A}_x \bar{\mathbf{x}}) + \mathbf{J}_{\varphi x}^+ \xi_2 \Delta \mathbf{F} - s_d + \mathbf{K}_\gamma \sigma,$$

$$s_r \triangleq \dot{\mathbf{q}} - \dot{\mathbf{q}}_r = \mathbf{J}^{-1}(\mathbf{q}) (\dot{\mathbf{x}} - \dot{\mathbf{x}}_r) = \mathbf{J}^{-1}(\mathbf{q}) s_x. \tag{34}$$

Our next step is to rewrite the robot dynamics (7) in terms of  $s_r$  as follows

$$\mathbf{H}(\mathbf{q}) \dot{s}_r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) s_r + \mathbf{D} s_r = \boldsymbol{\tau} - \mathbf{J}^T(\mathbf{q}) \mathbf{J}_{\varphi x}^T \boldsymbol{\lambda} - \mathbf{y}_a, \tag{35}$$

where

$$\mathbf{y}_a \triangleq \mathbf{H}(\mathbf{q}) \ddot{\mathbf{q}}_r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \dot{\mathbf{q}}_r + \mathbf{D} \dot{\mathbf{q}}_r + \mathbf{g}(\mathbf{q}) + \boldsymbol{\tau}_p. \tag{36}$$

Consider now the auxiliary variables

$$\dot{\mathbf{x}}_o \triangleq \dot{\hat{\mathbf{x}}} - \mathbf{A}_z \mathbf{z}, \tag{37}$$

$$\mathbf{r} \triangleq \dot{\mathbf{x}} - \dot{\mathbf{x}}_o = \dot{\mathbf{z}} + \mathbf{A}_z \mathbf{z}, \tag{38}$$

$$s_o \triangleq \dot{\mathbf{x}}_o - \dot{\mathbf{x}}_r. \tag{39}$$

Based on all previous definitions, the proposed control law is

$$\boldsymbol{\tau} = -\mathbf{K}_p \mathbf{J}^{-1} s_o + \mathbf{J}^T \mathbf{J}_{\varphi x}^T \boldsymbol{\lambda}_d - \mathbf{J}^T \mathbf{J}_{\varphi x}^T \xi_1 \Delta \mathbf{F}, \tag{40}$$

with  $\mathbf{K}_p \in \mathbb{R}^{n \times n}$  and  $\xi_1 \in \mathbb{R}^{m \times m}$  being positive definite matrices, and the dependence on  $\mathbf{q}$  has been omitted for simplicity. By substituting Eq. (40) in Eq. (35) and by taking into account that  $s_o = s_x - \mathbf{r}$ , one gets

$$\begin{aligned} \mathbf{H}(\mathbf{q}) \dot{s}_r + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) s_r + \mathbf{K}_{DP} s_r &= -\mathbf{J}^T \mathbf{J}_{\varphi x}^T \Delta \boldsymbol{\lambda} \\ &+ \mathbf{K}_p \mathbf{J}^{-1} \mathbf{r} - \mathbf{J}^T \mathbf{J}_{\varphi x}^T \xi_1 \Delta \mathbf{F} - \mathbf{y}_a, \end{aligned} \tag{41}$$

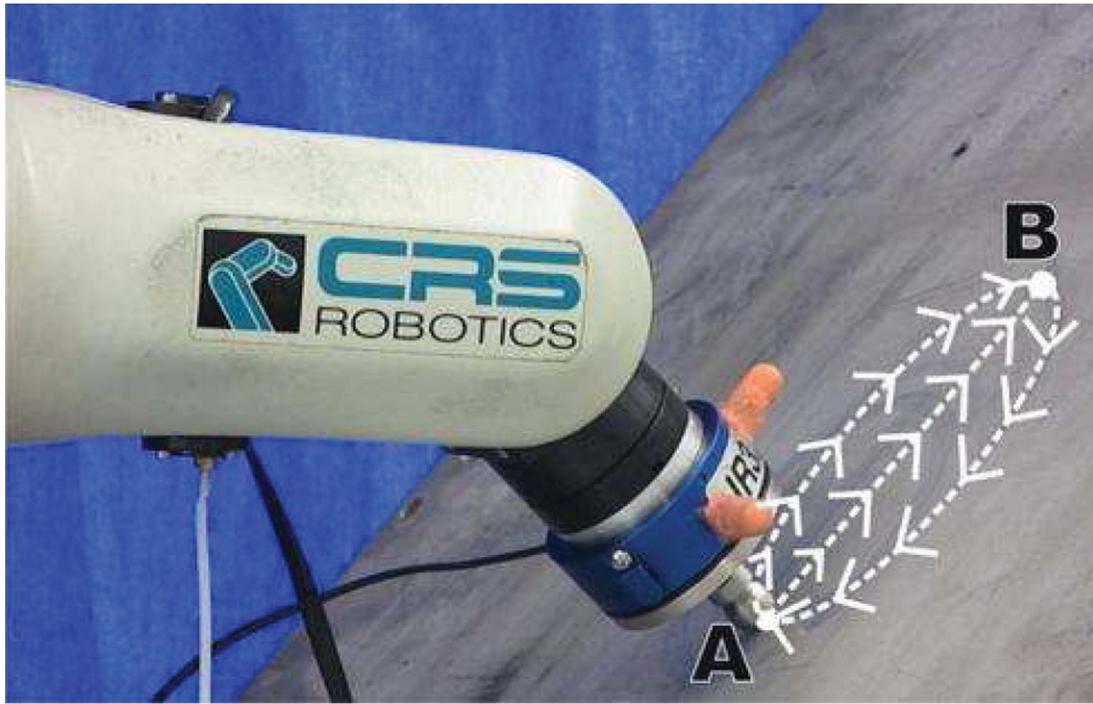


Fig. 2. (Colour online) Robot A465 of CRS Robotics in contact with a flat surface.

where  $\mathbf{K}_{DP} \triangleq \mathbf{D} + \mathbf{K}_p$ . Equation (41) is related to the boundedness of the tracking and force errors. We still need to describe the dynamics of the observation error. This can be done as explained in ref. [10] by assuming  $\Lambda_z \mathbf{Q}_x = \mathbf{Q}_x \Lambda_z$  (one can choose  $\Lambda_z = k_z \mathbf{I}$ , with  $k_z > 0$ ) to get

$$\dot{\mathbf{r}} + k_d \mathbf{r} = \mathbf{Q}_x (\Delta \ddot{\mathbf{x}} + \Lambda_x \dot{\mathbf{x}}). \quad (42)$$

Now consider the following definition for the state of the error dynamics (17), (41), and (42)

$$\mathbf{y} \triangleq [s_r^T \quad \mathbf{r}^T \quad \Delta \mathbf{F}^T]^T. \quad (43)$$

**Theorem 3.1.** Consider a bounded continuous trajectory  $\mathbf{x}_a$  (with bounded derivatives), which is chosen far away enough from any singularity, and a desired bounded force  $\lambda_d$ , with bounded first derivative. Then, for the observer (23)–(24) and the control law (40) in closed loop with system (7), a proper combination of the gains  $k_1, k_d, \Lambda_x, \Lambda_z, \mathbf{K}_\beta, \mathbf{K}_\gamma, \xi_1, \xi_2$ , and  $\mathbf{K}_p$  can always be found depending on the initial condition  $\mathbf{y}(0)$ , the desired trajectories, and the robot model parameters so that any variable in the error dynamics given by Eqs. (17), (41), and (42) is bounded and tracking and observation errors ( $\Delta \mathbf{x}, \Delta \dot{\mathbf{x}}, z, \dot{z}, \Delta \mathbf{F}, \Delta \lambda$ ) tend to zero.  $\Delta$

**Remark 3.1.** The result of Theorem 3.1 is only *local*, since gains depend on  $\mathbf{y}(0)$ . Furthermore the region of attraction cannot be made arbitrarily large in general and, actually, it should be chosen rather small to work with curved surfaces. A sketch of the proof is given in Appendix A.  $\Delta$

**Remark 3.2.** As explained before, definition (4) does not provide any physical meaning. However, for force control the unconstrained orientation should be perpendicular to

the contact surface. The best option is to employ the unit quaternion. First we define

$$\dot{\mathbf{x}}_a = \begin{bmatrix} {}^0 \dot{\mathbf{p}}_d \\ {}^0 \boldsymbol{\omega}_d - k_\epsilon {}^0 \mathbf{R}_d {}^d \boldsymbol{\epsilon}_n \end{bmatrix}, \quad (44)$$

with  $\mathbf{x}_a = \int_0^t \dot{\mathbf{x}}_a d\vartheta$ .  $k_\epsilon$  is a positive scalar gain,  ${}^0 \mathbf{p}_d$  is the desired position of the end-effector as before,  ${}^0 \boldsymbol{\omega}_d \in \mathbb{R}^3$  is the desired angular velocity, and  ${}^0 \mathbf{R}_d \in \mathbb{R}^{3 \times 3}$  is the desired rotation matrix, *i.e.* it represents the desired orientation.  ${}^d \boldsymbol{\epsilon}_n \in \mathbb{R}^3$  is the vector part of the unit quaternion associated to the rotation matrix given by  ${}^d \mathbf{R}_n = {}^0 \mathbf{R}_d^T \mathbf{R}_n$ , where  ${}^0 \mathbf{R}_n \in \mathbb{R}^{3 \times 3}$  is the rotation matrix between the end-effector frame and the base frame. Note that the desired position  ${}^0 \mathbf{p}_d$  has to be chosen to satisfy Eq. (5), and care should be taken to avoid getting a not-bounded  $\mathbf{x}_a$ . Also, the desired orientation  ${}^0 \mathbf{R}_d$  has to be chosen perpendicular to the surface at  $\boldsymbol{\varphi}({}^0 \mathbf{p}_d) = \mathbf{0}$ . Then, according to Theorem 3.1, it is guaranteed that  $\Delta \mathbf{x} \rightarrow \mathbf{0}$  and  $\Delta \dot{\mathbf{x}} \rightarrow \mathbf{0}$ . Clearly, the position error requires no extra analysis, but the orientation case does. When  $\Delta \mathbf{x} = \mathbf{0}$ , the orientation error dynamics is easily computed as

$$\Delta \boldsymbol{\omega}_n + k_\epsilon {}^0 \mathbf{R}_d {}^d \boldsymbol{\epsilon}_n = \mathbf{0}, \quad (45)$$

where  $\Delta \boldsymbol{\omega}_n = {}^0 \boldsymbol{\omega}_n - {}^0 \boldsymbol{\omega}_d$ . It is worthy to note that the closed loop dynamics presented in ref. [2] is more complex and involves more gains, since it is given by

$$\Delta \dot{\boldsymbol{\omega}}_n + \mathbf{K}_{D_0} \Delta \boldsymbol{\omega}_n + \mathbf{K}_{P_0} {}^0 \mathbf{R}_d {}^d \boldsymbol{\epsilon}_n = \mathbf{0}, \quad (46)$$

where  $\mathbf{K}_{D_0}$  and  $\mathbf{K}_{P_0}$  are positive definite matrices. However, the stability analysis can be carried out in a similar way

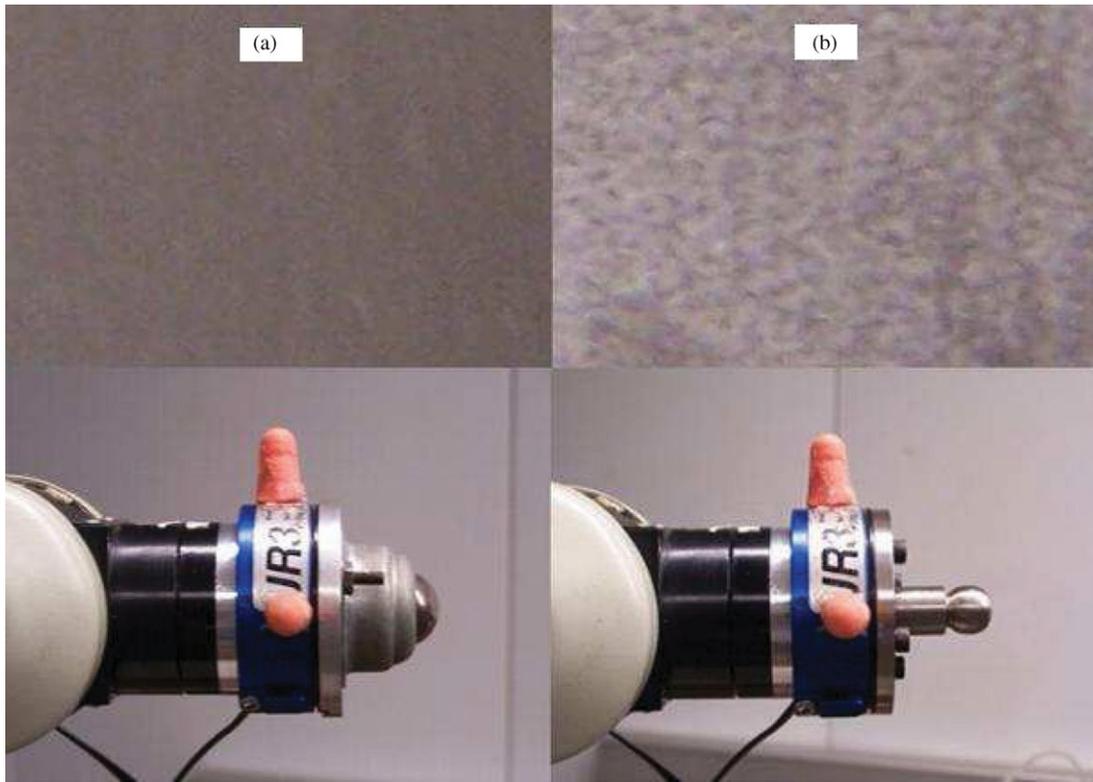


Fig. 3. (Colour online) (a) Polished surface with rolling end-effector (less friction); (b) not-polished surface with finger (more friction).

as done in the reference to conclude that  ${}^0\mathbf{R}_n \rightarrow {}^0\mathbf{R}_d$  and  ${}^0\Delta\boldsymbol{\omega} = {}^0\boldsymbol{\omega}_n - {}^0\boldsymbol{\omega}_d \rightarrow \mathbf{0}$ .  $\triangle$

## 4. Experimental Results

### 4.1. Testing different levels of friction

In this section the theory of Section 3 is tested. The test bed consists of the 6-DOF industrial rigid robot A465 of *CRS Robotics*, which has been especially adapted for implementation of research control algorithms<sup>19</sup> (see Figs. 2 and 10). We have carried out five experiments. For the first three, we used a plane as physical constraint, while in the fourth one a curve (part of a sphere) is employed. Experiment 5 assumes that the surface is flat, while it is actually slightly concave, thus introducing a mismatch in the constraint equation.

Our first two experiments consist in making a circle on a plane with a slope of  $45^\circ$ , as shown in Fig. 2. The movement begins at point A and goes to point B in 4 s. From there, a circle is made which finishes again in B. The experiment lasts 10 s. The orientation, i.e. the  $z$ -axis of the end-effector coordinate frame, has to be perpendicular to the surface (see Fig. 26). Also, we have chosen the  $x$ -axis to point upwards, parallel to the surface. To check out that the control algorithm works well under different conditions of friction, we use two surfaces and end-effectors as shown in Fig. 3.

We have implemented the following control law in terms of input voltages (see Appendix B for details),

$$\mathbf{V} = -\mathbf{K}_p \mathbf{J}^{-1} \mathbf{s}_o + \mathbf{D}_K^{-1} \mathbf{D}_n \{ \mathbf{J}^T \mathbf{J}_{\varphi_x}^T \boldsymbol{\lambda}_d - \mathbf{J}^T \mathbf{J}_{\varphi_x}^T \boldsymbol{\xi}_1 \Delta \mathbf{F} \}.$$

The different parameters of the control law are  $\mathbf{K}_p = \text{diag} \{2.625 \ 1.5 \ 2.625 \ 2.625 \ 2 \ 2.625\}$ ,  $\boldsymbol{\Lambda}_x = \text{diag} \{46.5 \ 44 \ 47 \ 46.5 \ 44 \ 46.5\}$ ,  $\boldsymbol{\Lambda}_z = \text{diag} \{45 \ 45 \ 45 \ 45 \ 45 \ 45\}$ ,  $\mathbf{K}_\gamma = \text{diag} \{0.07 \ 0.07 \ 0.07 \ 0.7 \ 0.7 \ 0.07\}$ ,  $\mathbf{K}_\beta = \text{diag} \{20 \ 19 \ 20 \ 19 \ 19 \ 19\}$ ,  $\xi_1 = 10.7$ ,  $\xi_2 = 0.00001$ ,  $k_d = 100$ ,  $k_1 = 0.001$ ,  $k_e = 0.001$ .

Figure 4 shows the tracking errors  $\Delta \mathbf{x}$ . Recall that difference in the experiments is the presence or absence of high friction on the contact surface. As can be appreciated, the results are quite similar so that we can conclude that the algorithm is working as foreseen in theory in this respect. The first three elements of  $\Delta \mathbf{x}$  correspond to position errors. As can be appreciated in both cases, they are under 2 mm for most of the time for the  $x$ - and  $z$ -axes. Regarding the  $y$ -axis, the error became suddenly larger after 4 s. The reason is that while the desired trajectories are continuous for position, we have done a discontinuity in the desired velocities at  $t = 4$  s, when the circle begins at point B, so that necessarily the error in velocity affects the position tracking. On the other hand, keeping the orientation constant makes the desired joint angles to change rather abruptly, which causes increment in the errors at  $t = 6$  s and 9 s, when the extremes of the circle are being done. Note, however, that the last three elements of  $\Delta \mathbf{x}$  show good outcomes. Recall that this part of the vector has to tend to zero in order for the orientation and angular velocity errors to do so as well. As can be seen in Fig. 4, the maximal error is about 0.1 rad, caused also by the extremes of the trajectory. It is important to point out that we got  ${}^0\boldsymbol{\phi}_n$  by employing

$${}^0\boldsymbol{\phi}_n(kT) = {}^0\boldsymbol{\phi}_n((k-1)T) + \mathbf{J}_\omega(\mathbf{q})\{\mathbf{q}(kT) - \mathbf{q}((k-1)T)\},$$

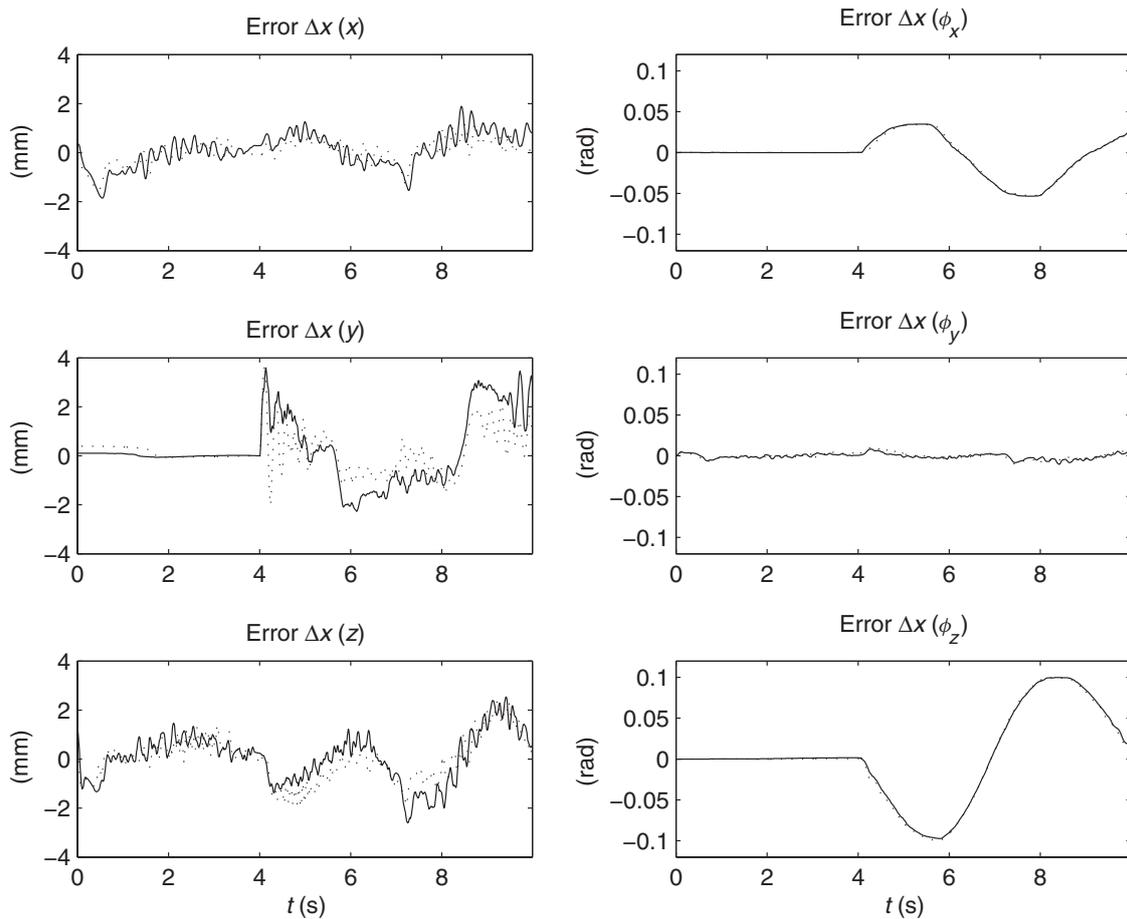


Fig. 4. Experiments 1 and 2. Testing different levels of friction on contact surfaces. Tracking error vector  $\Delta x$ . Surface with friction (—) and without friction (···).

where  $J_\omega(q) \in \mathbb{R}^{3 \times n}$  is the lower part of the geometrical Jacobian  $J(q)$  and  $T = 10$  ms is the sampling time.

In Fig. 5 the observation errors are shown. In order to have a better point of comparison, we use the same scales as in Fig. 4. It can be appreciated that the observer is working very well, and most remarkably, the outcomes are very similar for both cases, i.e. with and without friction.

Figure 6 shows real force against desired force, and the corresponding errors. The outcome values can only be considered to be acceptable, although they could improve with a smaller sampling time because it would allow to set gains larger, specially  $\xi_2$ . Note, how small this parameter had to be set. Still, the error is bounded by  $\pm 10$  N for most of the time and it is quite similar in both cases so that one can conclude that friction is not the cause of these errors. At about time  $t = 6$  s the force error becomes larger in the unpolished surface. However, we consider that the reason is not friction itself but a small hole that the irregular surface has at a particular point, which causes the robot to push stronger when falling there. We have opened a window from  $t = 5.5$  s to 6.5 s, where it can be seen that the errors remain bounded by  $\pm 10$  N for the polished plane.

#### 4.2. Testing the effect of velocity in friction

The third experiment that we have carried out is basically the very same as Experiment 1, but three times slower. Since

friction is dependent on velocity, it is aimed at analyzing whether the performance is affected, for the better or the worse, when the velocity diminishes. Note that it is natural to expect better results for slower movements. In Fig. 7 we show results for the tracking errors. It can be appreciated that the outcomes are slightly better for slower velocities. Then we can conclude that a slower movement is essentially not affecting the performance in the tracking errors.

On the other hand, by comparing Fig. 5 with Fig. 8, it can be seen that the observation errors improved. We claim that a lower velocity is not the one single cause for this, but also the fact that the implementation of many digitalized observer integrals becomes less sensitive to the sampling period.

Finally, a comparison of Fig. 6 with Fig. 9 clearly shows an improvement. In conclusion, the results are slightly better in all cases, and much better in the force control, so that in this case the friction factor might be the reason for the improvement.

#### 4.3. Testing the algorithm on a curved surface

This experiment is meant to test the proposed algorithm in a surface that is not a plane. For this goal, we use the part of a sphere as shown in Fig. 10. As pointed out in Section 3 (after Remark 2.4), the observer (23)–(24) together with the control law (40) can deal with this kind of constraints.

The desired trajectory and orientation can also be seen in Fig. 10, while the total time of the experiment is 12 s. The

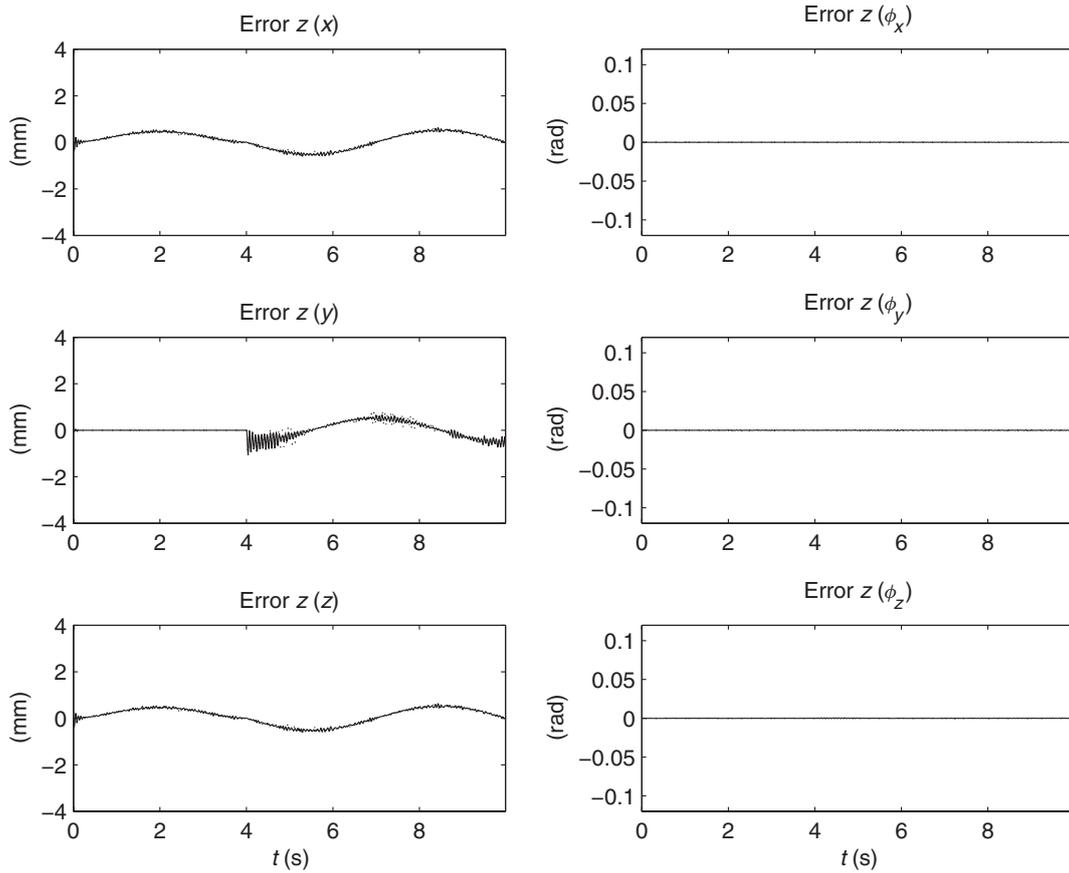


Fig. 5. Experiments 1 and 2. Testing different levels of friction on contact surfaces. Observer error vector  $z$ . Surface with friction (—) and without friction (···).

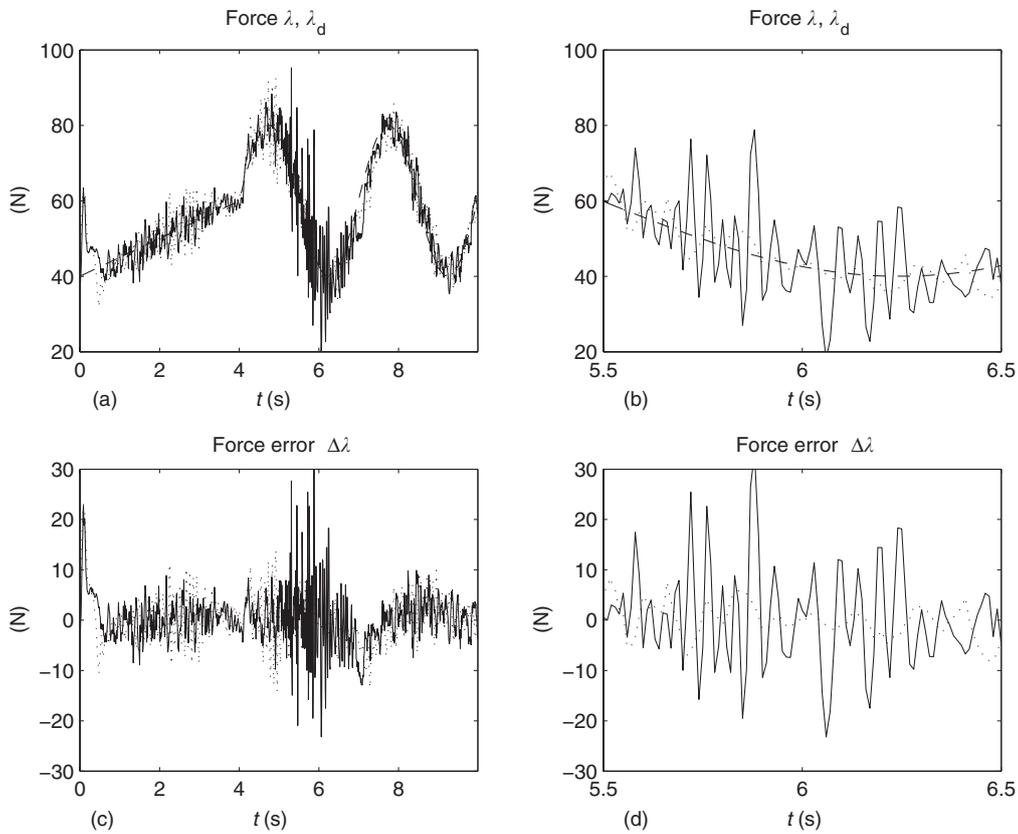


Fig. 6. Experiments 1 and 2. Testing different levels of friction on contact surfaces. Surface with friction (—) and without friction (···). (a–b) Desired force (---) versus real force; (c–d) Force error  $\Delta\lambda$ .

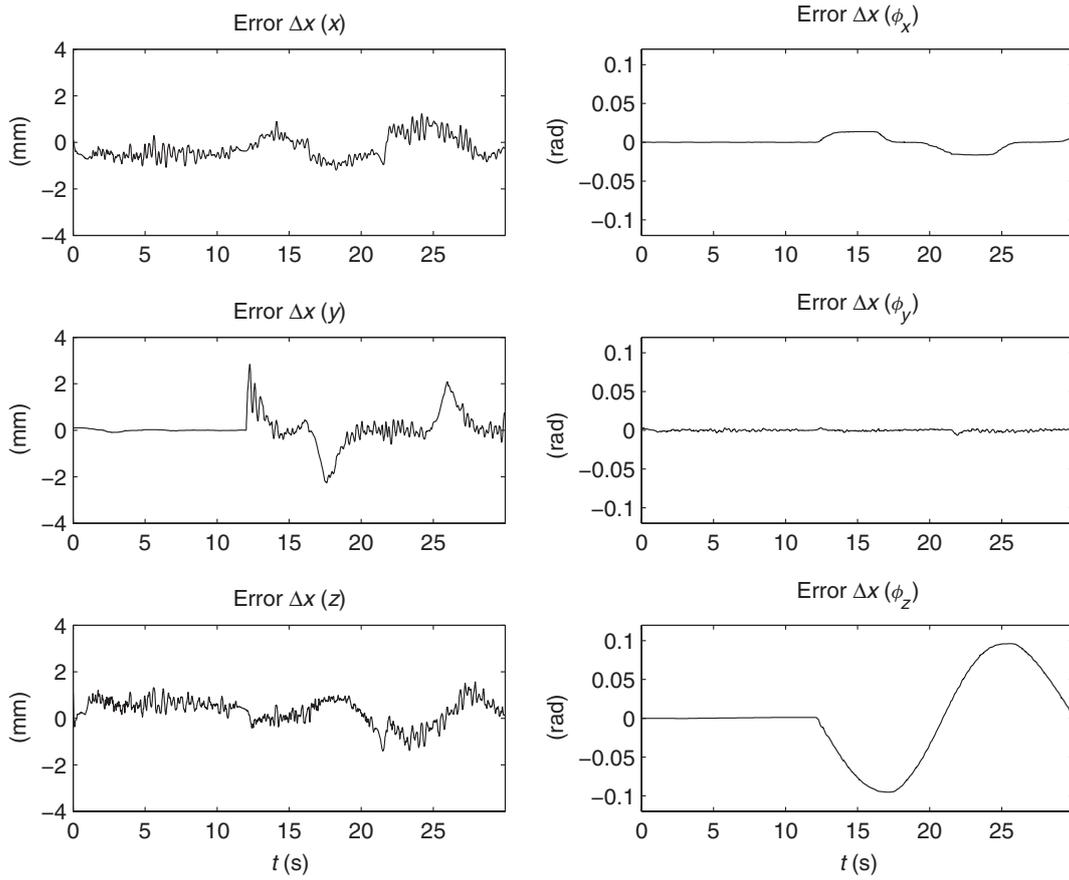


Fig. 7. Experiment 3. Slow desired trajectories. Tracking error vector  $\Delta \mathbf{x}$ .

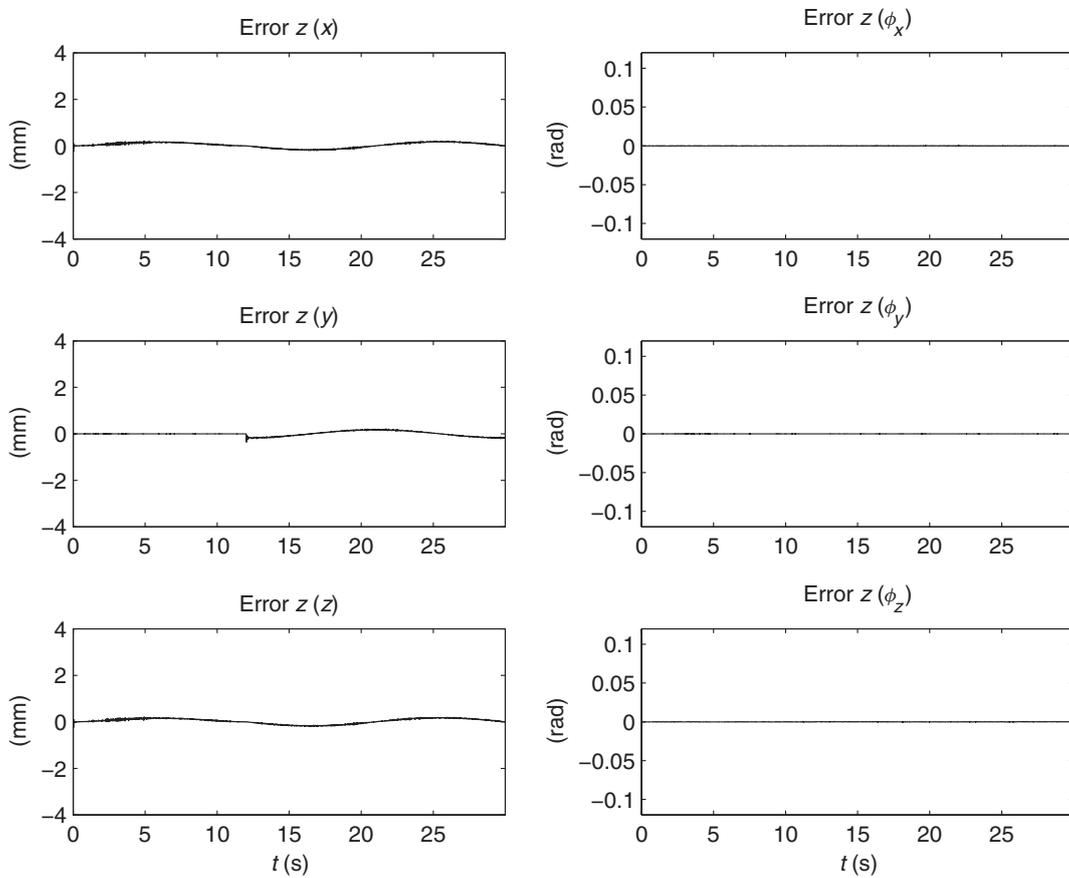


Fig. 8. Experiment 3. Slow desired trajectories. Observer error vector  $\mathbf{z}$ .

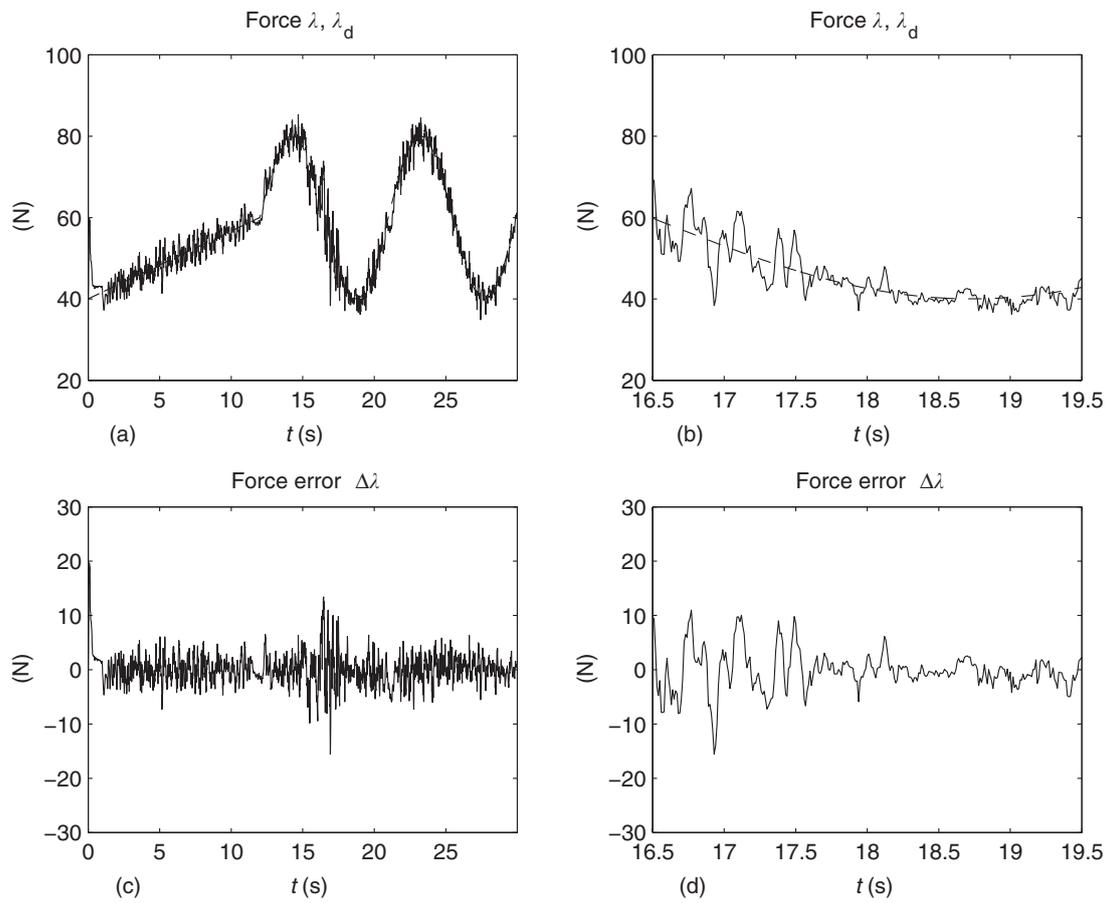


Fig. 9. Experiment 3. Slow desired trajectories. (a–b) Desired force (– –) versus real force (—); (c–d) force error  $\Delta\lambda$ .

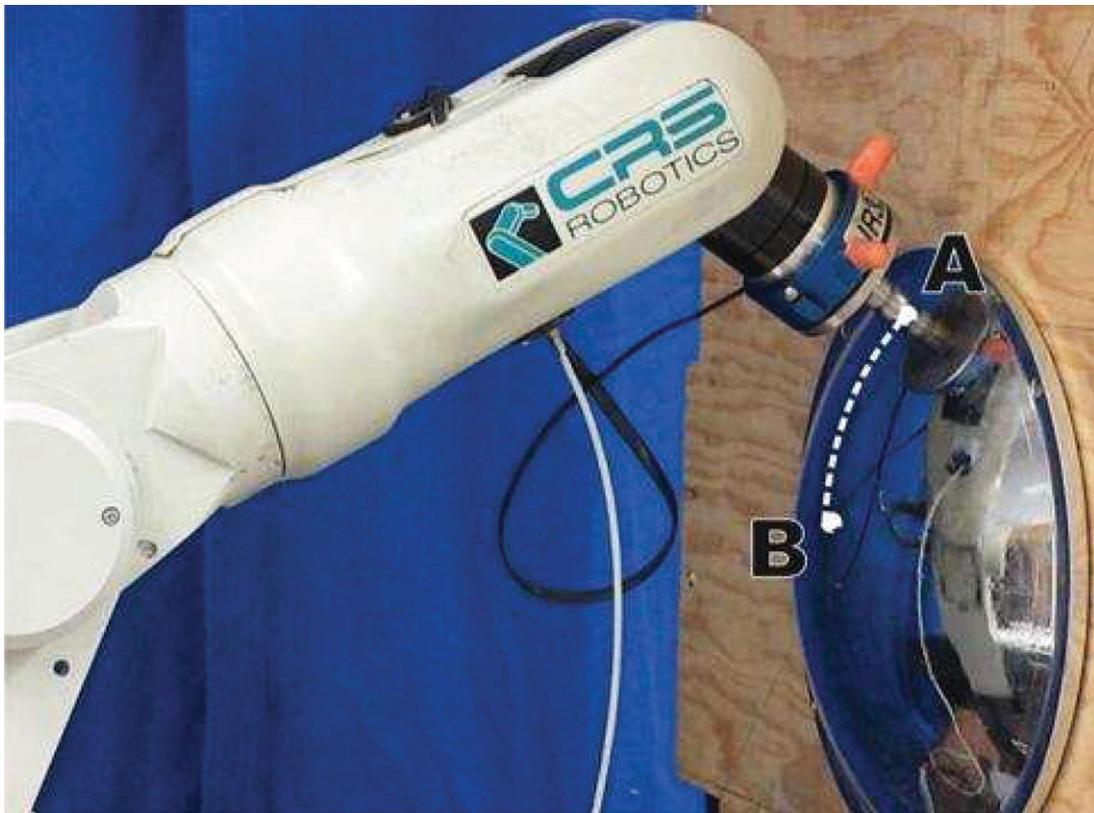


Fig. 10. (Colour online) Robot A465 of CRS Robotics in contact with a curved surface.

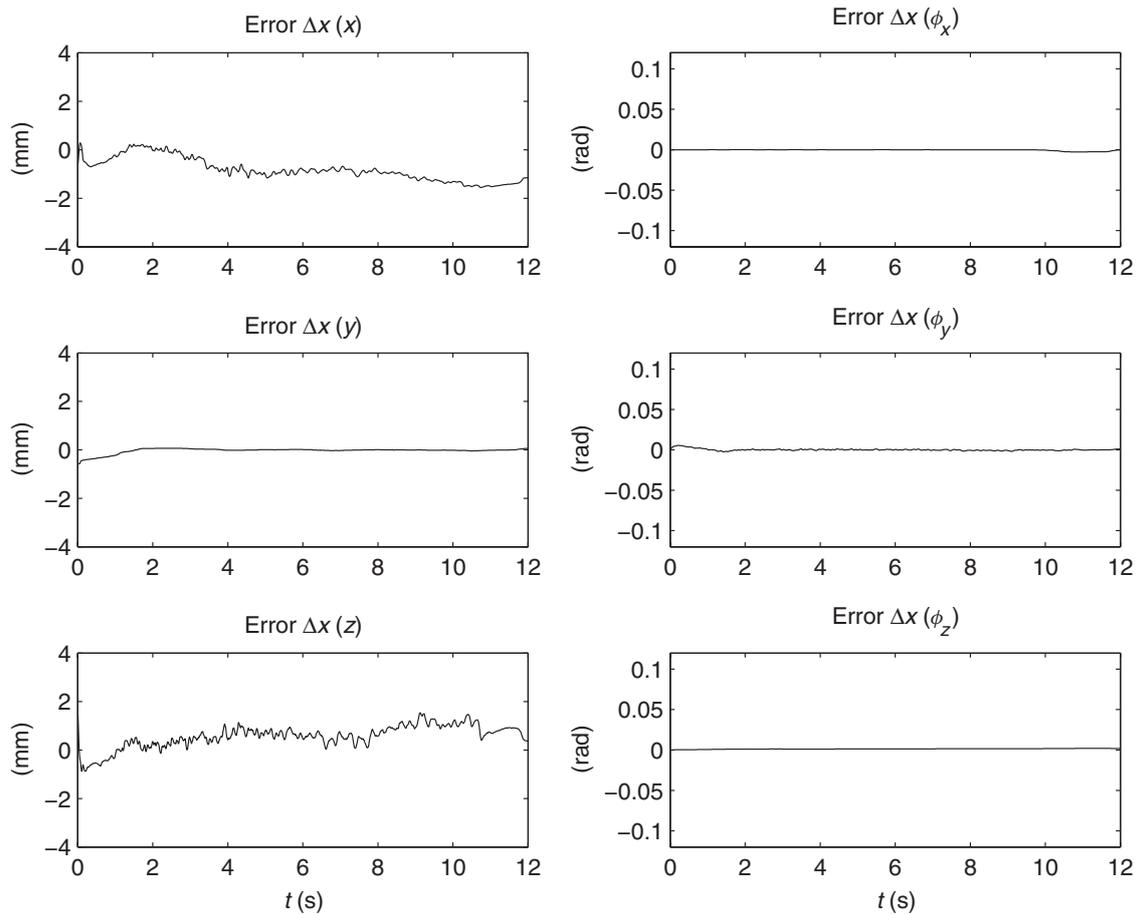


Fig. 11. Experiment 4. Movement on a curved surface. Tracking error vector  $\Delta\mathbf{x}$ .

results for the tracking error are shown in Fig. 11. It can be appreciated that the errors are in fact smaller than for the flat surface. The reason is probably that the movement is less complex from the required joint angles point of view.

Figure 12 shows the observation errors. By comparing with Figs. 5 and 8, it can be seen that these are similar to other cases. Finally, Fig. 13 shows the desired and real forces, together with the corresponding error. The results are indeed good, because for most of the time the errors are bounded by about  $\pm 5$  N.

#### 4.4. Testing robustness in the case of not well-known surfaces

To analyze the effect of the exact knowledge of constraint (5), we consider making the flat surface slightly concave. Our goal is to repeat either Experiment 1 or 2. To protect our test bed, we preferred to repeat Experiment 2. This should not invalidate the results because the outcomes of the first two experiments are similar. Besides, right now we are not testing the behavior because of friction. In Fig. 14 we show the concave surface, while it is a matter of course that it is assumed to be perfectly flat as before.

The results can be appreciated in Figs. 15 to 17. The direct comparison is with Figs. 4 to 6. In general, the results are slightly worse for position but slightly better for orientation so that altogether we consider them to be similar. As to force tracking, it gets clearly worse when the circle is being made. For the first 4s the outcomes are similar in magnitude.

Since we are carrying out the faster experiment, we could expect better results for slower movements. In any case we think that our approach shows a good degree of robustness. Experiments for a convex surface are quite similar to the case of the concave one so that we omit them here for lack of room.

### 5. Simulation Results

The experimental results of Section 4 were not as accurate as expected from the theory developed in Section 3. We claim that the reason is the digitalization process necessary for implementation. However, for hardware issues, we were not able to have a sampling time faster than 10 ms, which in practice proved to be too large. Not being able to set gains better, exact tracking was not always achieved in the experiments. To show that our claim is correct, a simulation has been carried out for the same robot A465 of *CRS Robotics*. In Appendix B the complete model of this manipulator can be found.

We have chosen to use the same sphere as that of Section 4.3. However, to clearly show the difference of the control algorithm of Section 3 and that of ref. [10], the task to be accomplished is more demanding. As illustrated in Fig. 18, the desired trajectory is larger than that depicted in Fig. 10. This makes more evident the curvature of the sphere. Furthermore, the required time to reach the final position will be smaller (the total simulation lasts for 8 s instead of 12 s

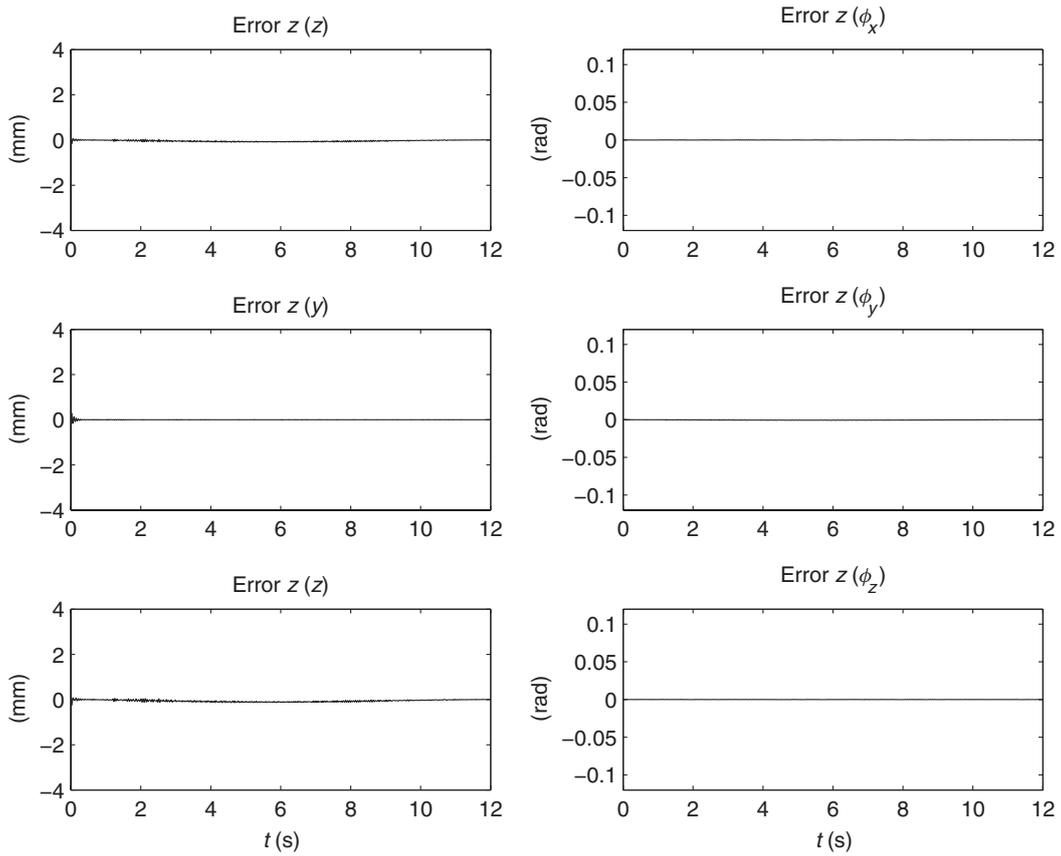


Fig. 12. Experiment 4. Movement on a curved surface. Observer error vector  $z$ .

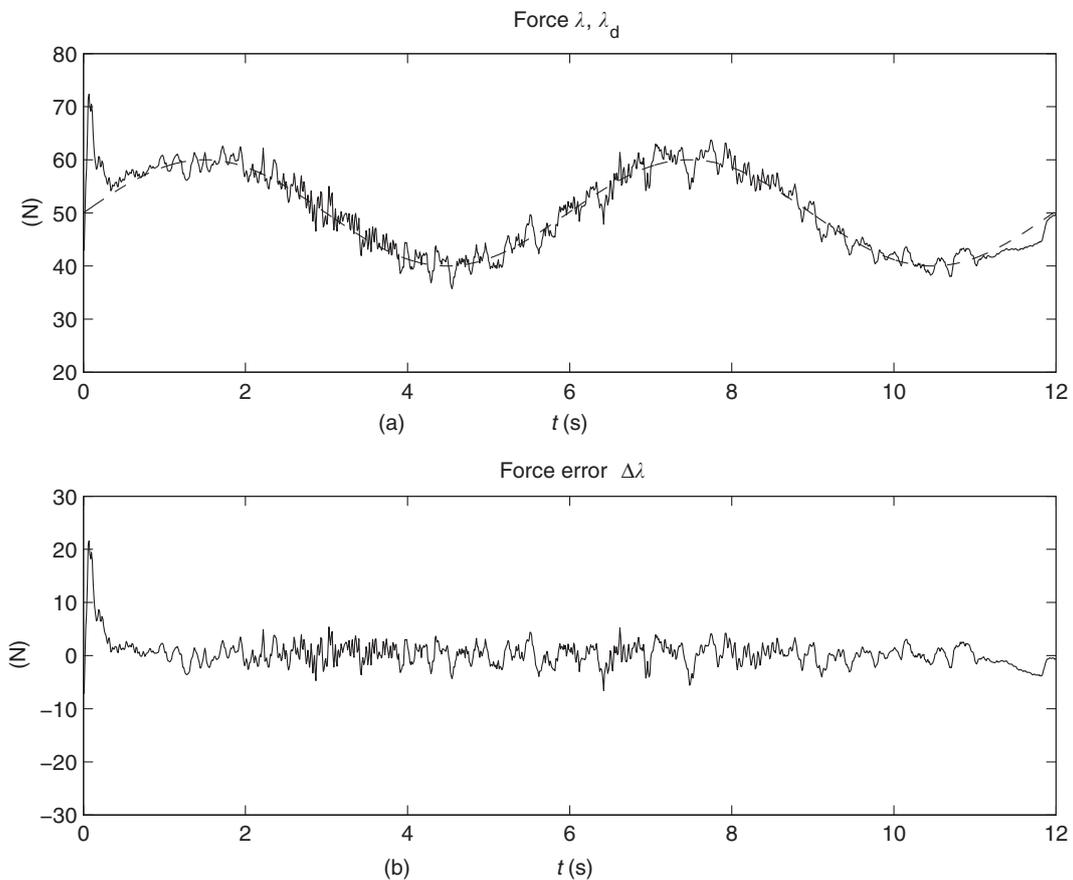


Fig. 13. Experiment 4. Movement on a curved surface. (a) Desired force ( - - ) versus real force ( — ); (b) force error  $\Delta\lambda$ .

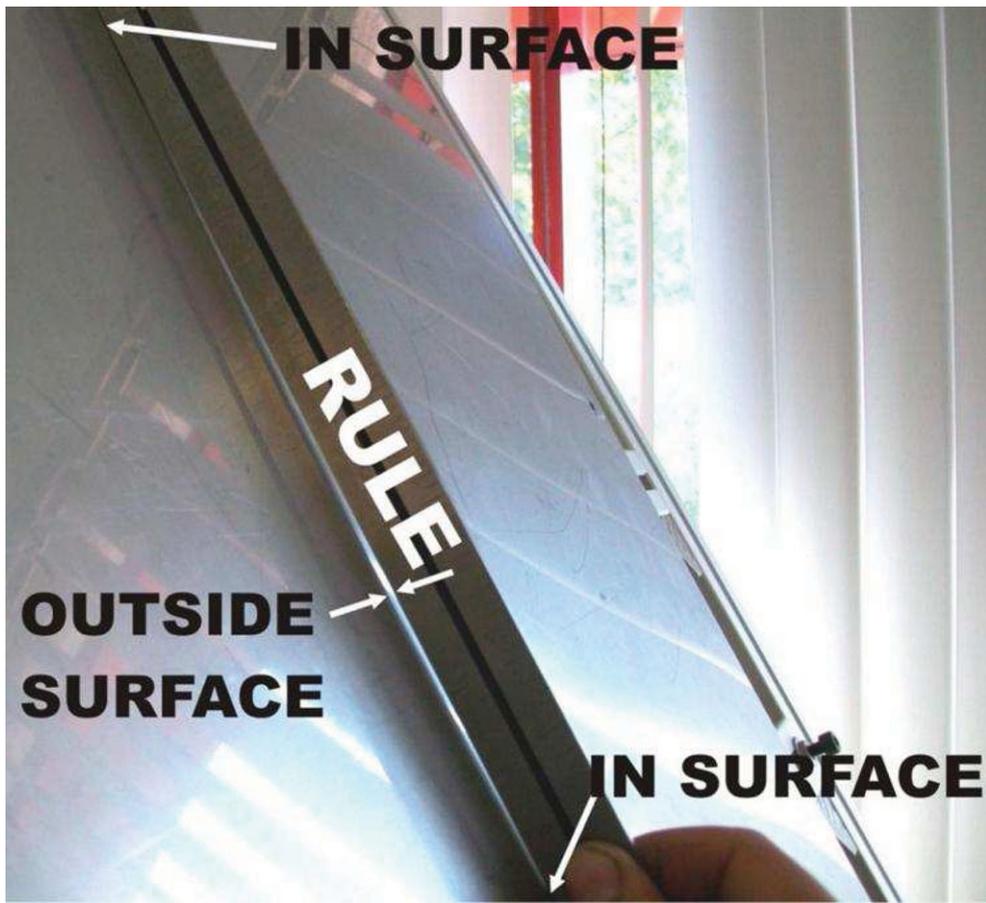


Fig. 14. (Colour online) Slightly concave surface.

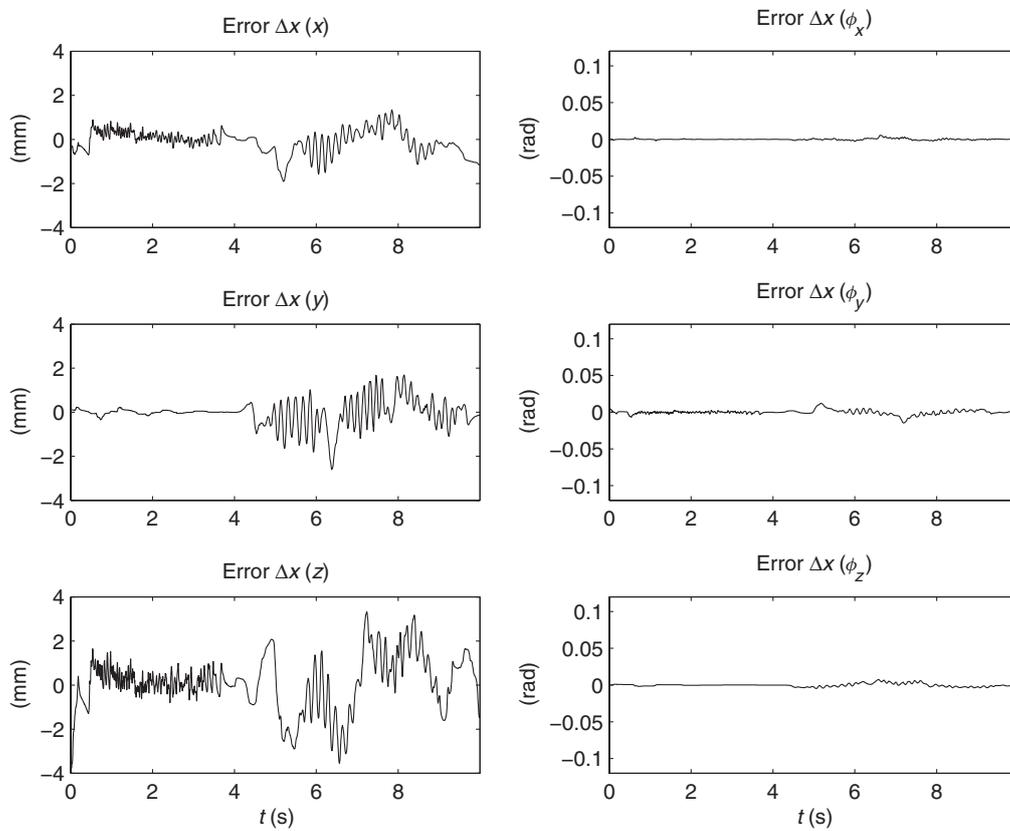


Fig. 15. Experiment 5. Testing robustness in the case of not well-known surfaces. Tracking error vector  $\Delta x$ .

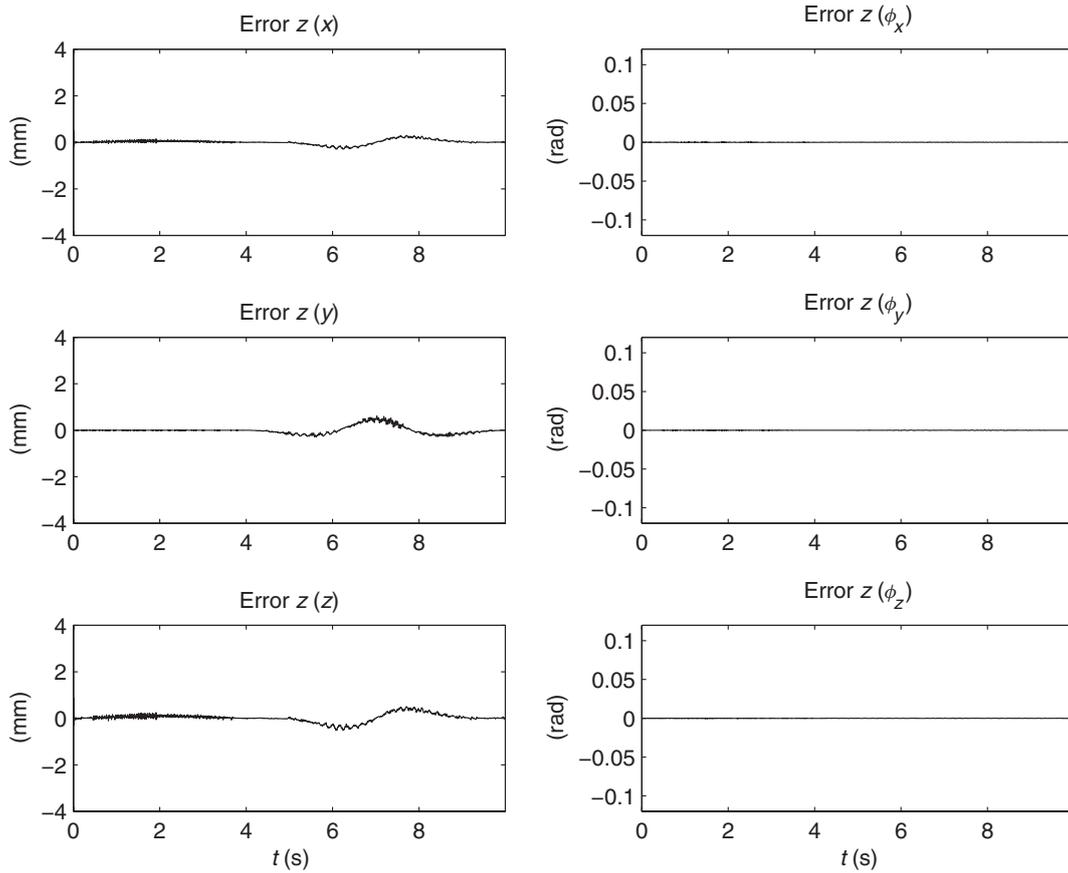


Fig. 16. Experiment 5. Testing robustness in the case of not well-known surfaces. Observer error vector  $z$ .

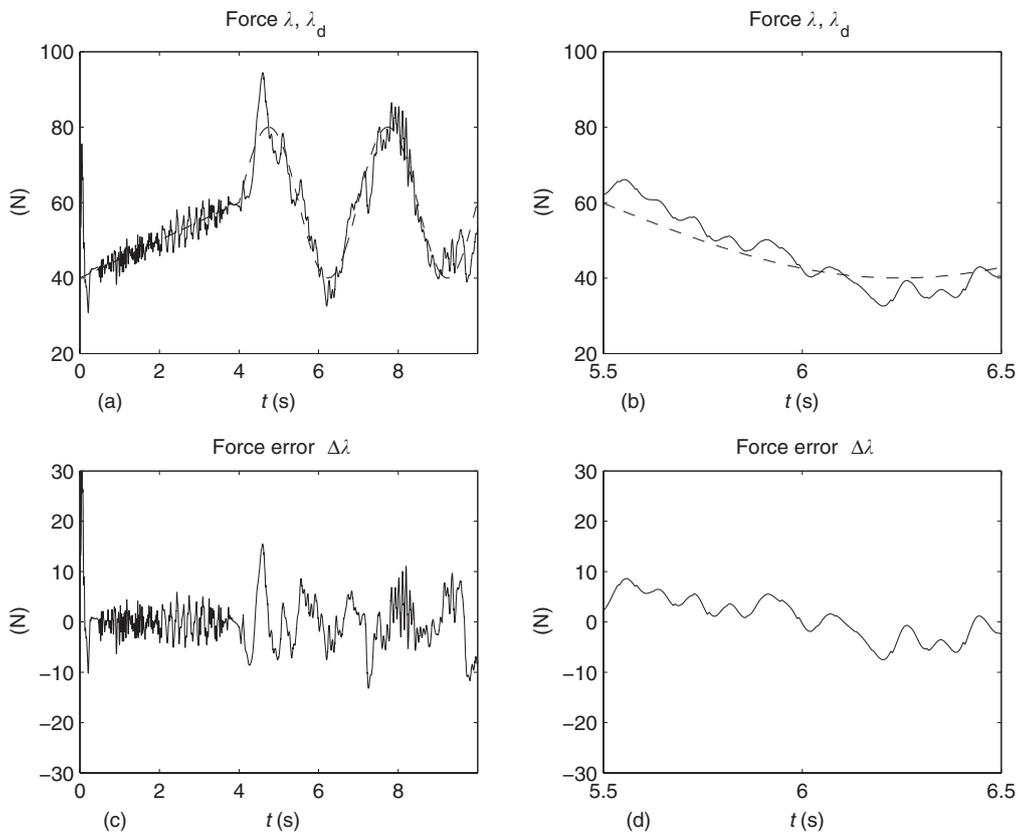


Fig. 17. Experiment 5. Testing robustness in the case of not well-known surfaces. (a–b) Desired force (---) versus real force (—); (c–d) force error  $\Delta\lambda$ .

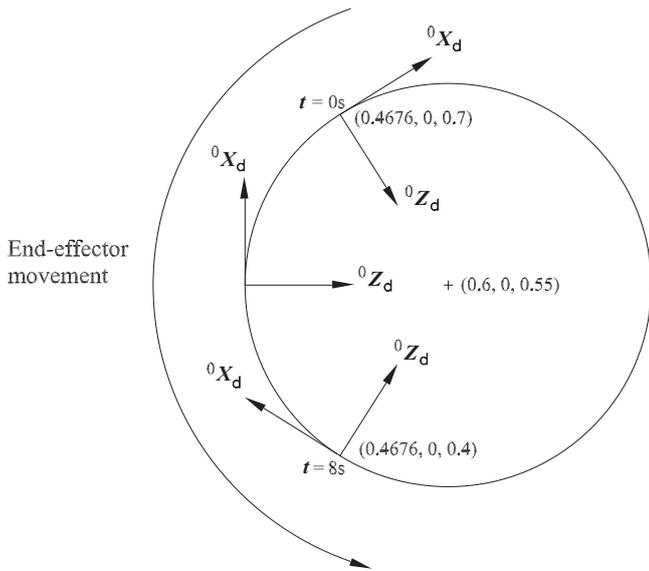


Fig. 18. Simulation. Desired trajectory for simulation.

of the experimental outcomes). The constraint is given by<sup>3</sup>

$$\varphi(\mathbf{x}) = \frac{1}{2r}(x - h)^2 + \frac{1}{2r}(y - k)^2 + \frac{1}{2r}(z - l)^2 - r/2 = 0, \tag{47}$$

with  $h = 0.6$  [m],  $k = 0$  [m],  $l = 0.55$  [m], and  $r = 0.2$  [m]. The sampling time has been chosen as 0.1 ms that allowed to set  $\mathbf{K}_p = \text{diag}\{0.75 \ 0.75 \ 0.75 \ 0.75 \ 0.75 \ 0.75\}$ ,  $\xi_1 = 20$ ,  $\xi_2 = 20$ ,  $\mathbf{A}_x = \text{diag}\{11,000 \ 11,000 \ 11,000 \ 11,000 \ 11,000 \ 11,000\}$ ,  $\mathbf{K}_\gamma = \text{diag}\{1.5 \ 1.5 \ 1.5 \ 1.5 \ 1.5 \ 1.5\}$ ,  $\mathbf{K}_\beta = \text{diag}\{0.5 \ 0.5 \ 0.5 \ 0.5 \ 0.5 \ 0.5\}$ ,  $k_d = 5000$ ,  $k_1 = 0.001$ ,  $k_\varepsilon = 5$ , and  $\mathbf{A}_z = \text{diag}\{5000 \ 5000 \ 5000 \ 5000 \ 5000 \ 5000\}$ .

**Remark 5.1.** By comparing the gains used for simulation and those employed in Section 4, it is easy to realize that most of the ones defined here are by far larger. In general, gains should be large enough to achieve a good performance and the user should begin with gains related to the observer, *i.e.*  $k_d$  and  $\mathbf{A}_z$ . Next  $\mathbf{A}_x$ ,  $\xi_1$ , and  $\xi_2$  should be set carefully large enough to diminish tracking errors. On the other hand,  $\mathbf{K}_p$  should be chosen not too large to avoid abrupt changes in the output. The integral gains should be kept zero until the tracking error becomes small enough. Only then the integral term should be included to achieve exact tracking.  $\Delta$

Figures 19 and 20 show tracking and observer errors, respectively. We kept the same scale as for the experimental results, this makes more evident that tracking has been accomplished. Recall that our approach guarantees only local stability so that we have chosen, just as for the experimental results, to set initial error conditions to zero. We do not consider this to be a disadvantage, since it is always possible

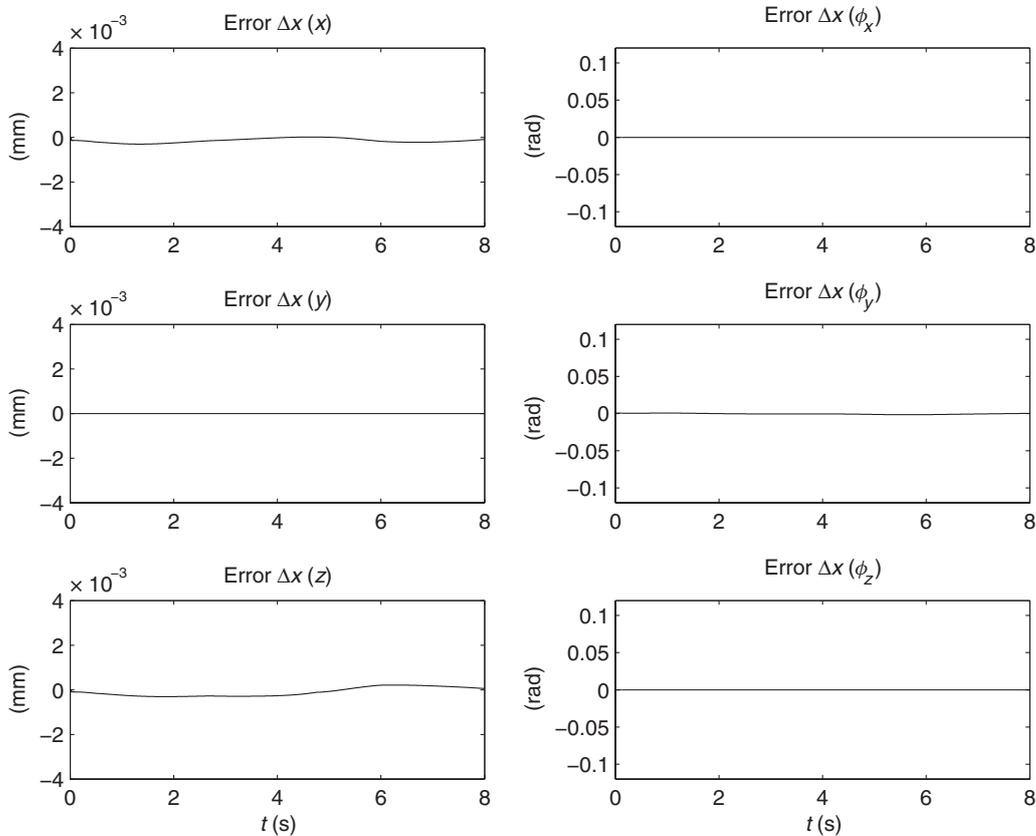


Fig. 19. Simulation. Tracking error vector  $\Delta\mathbf{x}$ .

<sup>3</sup> This definition allows to have  $\mathbf{J}_{\varphi_x} \mathbf{J}_{\varphi_x}^T = 1$ .

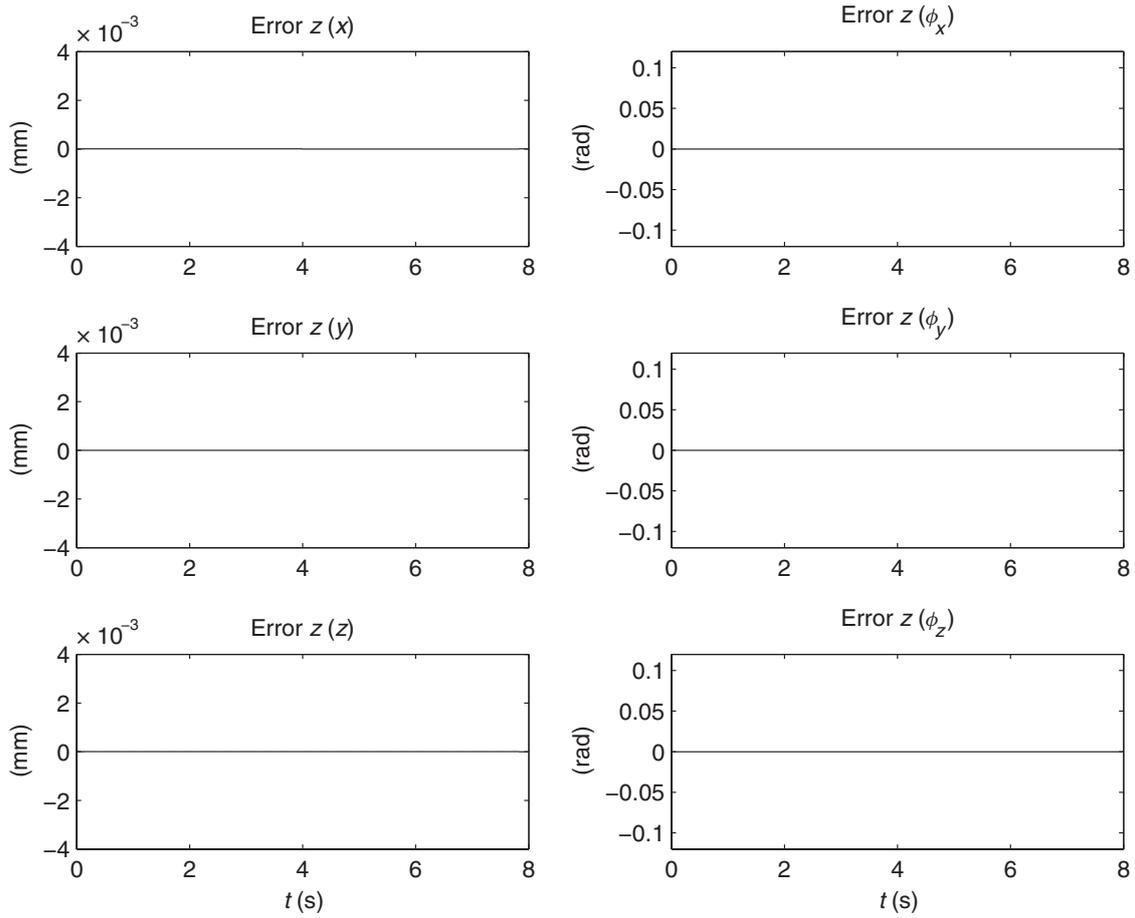


Fig. 20. Simulation. Observer error vector  $z$ .

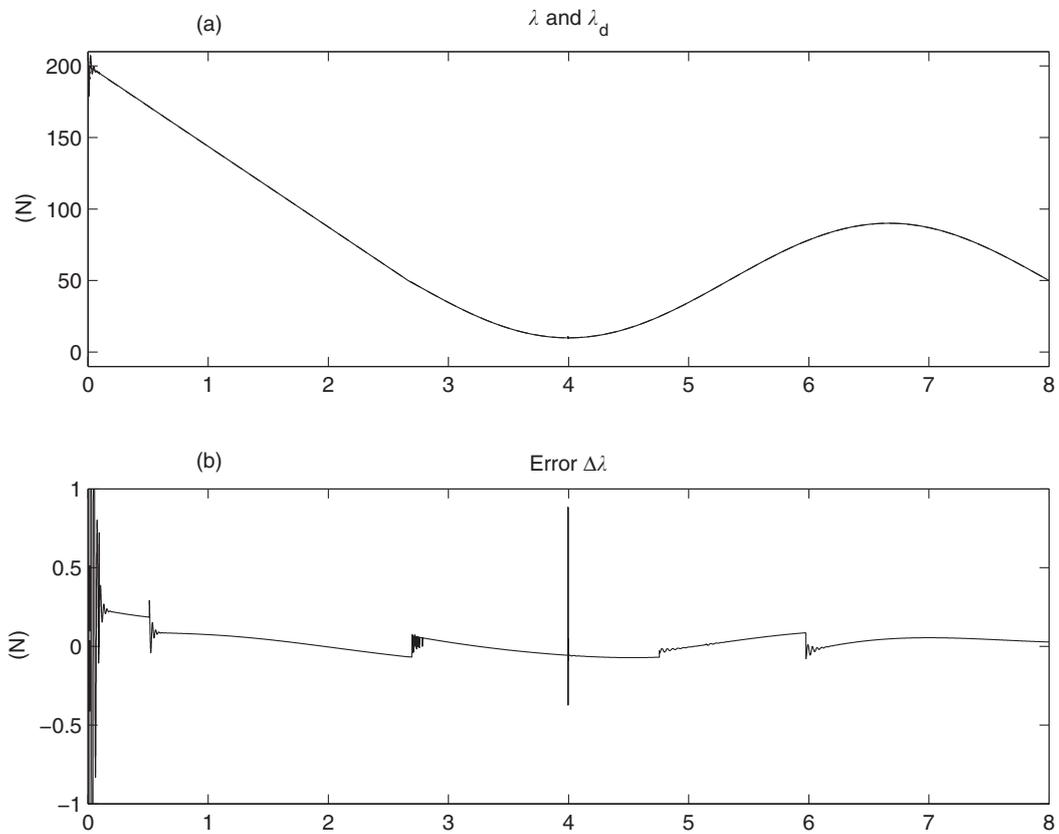


Fig. 21. Simulation. (a) Desired force (---) versus real force (—); (b) force error  $\Delta\lambda$ .

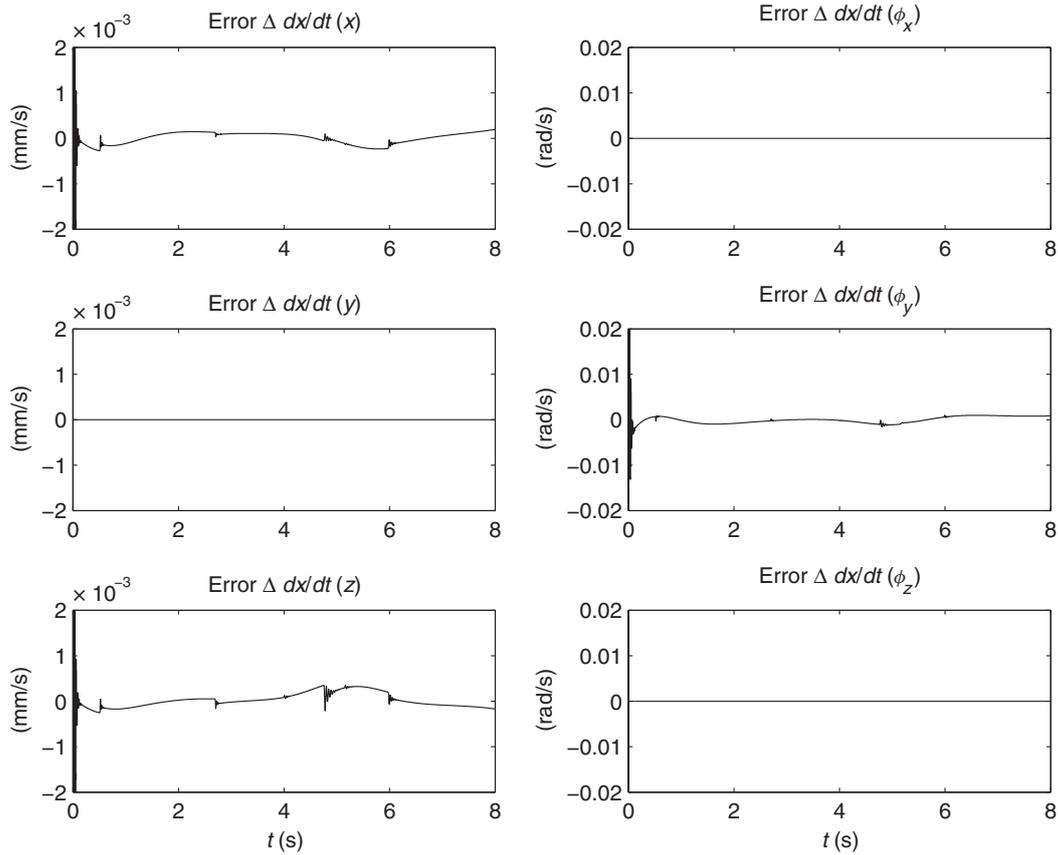


Fig. 22. Simulation. Velocity tracking error vector  $\Delta \dot{\mathbf{x}}$ .

to regulate the robot’s initial position and orientation to a specified value. As to the desired force, we were able to choose a more complex and larger reference (we do not risk to damage a sensor this time). The desired values go from 200 N to nearly 10 N. It can be appreciated in Fig. 21(a) how fast the error tends to zero, although in the end there is a small residual term of about 0.1 N (Fig. 21(b)). Still, we consider the outcome quite acceptable according to the theory.

Since for simulation joint and Cartesian velocities are available, in Figs. 22 and 23 the corresponding tracking and observation errors are shown. As could have been expected, these are also very good and in accordance with the theoretical development of Section 3. Finally, to show that exact orientation tracking is being accomplished (as it is indicated from the fact that tracking errors  $\Delta \mathbf{x}$  are zero), in Fig. 24 the orientation matrix  ${}^d \mathbf{R}_n = {}^0 \mathbf{R}_d^T {}^0 \mathbf{R}_n$  is shown. It can be checked that it is the unit matrix during the whole simulation, meaning that the end-effector acquires the desired orientation. Certainly, this can also be seen by drawing both  ${}^0 \mathbf{R}_d$  and  ${}^0 \mathbf{R}_n$  together, as done in Fig. 25. It can be recognized the rotation of the end-effector as depicted in Fig. 18.

**6. Conclusions**

Model-free tracking and force control for rigid robots are studied in this paper. Only unconstrained orientation motion

is considered. It is assumed that no velocity measurements are available so that an observer is introduced that delivers results directly in Cartesian coordinates. For implementation only the constraint equation and the manipulator Jacobian are necessary. To deal with the orientation problem, the unit quaternion has been employed. The control–observer scheme employs a second-order sliding mode variable to avoid the knowledge of the robot model. This characteristic allows to compensate for example friction terms without any previous identification. It has been shown that force, tracking, and observation errors tend to zero under the condition that no singularity is reached. This assumption is made for simplicity, but a subregion of the dextrous workspace free of singularities and proper gains can always be chosen to guarantee it. To simplify the mathematical stability analysis, it is assumed that the contact surface is a plane, but it is shown that the scheme also works for curved surfaces.

Experimental results are carried out to test the proposed algorithm. The outcomes are in acceptable agreement with the developed theory, both for flat and curve surfaces. However, because of hardware restrictions, the sampling time could not be set small enough to tune gains properly. This fact avoided to get even better results and certainly represents a disadvantage of the approach because it turns out to be sensitive to the discretization process. To show that with a faster sampling time outcomes can be improved, a complete simulation for the same robot manipulator in contact with the curved surface was carried out. For this case,

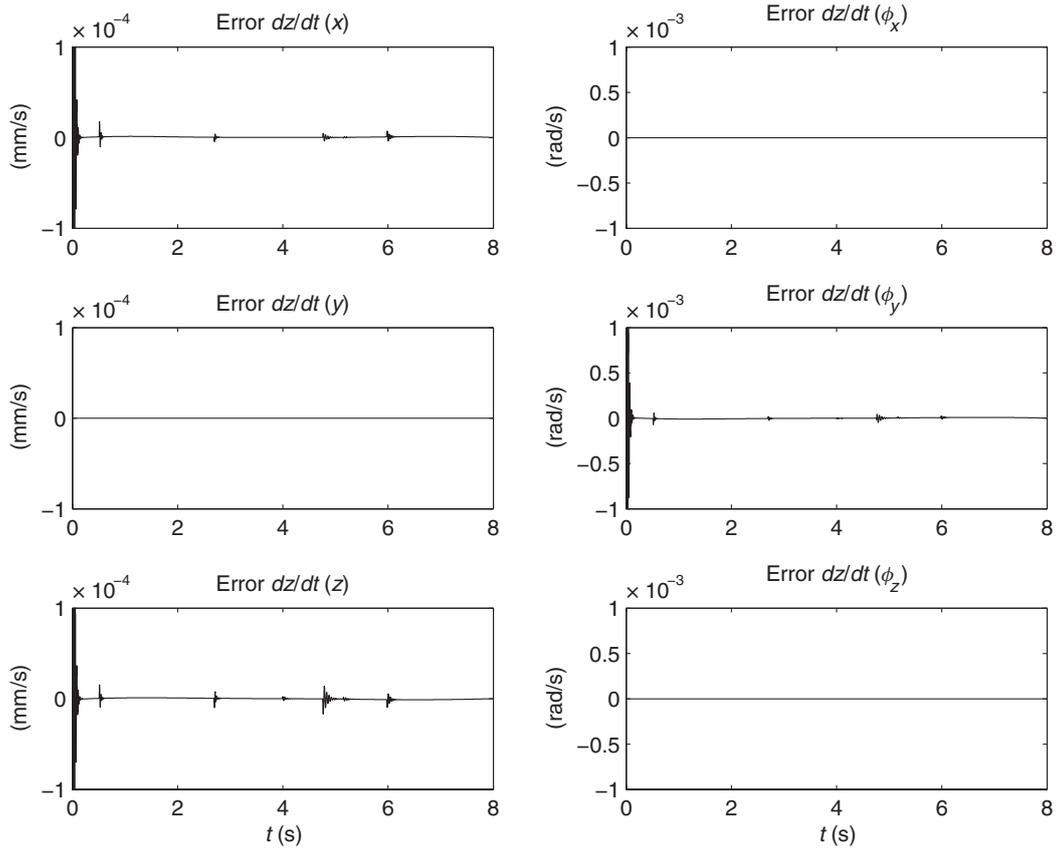


Fig. 23. Simulation. Velocity observer error vector  $\dot{z}$ .

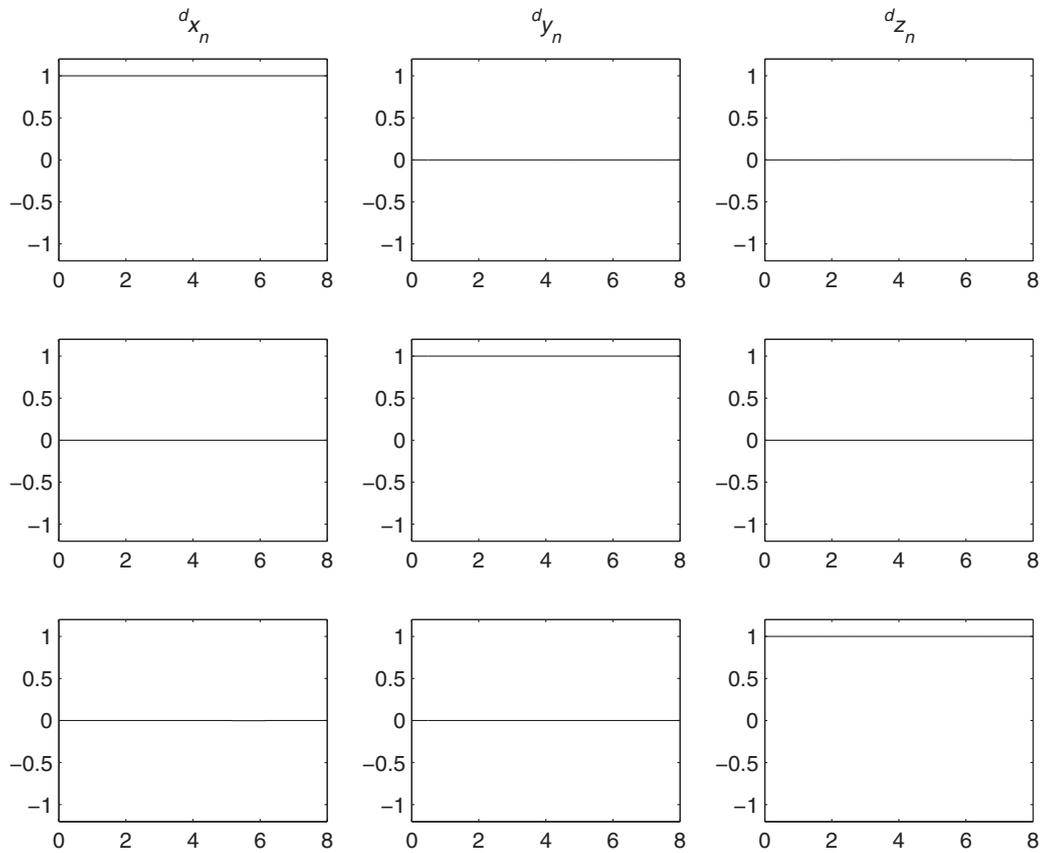


Fig. 24. Simulation. Orientation matrix  ${}^dR_n = [{}^d x_n \quad {}^d y_n \quad {}^d z_n]$ .

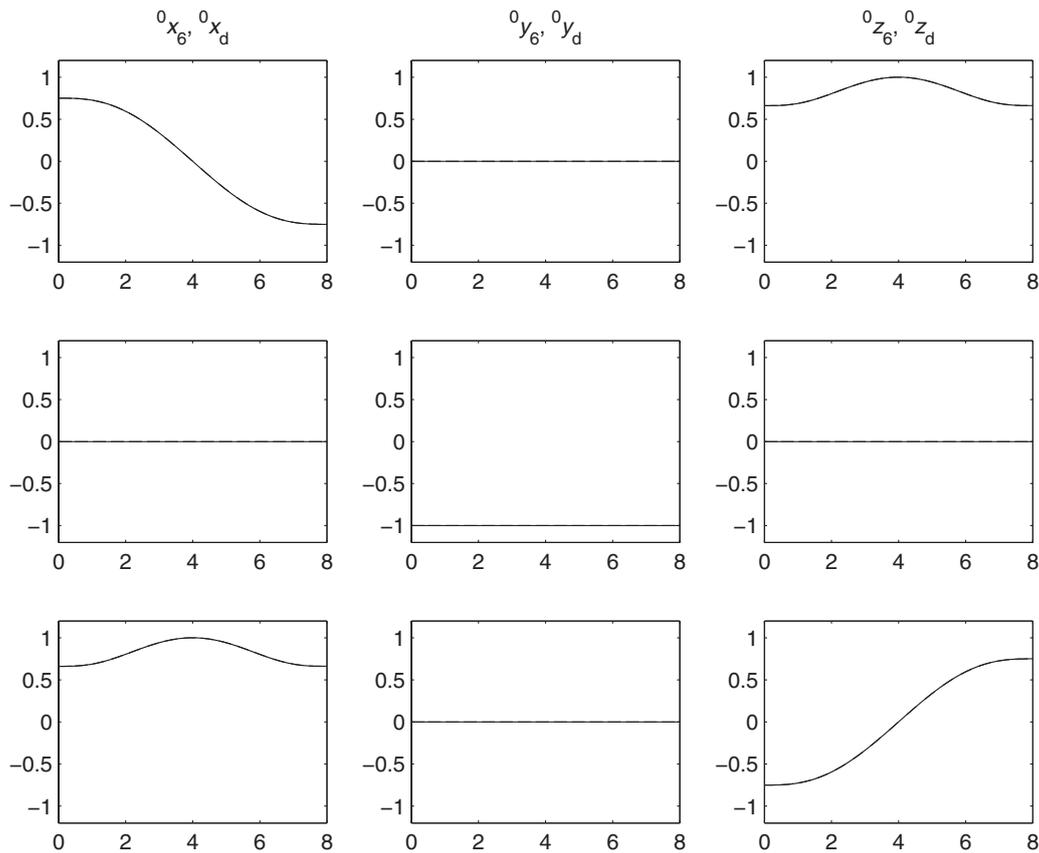


Fig. 25. Simulation.  ${}^0\mathbf{R}_n = [{}^0x_n \quad {}^0y_n \quad {}^0z_n]$  (—) and  ${}^0\mathbf{R}_d = [{}^0x_d \quad {}^0y_d \quad {}^0z_d]$  (- -).

the results were pretty well in accordance with the developed theory.

It remains as a future work an extension of the approach to torque control.

**Acknowledgments**

This work was supported by the DGAPA–UNAM under grant IN109611.

**References**

1. G. Zeng and A. Hemami, “An overview of robot force control,” *Robotica* **15**, 473–482 (1997).
2. B. Siciliano and L. Villani, *Robot Force Control* (Kluwer, The Netherlands, 1999).
3. T. Yoshikawa, “Force Control of Robot Manipulators,” *In: Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, USA (April 2000) pp. 220–226.
4. L. Sciavicco and B. Siciliano, *Modeling and Control of Robot Manipulators*, 2nd ed. (Springer–Verlag, London, UK, 2000).
5. Y. H. Liu and S. Arimoto, “Implicit and Explicit Force Controllers for Rheo–Holonomically Constrained Manipulators and Their Extension to Distributed Cooperation Control,” *In: IFAC 13th Triennial World Congress*, San Francisco, CA, USA (1996) pp. 618–623.
6. M. S. de Queiroz, D. M. Dawson and T. Burg, “Position/Force Control of Robot Manipulators Without Velocity/Force Measurements,” *In: Proceedings of the IEEE Conference on Robotics and Automation*, vol. 3, Minneapolis, MN, USA (April 1996) pp. 2561–2566.

7. J. C. Martínez-Rosas, M. A. Arteaga-Pérez and A. M. Castillo-Sánchez, “Decentralized control of cooperative robots without velocity-force measurements,” *Automatica* **42**, 329–336 (2006).
8. C. Canudas de Wit, H. Olsson, K. J. Åström and P. Lischinsky, “A new model for control of systems with friction,” *IEEE Trans. Autom. Control* **40**(3), 419–425 (1995).
9. E. Garcia, P. Gonzalez de Santos and C. Canudas de Wit, “Velocity dependence in the cyclic friction arising with gears,” *Int. J. Robot. Res.* **21**(9), 761–771 (2002).
10. M. A. Arteaga-Pérez and J. C. Rivera-Dueñas, “Force Control Without Inverse Kinematics nor Robot Model,” *In: Proceedings of the CD ROM, European Control Conference ECC07*, Kos Island, Greece (July 2007) pp. 4385–4392.
11. V. Parra-Vega, A. Rodríguez-Ángeles, S. Arimoto and G. Hirzinger, “High precision constrained grasping with cooperative adaptive hand control,” *J Intell. Robot. Syst.* **32**, 235–254 (2001).
12. M. A. Arteaga-Pérez, A. M. Castillo-Sánchez and V. Parra-Vega, “Cartesian control of robots without dynamic model and observer design,” *Automatica* **42**, 473–480 (2006).
13. H. K. Khalil, *Nonlinear Systems*, 3rd ed. (Prentice-Hall, Upper Saddle River, NJ, 2002).
14. R. M. Murray, Z. Li and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation* (CRC Press, Boca Raton, FL, 1994).
15. S. Jung and T. C. Hsia, “Reference Compensation Technique of Neural Force Tracking Impedance Control for Robot Manipulators,” *In: Proceedings of the 8th World Congress on Intelligent Control and Automation*, Jinan, China (Jul. 7–9, 2010) pp. 650–655.
16. C. C. Cheah, S. P. Hou, Y. Zhao and J.-J. E. Slotine, “Adaptive Vision and Force Tracking Control for Robots with Constraint

- Uncertainty,” *IEEE/ASME Trans. Mechatronics* **15**(3), 389–398 (2010).
17. J. C. Martínez-Rosas, “Identificación Paramétrica de un Robot Industrial con Diferentes Esquemas de Control Adaptable,” *Master’s Degree Thesis* (Universidad Veracruzana, Instituto de Ingeniería, Veracruz, Veracruz, Mexico, 2002).
  18. J. Gudiño-Lau and M. A. Arteaga-Pérez, “Dynamic model and simulation of cooperative robots: A case study,” *Robotica* **23**, 615–624 (2005).
  19. A. M. Castillo Sánchez, “Adaptación de Dos Robots Industriales Para su Utilización en el Desarrollo de Nuevas Técnicas y Algoritmos de Control,” *Bachelor’s Degree Thesis* (ENEP, Universidad Nacional Autónoma de México, Mexico, 2002).

**Appendix A: Proof of Theorem 3.1**

The proof of Theorem 3.1 is similar to that given in refs. [10, 12], so that we now present a sketch and stress few differences. First consider the next lemma.<sup>13</sup>

**Lemma 6.1.** If a differentiable function  $f(t)$  has a finite limit as  $t \rightarrow \infty$ , and if  $\dot{f}(t)$  is uniformly continuous, then it holds  $\dot{f}(t) \rightarrow 0$  as  $t \rightarrow \infty$ .  $\Delta$

As done in refs. [10, 12], we prove Theorem 3.1 in three steps: (a) The boundedness of  $\mathbf{y}$  in Eq. (43) implies the boundedness of any other closed loop variable. (b) With a proper choice of gains, the observer (23)–(24) and the control law (40) guarantee the boundedness of the state  $\mathbf{y}$  in closed loop. (c) If all closed loop variables are bounded, then the inclusion of the sign function in Eq. (31) makes all errors tend to zero.

(a) We have to show that if  $\|\mathbf{y}\|$  is bounded by  $0 < y_{\max} < \infty$ , then any other signal is bounded. As said before, the main difference of the algorithm presented here with that given in ref. [10] is a modification in the observer so that this part of the proof is all similar with step (a) of the proof in ref. [10]. However, in this work we present an alternative way to prove that the Lagrange multiplier  $\lambda$  is bounded. This term can be computed as explained in ref. [14] to get

$$\lambda = (J_\varphi H^{-1} J_\varphi^T)^{-1} \{ J_\varphi H^{-1} \{ \tau - \bar{\tau} \} + \dot{J}_\varphi \dot{q} \}, \quad (48)$$

where  $\bar{\tau} \triangleq C(q, \dot{q})\dot{q} + D\dot{q} + g(q) + \tau_p$  must be bounded because it depends only on bounded variables as shown in ref [14], and  $\tau_p$  is bounded by assumption. It only remains to show that the input torque  $\tau$  is bounded. By taking into account Eq. (40), and the fact that after Eq. (34) one has  $s_o = J(q)s_r - r$ , it is

$$\tau = -K_p(s_r - J^{-1}r) + J^T J_{\varphi x}^T \lambda_d - J^T J_{\varphi x}^T \xi_1 \Delta F,$$

which is bounded when  $\mathbf{y}$  is bounded.

(b) This part of the proof can also be carried out as explained in ref. [10] because the change in the observer does not essentially affect it.

(c) Till now we have only shown that  $\mathbf{y}$  is bounded. We still have to prove that tracking, force, and observation

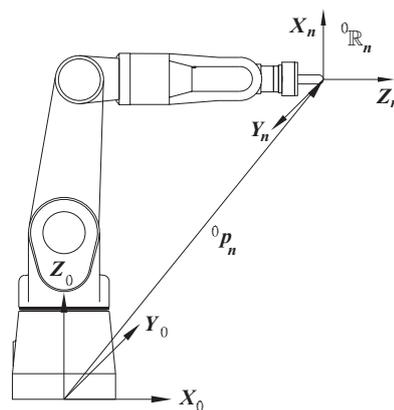


Fig. 26. Base and end-effector coordinate frames for robot A465 of *CRS robotics*.

errors tend to zero. Showing that  $\Delta x$ ,  $\Delta \dot{x}$ ,  $z$ ,  $\dot{z}$ , and  $\Delta F$  tend to zero can be done exactly the same way as done in step (c) of the proof in ref. [12]. One just have to take into account Eqs. (16) and (26)–(27). Then it only remains to show that  $\Delta \lambda \rightarrow 0$ . We use Lemma 6.1 for this. First of all, we know that  $\Delta F$  is bounded and has a limit ( $\Delta F \rightarrow 0$  as  $t \rightarrow \infty$ ). Then, in order for  $\Delta \lambda$  to tend to zero, we only have to show that it is uniformly continuous, or equivalently,  $\frac{d}{dt} \Delta \lambda$  is bounded. But, since  $\Delta \lambda = \lambda - \lambda_d$  and  $\lambda_d$  and its derivative are assumed to be bounded, it remains only to show that  $\frac{d}{dt} \lambda$  is bounded. It can be seen from Eq. (48) that  $\lambda = \lambda(q, \dot{q}, \dot{x}, \tau_p, s_r, r, \lambda_d, \Delta F)$ . Since the derivatives of all functions involved in model (1) exist, one has  $\frac{d}{dt} \lambda = \dot{\lambda}(q, \dot{q}, \ddot{q}, \dot{x}, \ddot{x}, \dot{\tau}_p, \dot{s}_r, \dot{r}, \dot{\lambda}_d, \dot{\Delta F}, \Delta \lambda)$ . By recalling that  $\dot{\tau}_p$  is bounded by assumption,  $\frac{d}{dt} \lambda$  must be bounded because after the discussion of step (a) all variables have been proven to be bounded, including  $\ddot{q}$  and  $\ddot{x}$ .  $\Delta$

**Appendix B: Kinematics and Dynamic Model of Robot A465 of *CRS Robotics***

The Cartesian position and orientation of the end-effector of the robot A465 of *CRS Robotics* used in Sections 4 and 5 for control implementation is computed by measuring joint positions and substituting them in the corresponding direct kinematics. To achieve this goal, the standard Denavit–Hartenberg approach has been employed.<sup>4</sup> The base and end-effector coordinate frames are shown in Fig. 26. For the sake of simplicity, the middle coordinate frames are not shown in the figure. The following homogeneous transformation matrices have been obtained:

$$A_1 = \begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2 c_2 \\ s_2 & c_2 & 0 & a_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} c_3 & 0 & s_3 & 0 \\ s_3 & 0 & -c_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Table I. Direct kinematic parameters of robot A465 of CRS Robotics.

$$d_1 = 0.330 \text{ (m)} \quad a_2 = 0.305 \text{ (m)} \quad d_4 = 0.330 \text{ (m)} \quad d_6 = 0.148 \text{ (m)}$$

Table II. Nomenclature for trigonometric functions.

$s_1 = \sin(q_1)$	$c_1 = \cos(q_1)$	$s_{23} = \sin(q_2 + q_3)$
$s_2 = \sin(q_2)$	$c_2 = \cos(q_2)$	$c_{23} = \cos(q_2 + q_3)$
$s_3 = \sin(q_3)$	$c_3 = \cos(q_3)$	
$s_4 = \sin(q_4)$	$c_4 = \cos(q_4)$	
$s_5 = \sin(q_5)$	$c_5 = \cos(q_5)$	
$s_6 = \sin(q_6)$	$c_6 = \cos(q_6)$	

$$A_4 = \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_5 = \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The relationship between the robot end-effector position and orientation with respect to the base frame is given by

$${}^0T_n = \begin{bmatrix} {}^0R_n & {}^0d_n \\ \mathbf{0}^T & 1 \end{bmatrix} = A_1 A_2 A_3 A_4 A_5 A_6.$$

Note that it is  ${}^0p_n = {}^0d_n$ . The values and definitions of different parameters can be read in Tables I and II. Note that inverse kinematics is not necessary for our approach. The elements of the associated geometrical Jacobian  $J(q)$  have been computed as

$$j_{11} = -d_6 s_5 s_1 c_{23} c_4 + d_6 s_5 c_1 s_4 + d_6 s_1 s_{23} c_5 - s_1 s_{23} d_4 - s_1 a_2 c_2,$$

$$j_{12} = -c_1 (d_6 s_{23} c_4 s_5 - d_6 c_{23} c_5 - c_{23} d_4 + a_2 s_2),$$

$$j_{13} = c_1 (-d_6 s_{23} c_4 s_5 + d_6 c_{23} c_5 + c_{23} d_4),$$

$$j_{14} = -d_6 s_5 (-s_1 c_4 + c_1 c_{23} s_4),$$

$$j_{15} = d_6 (c_1 c_4 c_{23} c_5 - c_1 s_{23} s_5 + s_4 s_1 c_5),$$

$$j_{16} = 0,$$

$$j_{21} = d_6 s_5 c_1 c_{23} c_4 + d_6 s_5 s_1 s_4 + d_6 c_1 s_{23} c_5 + c_1 s_{23} d_4 + c_1 a_2 c_2,$$

$$j_{22} = -s_1 (d_6 s_{23} c_4 s_5 - d_6 c_{23} c_5 - c_{23} d_4 + a_2 s_2),$$

$$j_{23} = s_1 (-d_6 s_{23} c_4 s_5 + d_6 c_{23} c_5 + c_{23} d_4),$$

$$j_{24} = -d_6 s_5 (s_1 c_{23} s_4 + c_1 c_4),$$

$$j_{25} = -d_6 (s_1 s_{23} s_5 + s_4 c_1 c_5 - s_1 c_4 c_{23} c_5),$$

$$j_{26} = 0,$$

$$j_{31} = 0,$$

$$j_{32} = d_6 s_5 c_{23} c_4 + d_6 s_{23} c_5 + s_{23} d_4 + a_2 c_2,$$

$$j_{33} = d_6 s_5 c_{23} c_4 + d_6 s_{23} c_5 + s_{23} d_4,$$

$$j_{34} = -s_{23} d_6 s_4,$$

$$j_{35} = d_6 (s_{23} c_4 c_5 + c_{23} s_5),$$

$$j_{36} = 0,$$

$$j_{41} = 0,$$

$$j_{42} = s_1,$$

$$j_{43} = s_1,$$

$$j_{44} = c_1 s_{23},$$

$$j_{45} = -c_1 c_{23} s_4 + s_1 c_4,$$

$$j_{46} = s_5 c_1 c_{23} c_4 + s_5 s_1 s_4 + c_1 s_{23} c_5,$$

$$j_{51} = 0,$$

$$j_{52} = -c_1,$$

$$j_{53} = -c_1,$$

$$j_{54} = s_1 s_{23},$$

$$j_{55} = -s_1 c_{23} s_4 - c_1 c_4,$$

$$j_{56} = s_5 s_1 c_{23} c_4 - s_5 c_1 s_4 + s_1 s_{23} c_5,$$

$$j_{61} = 1,$$

$$j_{62} = 0,$$

$$j_{63} = 0,$$

$$j_{64} = -c_{23},$$

$$j_{65} = -s_{23} s_4,$$

$$j_{66} = s_{23} c_4 s_5 - c_{23} c_5.$$

In order to calculate the dynamic model of the robot, the Euler–Lagrange method was used.<sup>4</sup> To have a simulation as accurate as possible, motors’ dynamics were also included, resulting in<sup>17</sup>

$$(H(q) + D_n^{-1} D_j) \ddot{q} + C(q, \dot{q}) \dot{q} + (D + D_n^{-1} D_f) \dot{q} + g(q) = D_n^{-1} D_K V - J^T(q) (J_{\varphi x}^T \lambda + f_f).$$

$f_f \in \mathbb{R}^n$  represents the friction between the robot end-effector and the contact surface.  $D_n$ ,  $D_j$ ,  $D_f$ , and  $D_K \in \mathbb{R}^{n \times n}$  are all diagonal matrices given by

$$D_n = \text{diag} \left\{ \frac{1}{r_1^2} \quad \cdots \quad \frac{1}{r_6^2} \right\},$$

$$D_j = \text{diag} \{ J_{m1} \quad \cdots \quad J_{m6} \},$$

$$D_f = \text{diag} \left\{ f_{m1} + \frac{K_{a1} K_{b1}}{R_{a1}} \quad \cdots \quad f_{m6} + \frac{K_{a6} K_{b6}}{R_{a6}} \right\},$$

$$D_K = \text{diag} \left\{ \frac{K_{a1}}{R_{a1} r_1} \quad \cdots \quad \frac{K_{a6}}{R_{a6} r_6} \right\}.$$

For  $i = 1, \dots, 6$ ,  $r_i$  is the gear ratio,  $J_{m1}$  is the rotor inertia,  $f_{m1}$  is the rotor friction coefficient,  $K_{a1}$  is the torque constant,  $R_{a1}$  is the armature resistance, and  $K_{b1}$  is the back emf constant. The corresponding values can be read in Table III.

By recalling that the inertia matrix is symmetric, the elements of the lower part of  $H(q) \in \mathbb{R}^{n \times n}$  are given by

Table III. Motors parameters.

Motor	1	2	3	4	5	6
$r_i$	100	100	100	101	100	101
$J_{m1}$ (kg m <sup>2</sup> )	$9.0376 \cdot 10^{-5}$	$9.0376 \cdot 10^{-5}$	$9.0376 \cdot 10^{-5}$	$4.9 \cdot 10^{-6}$	$4.9 \cdot 10^{-6}$	$4.9 \cdot 10^{-6}$
$f_{m1}$ (kg m <sup>2</sup> /s)	$9.70996 \cdot 10^{-5}$	$9.70996 \cdot 10^{-5}$	$9.70996 \cdot 10^{-5}$	0.02234	0.02234	$4.6792 \cdot 10^{-5}$
$K_{a1}$ (N m/A)	0.14234	0.14234	0.14234	0.053	0.053	0.0392
$R_{a1}$ ( $\Omega$ )	0.84	0.84	0.84	2.7	2.7	6.9
$K_{b1}$ (V s/rad)	0.14229	0.14229	0.14229	0.0534	0.0534	0.0392

$$\begin{aligned}
 h_{11} = & m_2 l_{c2}^2 c_2^2 + m_3 (a_2 c_2 + l_{c3} s_{23})^2 + m_4 (a_2 c_2 + l_{c4} s_{23})^2 && - I_{422} c_{23} - I_{511} s_5 (s_{23} c_4 c_5 + c_{23} s_5) + I_{533} c_5 (s_{23} c_4 s_5 \\
 & + m_5 ((a_2 c_2 + l_{c5} s_{23} c_5 + d_4 s_{23} + l_{c5} c_{23} c_4 s_5)^2 + l_{c5}^2 s_4^2 s_5^2) && - c_{23} c_5) - I_{611} s_5 c_6 (s_{23} c_4 c_5 c_6 + c_{23} s_5 c_6 - s_{23} s_4 s_6) \\
 & + m_6 ((a_2 c_2 + l_{c6} s_{23} c_5 + d_4 s_{23} + l_{c6} c_{23} c_4 s_5)^2 + l_{c6}^2 s_4^2 s_5^2) && - I_{622} s_5 s_6 (s_{23} c_4 c_5 s_6 + c_{23} s_5 s_6 + s_{23} s_4 c_6) \\
 & + I_{122} + I_{211} s_2^2 + I_{222} c_2^2 + I_{311} s_{23}^2 + I_{333} c_{23}^2 && + I_{633} c_5 (s_{23} c_4 s_5 - c_{23} c_5), \\
 & + I_{411} s_{23}^2 c_4^2 + I_{422} c_{23}^2 + I_{433} s_{23}^2 s_4^2 + I_{511} (s_{23} c_4 c_5 && h_{51} = -m_5 l_{c5} s_4 (s_{23} (l_{c5} + d_4 c_5) + a_2 c_2 c_5) \\
 & + c_{23} s_5)^2 + I_{522} s_{23}^2 s_4^2 + I_{533} (s_{23} c_4 s_5 - c_{23} c_5)^2 && - m_6 l_{c6} s_4 (s_{23} (l_{c6} + d_4 c_5) + a_2 c_2 c_5) - s_{23} s_4 I_{522} \\
 & + I_{611} (s_{23} c_4 c_5 c_6 + c_{23} s_5 c_6 - s_{23} s_4 s_6)^2 + I_{622} (s_{23} c_4 c_5 s_6 && + I_{611} s_6 (s_{23} c_4 c_5 c_6 + c_{23} s_5 c_6 - s_{23} s_4 s_6) \\
 & + c_{23} s_5 s_6 + s_{23} s_4 c_6)^2 + I_{633} (s_{23} c_4 s_5 - c_{23} c_5)^2, && - I_{622} c_6 (s_{23} c_4 c_5 s_6 + c_{23} s_5 s_6 + s_{23} s_4 c_6), \\
 h_{21} = & m_5 l_{c5} s_4 s_5 (c_{23} (d_4 + l_{c5} c_5) - a_2 s_2 - l_{c5} s_{23} c_4 s_5) && h_{61} = (s_{23} c_4 s_5 - c_{23} c_5) I_{633}, \\
 & + m_6 l_{c6} s_4 s_5 (c_{23} (d_4 + l_{c6} c_5) - a_2 s_2 - l_{c6} s_{23} c_4 s_5) && \\
 & + s_{23} c_4 s_4 (I_{411} - I_{433}) + I_{511} s_4 c_5 (s_{23} c_4 c_5 + c_{23} s_5) && h_{22} = m_2 l_{c2}^2 + m_3 (a_2^2 + l_{c3}^2 + 2a_2 l_{c3} s_3) \\
 & - I_{522} s_{23} s_4 c_4 + I_{533} s_4 s_5 (s_{23} c_4 s_5 - c_{23} c_5) && + m_4 (a_2^2 + l_{c4}^2 + 2a_2 l_{c4} s_3) \\
 & + I_{611} (s_{23} c_4 c_5 c_6 + c_{23} s_5 c_6 - s_{23} s_4 s_6) (s_4 c_5 c_6 + c_4 s_6) && + m_5 (a_2^2 + (s_5 c_4 l_{c5})^2 + (d_4 + c_5 l_{c5})^2 + 2a_2 s_3 (d_4 \\
 & + I_{622} (s_{23} c_4 c_5 s_6 + c_{23} s_5 s_6 + s_{23} s_4 c_6) (s_4 c_5 s_6 - c_4 c_6) && + c_5 l_{c5}) + 2l_{c5} a_2 c_3 c_4 s_5) + m_6 (a_2^2 + (s_5 c_4 l_{c6})^2 \\
 & + I_{633} s_4 s_5 (s_{23} c_4 s_5 - c_{23} c_5), && + (d_4 + c_5 l_{c6})^2 + 2a_2 s_3 (d_4 + c_5 l_{c6}) + 2l_{c6} a_2 c_3 c_4 s_5) \\
 h_{31} = & -m_5 l_{c5} s_4 s_5 (l_{c5} s_{23} c_4 s_5 - c_{23} (d_4 + l_{c5} c_5)) && + I_{233} + I_{322} + I_{411} s_4^2 + I_{433} c_4^2 + I_{511} s_4^2 c_5^2 \\
 & - m_6 l_{c6} s_4 s_5 (l_{c6} s_{23} c_4 s_5 - c_{23} (d_4 + l_{c6} c_5)) && + I_{522} c_4^2 + I_{533} s_4^2 s_5^2 + I_{611} (s_4 c_5 c_6 + c_4 s_6)^2 \\
 & + s_{23} c_4 s_4 (I_{411} - I_{433}) + I_{511} s_4 c_5 (s_{23} c_4 c_5 + c_{23} s_5) && + I_{622} (s_4 c_5 s_6 - c_4 c_6)^2 + I_{633} s_4^2 s_5^2, \\
 & - I_{522} s_{23} s_4 c_4 + I_{533} s_4 s_5 (s_{23} c_4 s_5 - c_{23} c_5) && h_{32} = m_3 l_{c3} (l_{c3} + a_2 s_3) + m_4 l_{c4} (l_{c4} + a_2 s_3) \\
 & + I_{611} (s_{23} c_4 c_5 c_6 + c_{23} s_5 c_6 - s_{23} s_4 s_6) (s_4 c_5 c_6 + c_4 s_6) && + m_5 ((d_4 + l_{c5} c_5)^2 + a_2 s_3 (d_4 + l_{c5} c_5) + (l_{c5} c_4 s_5)^2 \\
 & + I_{622} (s_{23} c_4 c_5 s_6 + c_{23} s_5 s_6 + s_{23} s_4 c_6) (s_4 c_5 s_6 - c_4 c_6) && + l_{c5} a_2 c_3 c_4 s_5) \\
 & + I_{633} s_4 s_5 (s_{23} c_4 s_5 - c_{23} c_5), && + m_6 ((d_4 + l_{c6} c_5)^2 + a_2 s_3 (d_4 + l_{c6} c_5) + (l_{c6} c_4 s_5)^2 \\
 h_{41} = & -m_5 l_{c5} s_5 (l_{c5} c_{23} s_5 + a_2 c_2 c_4 + s_{23} c_4 (d_4 + l_{c5} c_5)) && + l_{c6} a_2 c_3 c_4 s_5) + I_{322} + I_{411} s_4^2 + I_{433} c_4^2 \\
 & - m_6 l_{c6} s_5 (l_{c6} c_{23} s_5 + a_2 c_2 c_4 + s_{23} c_4 (d_4 + l_{c6} c_5)) && + I_{511} s_4^2 c_5^2 + I_{522} c_4^2 + I_{533} s_4^2 s_5^2 + I_{611} (s_4 c_5 c_6 + c_4 s_6)^2
 \end{aligned}$$

Table IV. Masses, friction coefficients, moments of inertia, and centers of mass.

$l_{c1} = 0.150$ (m)	$l_{c2} = 0.140$ (m)	$l_{c3} = 0.070$ (m)	$l_{c4} = 0.140$ (m)	$l_{c5} = 0.03$ (m)	$l_{c6} = 0.04$ (m)
$m_1 = 18.3$ (kg)	$m_2 = 15$ (kg)	$m_3 = 13.5$ (kg)	$m_4 = 10.8$ (kg)	$m_5 = 5.8$ (kg)	$m_6 = 1$ (kg)
$I_{111} = 0.8$ (kg m <sup>2</sup> )	$I_{122} = 0.7$ (kg m <sup>2</sup> )	$I_{133} = 0.9$ (kg m <sup>2</sup> )	$I_{211} = 0.85$ [kg m <sup>2</sup> ]	$I_{222} = 0.8$ [kg m <sup>2</sup> ]	$I_{233} = 0.75$ [kg m <sup>2</sup> ]
$I_{311} = 0.70$ [kg m <sup>2</sup> ]	$I_{322} = 0.75$ [kg m <sup>2</sup> ]	$I_{333} = 0.60$ [kg m <sup>2</sup> ]	$I_{411} = 0.45$ [kg m <sup>2</sup> ]	$I_{422} = 0.50$ [kg m <sup>2</sup> ]	$I_{433} = 0.40$ [kg m <sup>2</sup> ]
$I_{511} = 0.18$ [kg m <sup>2</sup> ]	$I_{522} = 0.20$ [kg m <sup>2</sup> ]	$I_{533} = 0.15$ (kg m <sup>2</sup> )	$I_{611} = 0.10$ [kg m <sup>2</sup> ]	$I_{622} = 0.08$ [kg m <sup>2</sup> ]	$I_{633} = 0.09$ [kg m <sup>2</sup> ]
$d_{11} = 3$ [kg m <sup>2</sup> /s]	$d_{22} = 2.6$ [kg m <sup>2</sup> /s]	$d_{33} = 2.5$ [kg m <sup>2</sup> /s]	$d_{44} = 1.8$ [kg m <sup>2</sup> /s]	$d_{55} = 1.5$ [kg m <sup>2</sup> /s]	$d_{66} = 1.2$ [kg m <sup>2</sup> /s]
$b_x = 0.61$ [N s]	$b_y = 0.61$ [N s]	$b_z = 0.61$ [N s]			
$cb_x = 0.47$ (N)	$cb_y = 0.47$ [N]	$cb_z = 0.47$ [N]			

$$\begin{aligned}
 &+ I_{622}(s_4c_5s_6 - c_4c_6)^2 + I_{633}s_4^2s_5^2, \\
 h_{42} = &-m_5l_{c_5}s_4s_5(d_4 + l_{c_5}c_5 + a_2s_3) - m_6l_{c_6}s_4s_5(d_4 + l_{c_6}c_5 \\
 &+ a_2s_3) + s_4c_5s_5(I_{533} - I_{511}) \\
 &- I_{611}s_5c_6(s_4c_5c_6 + c_4s_6) - I_{622}s_5s_6(s_4c_5s_6 - c_4c_6) \\
 &+ I_{633}s_4s_5c_5, \\
 h_{52} = &m_5l_{c_5}c_4(l_{c_5} + d_4c_5) + m_5l_{c_5}a_2(s_3c_4c_5 + c_3s_5) \\
 &+ m_6l_{c_6}c_4(l_{c_6} + d_4c_5) + m_6l_{c_6}a_2(s_3c_4c_5 + c_3s_5) \\
 &+ I_{522}c_4 + I_{611}s_6(s_4c_5c_6 + c_4s_6) \\
 &- I_{622}c_6(s_4c_5s_6 - c_4c_6),
 \end{aligned}$$

$$h_{62} = s_5s_4I_{633},$$

$$\begin{aligned}
 h_{33} = &m_3l_{c_3}^2 + m_4l_{c_4}^2 + m_5(d_4 + l_{c_5}c_5)^2 + m_5(l_{c_5}c_4s_5)^2 \\
 &+ m_6(d_4 + l_{c_6}c_5)^2 + m_6(l_{c_6}c_4s_5)^2 + I_{322} + I_{411}s_4^2 \\
 &+ I_{433}c_4^2 + I_{511}s_4^2c_5^2 + I_{522}c_4^2 + I_{533}s_4^2s_5^2 + I_{611}(s_4c_5c_6 \\
 &+ c_4s_6)^2 + I_{622}(s_4c_5s_6 - c_4c_6)^2 + I_{633}s_4^2s_5^2,
 \end{aligned}$$

$$\begin{aligned}
 h_{43} = &-m_5l_{c_5}s_4s_5(d_4 + l_{c_5}c_5) - m_6l_{c_6}s_4s_5(d_4 + l_{c_6}c_5) \\
 &+ s_4c_5s_5(I_{533} - I_{511}) - I_{611}s_5c_6(s_4c_5c_6 + c_4s_6) \\
 &- I_{622}s_5s_6(s_4c_5s_6 - c_4c_6) + I_{633}s_4s_5c_5,
 \end{aligned}$$

$$\begin{aligned}
 h_{53} = &m_5l_{c_5}c_4(l_{c_5} + d_4c_5) + m_6l_{c_6}c_4(l_{c_6} + d_4c_5) + I_{522}c_4 \\
 &+ I_{611}s_6(s_4c_5c_6 + c_4s_6) - I_{622}c_6(s_4c_5s_6 - c_4c_6),
 \end{aligned}$$

$$h_{63} = s_4s_5I_{633},$$

$$\begin{aligned}
 h_{44} = &m_5(l_{c_5}s_5)^2 + m_6(l_{c_6}s_5)^2 + I_{422} + I_{511}s_5^2 + I_{533}c_5^2 \\
 &+ I_{611}s_5^2c_6^2 + I_{622}s_5^2s_6^2 + I_{633}c_5^2,
 \end{aligned}$$

$$h_{54} = s_5c_6s_6(I_{622} - I_{611}),$$

$$h_{64} = I_{633}c_5,$$

$$h_{55} = m_5l_{c_5}^2 + m_6l_{c_6}^2 + I_{522} + I_{611}s_6^2 + I_{622}c_6^2,$$

$$h_{65} = 0,$$

$$h_{66} = I_{633}.$$

The matrix  $C(q, \dot{q})$  can be computed by using the Christoffel symbols after  $H(q)$  as explained in refs. [4, 17]. We omit it here for lack of room. The elements of the gravity vector  $g(q) \in \mathbb{R}^n$  are given by

$$g_1 = 0,$$

$$\begin{aligned}
 g_2 = &m_2gl_{c_2}c_2 + m_3g(a_2c_2 + l_{c_3}s_{23}) + m_4g(a_2c_2 + l_{c_4}s_{23}) \\
 &+ m_5g(a_2c_2 + d_4s_{23} + (c_{23}c_4s_5 + s_{23}c_5)l_{c_5}) \\
 &+ m_6g(a_2c_2 + d_4s_{23} + (c_{23}c_4s_5 + s_{23}c_5)l_{c_6}),
 \end{aligned}$$

$$\begin{aligned}
 g_3 = &m_3gl_{c_3}s_{23} + m_4gl_{c_4}s_{23} + m_5g(d_4s_{23} + (c_{23}c_4s_5 \\
 &+ s_{23}c_5)l_{c_5}) + m_6g(d_4s_{23} + (c_{23}c_4s_5 + s_{23}c_5)l_{c_6}),
 \end{aligned}$$

$$g_4 = -m_5gs_{23}s_4s_5l_{c_5} - m_6gs_{23}s_4s_5l_{c_6},$$

$$g_5 = m_5g(s_{23}c_4c_5 + c_{23}s_5)l_{c_5} + m_6g(s_{23}c_4c_5 + c_{23}s_5)l_{c_6},$$

$$g_6 = 0,$$

where  $g = 9.81(\text{m/s}^2)$  is the gravity constant. The values of the masses, moments of inertia, and center mass positions can be seen in Table IV. Besides the motor friction coefficients  $D_f$ , we consider only viscous frictions. As to the matrix  $D \in \mathbb{R}^{n \times n}$ , it is given by

$$D = \text{diag}\{d_{11} \ \dots \ d_{66}\}. \tag{49}$$

To model friction between the surface and the robot end-effector, it is considered that the former is made out of steel, while the later of aluminium so that it is  $f_f = B_v \dot{x} + B_C \text{sign}(\dot{x})$ , with

$$B_v = \text{diag}\{b_x \ b_y \ b_z \ 0 \ 0 \ 0\}, \tag{50}$$

$$B_C = \text{diag}\{cb_x \ cb_y \ cb_z \ 0 \ 0 \ 0\}. \tag{51}$$

Different parameters for  $D$ ,  $B_v$ , and  $B_C$  are given in Table IV. The values for  $B_v$  and  $B_C$  can readily be found by a simple Internet search.

The Lagrange multiplier was computed according to

$$\begin{aligned}
 \lambda = &(J_{\varphi_x} J(q) (H(q) + D_n^{-1} D_j)^{-1} J(q)^T J_{\varphi_x}^T)^{-1} \\
 &\times [-\ddot{\varphi}(x) + J_{\varphi_x} \dot{J}(q) \dot{q} + \dot{J}_{\varphi_x} J(q) \dot{q} \\
 &+ J_{\varphi_x} J(q) (H(q) + D_n^{-1} D_j)^{-1} (\tau - C(q, \dot{q}) \dot{q} \\
 &- (D + D_n^{-1} D_f) \dot{q} - g(q) - J^T(q) f_f)], \tag{52}
 \end{aligned}$$

where

$$J_{\varphi_x} = \frac{1}{r} [x - h \quad y - k \quad z - l \quad 0 \quad 0 \quad 0]. \tag{53}$$

Note that instead of substituting directly the second derivative of constraint (47), the following linear stable equation was implemented:

$$\ddot{\varphi}(x) + \alpha \dot{\varphi}(x) + \beta \varphi(x) = 0, \tag{54}$$

with  $\alpha = 300$  and  $\beta = 22500$ . See ref. [18] for details.