6

Cartesian Controllers for Tracking of Robot Manipulators under Parametric Uncertainties

R. García-Rodríguez and P. Zegers
College of Engineering and Applied Sciences, Universidad de los Andes
Chile

1. Introduction

The relevance of robot manipulators in different processes has created the need to design efficient controllers with low computational costs. Although several applications for this problem are defined in operational coordinates, a wide variety of controllers reported in the literature are defined in joint coordinates. Then, for a joint robot control the desired joint references are computed from desired Cartesian coordinates using inverse mappings and its derivatives up to second order. However, computing the inverse kinematics mappings is difficult due to the ill-posed nature of these mappings.
To circumvent the computation of inverse kinematics, a very old but not less important approach coined as Cartesian control can be used. Cartesian control deals with the problem of designing controllers in terms of desired Cartesian or operational coordinates. This allows saving a significant amount of time in real time applications due to the inherent simplification.

1.1 Cartesian control

Based on the seminal work of Miyazaki and Masutani [Miyazaki & Masutani (1990)] have been presented several approaches for regulating tasks, working with the assumption that the Jacobian is uncertain. Several approaches for setpoint control are presented [Yazarel & Cheah (2001)], [Chea et.al. (1999)], [Chea et.al. (2001)] [Huang et.al. (2002)], [Chea et.al. (2004)], assuming that the jacobian matrix can be parameterized linearly. Now, if we are interested that having the end effector of the robot manipulator follow a desired trajectory, Cartesian robot dynamics knowledge is required. However, Cartesian robot dynamics demands even more computational power than computing the inverse kinematics. Therefore, non-model based control strategies which guarantee convergence of the Cartesian tracking errors is desirable. In addition, Cartesian controllers should be robust and efficient with very low computational cost.
To differentiate this work from other approaches for tracking tasks [Chea et.al. (2006)], [Chea et.al. (2006)], [Moosavian & Papadopoulus (2007)], [Zhao et.al. (2007)] in this chapter it is assumed that the initial condition and desired trajectories belong to the Cartesian workspace Ω, which defines the hyperspace free of singular configurations, an standard assumption for joint robot control. However, this assumption is not evident for others Cartesian controllers [Huang et.al. (2002)], [Chea et.al. (2001)]. This assumption allows us to use a well posed inverse Jacobian for any initial condition. In addition, it is possible to prove that exponential

stability is guaranteed despite the fact the Jacobian is not exactly known and the Jacobian adaptive law is avoided.

Brief introduction to sliding mode control

The name *variable structure control* (sliding mode control) comes from the fact that the control signal is provided by one of two controllers. Which one? It depends on the sign of a scalar switching function **S** that in turn depends on the states of the system. If the outcome of this function is positive, one controller is used. If not, the other one. It is clear that the selection of the switching function is crucial for the control and that it allows to the designer to generate a rich family of behaviors.

If this switching function is designed such that the state velocity vectors in the vicinity of the switching surface (the geometric locus of the states that comply with $\mathbf{S} = 0$) points to the surface, then it is said that a *sliding surface* exists. Why this name? Because once the system intercepts such a surface it continues sliding within it until an equilibrium point is reached.

Therefore, sliding mode control needs to comply with two conditions

- The control law has to provide with sufficient conditions to guarantee the existence and the reachability of the sliding surface.
- Once the state space behavior of the system is restricted to the sliding surface, the dynamics corresponds to the desired one, i.e. stability or tracking.

The properties of sliding mode control ensure that a properly controlled system will reach the sliding surface in a finite time $t_h < \infty$, beyond which the states of the system are ketp within the sliding surface and displaying the desired dynamics.

All the considerations given above rest on assuming ideal sliding modes. This implies having the capability of producing infinitely fast switchings, something of course impossible in the physical world. Therefore, the states of the system oscillate within a neighborhood of the sliding surface. This effect translates into a *chattering* signal [Utkin (1977)], [DeCarlo et.al. (1988)], [Hung et.al. (1993)] that looks like noise.

Contribution

In this chapter, free-chattering second order sliding mode control is presented in order to guarantee convergence of the tracking errors of the robot manipulator under parametric uncertainty. Specifically, a Cartesian second order sliding mode surface is proposed, which drives the sliding PID input. Therefore, the closed loop system renders a sliding mode for all time, whose solution converges to the sliding surface in finite time and a perfect tracking is guaranteed under assumption that the Jacobian is uncertain.

The main characteristics of the proposed scheme can be summarized as follows:

- The regressor is not required.
- Very fast tracking is guaranteed.
- The controller is smooth.
- An exact Jacobian is not required.
- A conservative tuning of feedback gains is required.

The chapter is organized as follows: Section II presents the dynamical model of a rigid n-link serial non-redundant robot manipulator and some useful properties. Section III presents a parameterization of the system in terms of the Cartesian coordinates. Furthermore, two

Cartesian controllers are presented assuming parametric uncertainty. In the first case, a traditional Cartesian controller based on the inverse Jacobian is presented. Now, assuming that the Jacobian is uncertain a Cartesian controller is proposed as a second case. In Section IV, numerical simulations using the proposed approaches are provided. Finally, some conclusions are presented in section V.

2. Dynamical equations of robot manipulator

The dynamical model of a non-redundant rigid serial n-link robot manipulator with all revolute joints is described as follows

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \left(\frac{1}{2}\dot{\mathbf{H}}(\mathbf{q}) + \mathbf{S}(\mathbf{q},\dot{\mathbf{q}})\right)\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{u} \tag{1}$$

where $\mathbf{q}, \dot{\mathbf{q}} \in \Re^n$ are the joint position and velocity vectors, $\mathbf{H}(\mathbf{q}) \in \Re^{nxn}$ denotes a symmetric positive definite inertial matrix, the second term in the left side represent the Coriolis and centripetal forces, $\mathbf{g}(\mathbf{q}) \in \Re^n$ models the gravitational forces, and $\mathbf{u} \in \Re^n$ stands for the torque input.
Some important properties of robot dynamics that will be used in this chapter are:

Property 1. *Matrix $\boldsymbol{H}(\boldsymbol{q})$ is symmetric and positive definite, and both $\boldsymbol{H}(\boldsymbol{q})$ and $\boldsymbol{H}^{-1}(\boldsymbol{q})$ are uniformly bounded as a function of $\boldsymbol{q} \in \Re^n$ [Arimoto (1996)].*

Property 2. *Matrix $\boldsymbol{S}(\boldsymbol{q},\dot{\boldsymbol{q}})$ is skew symmetric and hence satisface [Arimoto (1996)]:*

$$\dot{\boldsymbol{q}}^T\boldsymbol{S}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} = 0 \qquad \forall \boldsymbol{q},\dot{\boldsymbol{q}} \in \Re^n$$

Property 3. *The left-hand side of (1) can be parameterized linearly [Slotine & Li (1987)], that is, a linear combination in terms of suitable selected set of robot and load parameters, i.e.*

$$Y\Theta = \boldsymbol{H}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \left(\frac{1}{2}\dot{\boldsymbol{H}}(\boldsymbol{q}) + \boldsymbol{S}(\boldsymbol{q},\dot{\boldsymbol{q}})\right)\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q})$$

where $\boldsymbol{Y} = \boldsymbol{Y}(\boldsymbol{q},\dot{\boldsymbol{q}},\dot{\boldsymbol{q}},\ddot{\boldsymbol{q}}) \in \Re^{nxp}$ is known as the regressor and $\Theta \in \Re^p$ is a vector constant parameters of the robot manipulator.

2.1 Open loop error equation

In order to obtain a useful representation of the dynamical equation of the robot manipulator for control proposes, equation (1) is represented in terms of the nominal reference $(\dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r) \in \Re^{2n}$ as follows, [Lewis (1994)]:

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}}_r + \left(\frac{1}{2}\dot{\mathbf{H}}(\mathbf{q}) + \mathbf{S}(\mathbf{q},\dot{\mathbf{q}})\right)\dot{\mathbf{q}}_r + \mathbf{g}(\mathbf{q}) = \mathbf{Y}_r\Theta_r \tag{2}$$

where the regressor $\mathbf{Y}_r = \mathbf{Y}_r(\mathbf{q},\dot{\mathbf{q}},\dot{\mathbf{q}}_r,\ddot{\mathbf{q}}_r) \in \Re^{nxp}$ and $\Theta_r \in \Re^p$.
If we add and subtract equation (2) into (1) we obtain the open loop error equation

$$\mathbf{H}(\mathbf{q})\dot{\mathbf{S}}_r + \left(\frac{1}{2}\dot{\mathbf{H}}(\mathbf{q}) + \mathbf{S}(\mathbf{q},\dot{\mathbf{q}})\right)\mathbf{S}_r = \mathbf{u} - \mathbf{Y}_r\Theta \tag{3}$$

where the joint error manifold $\mathbf{S}_r$ is defined as

$$\mathbf{S}_r = \dot{\mathbf{q}} - \dot{\mathbf{q}}_r \tag{4}$$

The robot dynamical equation (3) is very useful to design controllers for several control techniques which are based on errors with respect to the nominal reference [Brogliato et.al. (1991)], [Ge & Hang (1998)], [Liu et.al. (2006)].
Specially, we are interesting in to design controllers for tracking tasks without resorting on $\mathbf{H}(\mathbf{q}), \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}}), \mathbf{g}(\mathbf{q})$. Also, to avoid the ill-posed inverse kinematics in the robot manipulator, a desired Cartesian coordinate system will be used rather than desired joint coordinates $(\mathbf{q}_d^T, \dot{\mathbf{q}}_d^T)^T \in \Re^{3n}$.
In the next section we design a convenient open loop error dynamics system based on Cartesian errors.

3. Cartesian controllers

3.1 Cartesian error manifolds

Let the forward kinematics be a mapping between joint space and task space (in this case Cartesian coordinates) given by [1]

$$\mathbf{X} = \mathbf{f}(\mathbf{q}) \tag{5}$$

where $\mathbf{X}$ is the end-effector position vector with respect to a fixed reference inertial frame, and $\mathbf{f}(\mathbf{q}) : \Re^n \rightarrow \Re^m$ is generally non-linear transformation. Taking the time derivative of the equation (5), it is possible to define a differential kinematics which establishes a mapping at level velocity between joint space and task space, that is

$$\dot{\mathbf{q}} = \mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{X}} \tag{6}$$

where $\mathbf{J}^{-1}(\mathbf{q})$ stands for the inverse Jacobian of $\mathbf{J}(\mathbf{q}) \in \Re^{n \times n}$.
Given that the joint error manifold $\mathbf{S}_r$ is defined at level velocities, equation (6) can be used to defined the nominal reference as

$$\dot{\mathbf{q}}_r = \mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{X}}_r \tag{7}$$

where $\dot{\mathbf{X}}_r$ represents the Cartesian nominal reference which will be designed by the user. Thus, a system parameterization in terms of Cartesian coordinates can be obtained by the equation (7). However an exact knowledge on the inverse Jacobian is required.
Substituting equations (6) and (7) in (4), the joint error manifold $\mathbf{S}_r$ becomes

$$\begin{aligned} \mathbf{S}_r &= \mathbf{J}^{-1}(\mathbf{q})(\dot{\mathbf{X}} - \dot{\mathbf{X}}_r) \\ &\triangleq \mathbf{J}^{-1}(\mathbf{q})\mathbf{S}_x \end{aligned} \tag{8}$$

where $\mathbf{S}_x$ is called as Cartesian error manifold. That is, the joint error manifold is driven by Cartesian errors through Cartesian error manifold.
Now two Cartesian controllers are presented, in order to solve the parametric uncertainty.
Case No.1
Given that the parameters of robot manipulator are changing constantly when it executes a task, or that they are sometimes unknown, then a robust adaptive Cartesian controller can be designed to compensate the uncertainty as follows [Slotine & Li (1987)]

$$\mathbf{u} = -\mathbf{K}_{d1}\mathbf{S}_{r1} + \mathbf{Y}_r\hat{\Theta} \tag{9}$$

$$\dot{\hat{\Theta}} = -\Gamma \mathbf{Y}_r^T \mathbf{S}_{r1} \tag{10}$$

[1] In this paper we consider that the robot manipulator is non-redundant, thus $m = n$.

where $\mathbf{K}_{d1} = \mathbf{K}_{d1}^T > 0 \in \Re^{n \times n}$, $\Gamma = \Gamma^T > 0 \in \Re^{p \times p}$.
Substituting equation (9) into (3), we obtain the following closed loop error equation

$$\mathbf{H}(\mathbf{q})\dot{\mathbf{S}}_{r1} + \left(\frac{1}{2}\dot{\mathbf{H}}(\mathbf{q}) + \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}})\right)\mathbf{S}_{r1} = -\mathbf{K}_{d1}\mathbf{S}_{r1} + \mathbf{Y}_r \Delta\Theta$$

where $\Delta\Theta = \hat{\Theta} - \Theta$. If the nominal reference is defined as $\dot{\mathbf{X}}_{r1} = \mathbf{x}_d - \alpha_1 \Delta \mathbf{x}_1$ where α_1 is a positive-definite diagonal matrix, $\Delta \mathbf{x}_1 = \mathbf{x}_1 - \mathbf{x}_d$ and subscript d denotes desired trajectories, the following result can be obtained.

Assumption 1. *The desired Cartesian references $\mathbf{x}_d$ are assumed to be bounded and uniformly continuous, and its derivatives up to second order are bounded and uniformly continuous.*

Theorem 1. *[Asymptotic Stability] Assuming that the initial conditions and the desired trajectories are defined in a singularities-free space. The closed loop error dynamics used in equations (9), (10) guarantees that $\Delta \mathbf{x}_1$ and $\Delta \dot{\mathbf{x}}_1$ tends to zero asymptotically.*

Proof. Consider the Lyapunov function

$$V = \frac{1}{2}\mathbf{S}_{r1}^T \mathbf{H}(\mathbf{q})\mathbf{S}_{r1} + \frac{1}{2}\Delta\Theta^T \Gamma^{-1} \Delta\Theta$$

Differentiating V with respect to time, we get

$$\dot{V} = -\mathbf{S}_{r1}\mathbf{K}_{d1}\mathbf{S}_{r1} \leq 0$$

Since $\dot{V} \leq 0$, we can state that V is also bounded. Therefore, $\mathbf{S}_{r1}$ and $\Delta\Theta$ are bounded. This implies that $\hat{\Theta}$ and $\mathbf{J}^{-1}(\mathbf{q})\mathbf{S}_{x1}$ are bounded if $\mathbf{J}^{-1}(\mathbf{q})$ is well posed for all t. From the definition of $\mathbf{S}_{x1}$ we have that $\Delta \dot{\mathbf{x}}_1$, and $\Delta \mathbf{x}_1$ are also bounded. Since $\Delta \dot{\mathbf{x}}_1$, $\Delta \mathbf{x}_1$, $\Delta\Theta$, and $\mathbf{S}_{r1}$ are bounded, we have that $\dot{\mathbf{S}}_{r1}$ is bounded. This shows that $\ddot{V}$ is bounded. Hence, $\dot{V}$ is uniformly continuous. Using the Barbalat's lemma [Slotine & Li (1987)], we have that $\dot{V} \to 0$ at $t \to \infty$. This implies that $\Delta \mathbf{x}_1$ and $\Delta \dot{\mathbf{x}}_1$ tend to zero as t tends to infinity. Then, tracking errors $\Delta \mathbf{x}_1$ and $\Delta \dot{\mathbf{x}}_1$ are asymptotically stable [Lewis (1994)]. □

The properties of this controller can be numbered as:

a) On-line computing regressor and the exact knowledge of $\mathbf{J}^{-1}(\mathbf{q})$ are required.
b) Asymptotic stability is guaranteed assuming that $\mathbf{J}^{-1}(\mathbf{q})$ is well posed for all time. Therefore, the stability domain is very small because $\mathbf{q}(t)$ may exhibit a transient response such that $\mathbf{J}(\mathbf{q})$ losses rank.

In order to avoid the dependence on the inverse Jacobian, in the next case it is assumed that the Jacobian is uncertain. At the same time, the drawbacks presented in the Case No.1 are solved.
Case No.2 Considering that the Jacobian is uncertain, i.e. the Jacobian is not exactly known, the nominal reference proposed in equation (7) is now defined as

$$\hat{\dot{\mathbf{q}}}_r = \hat{\mathbf{J}}^{-1}(\mathbf{q})\dot{\mathbf{X}}_{r2} \quad (11)$$

where $\hat{\mathbf{J}}^{-1}(\mathbf{q})$ stands as an estimates of $\mathbf{J}^{-1}(\mathbf{q})$ such that $rank(\hat{\mathbf{J}}^{-1}(\mathbf{q})) = n$ for all t and for all $\mathbf{q} \in \Omega$ where $\Omega = \{\mathbf{q}|rank(\mathbf{J}(\mathbf{q})) = n\}$. Therefore, a new joint error manifold arises coined as uncertain Cartesian error manifold is defined as follows

$$\begin{aligned}\hat{\mathbf{S}}_{r2} &= \dot{\mathbf{q}} - \hat{\dot{\mathbf{q}}}_r \\ &= \mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{X}} - \hat{\mathbf{J}}^{-1}(\mathbf{q})\dot{\mathbf{X}}_{r2}\end{aligned} \tag{12}$$

In order to guarantee that the Cartesian trajectories remain on the manifold $\mathbf{S}_x$ although the Jacobian is uncertain, a second order sliding mode is proposed by means of tailoring $\dot{\mathbf{X}}_{r2}$. That is, a switching surface over the Cartesian manifold $\mathbf{S}_x$ should be invariant to changes in $\mathbf{J}^{-1}(\mathbf{q})$. Hence, high feedback gains can to ensure the boundedness of all closed loop signals and the exponential convergence is guaranteed despite Jacobian uncertainty.
Let the new nominal reference $\dot{\mathbf{X}}_{r2}$ be defined as

$$\begin{aligned}\dot{\mathbf{X}}_{r2} =& \dot{\mathbf{x}}_d - \alpha_2 \Delta \mathbf{x}_2 + \mathbf{S}_d - \gamma_p \sigma \\ \dot{\sigma} =& sgn(\mathbf{S}_e)\end{aligned} \tag{13}$$

where α_2 is a positive-definite diagonal matrix, $\Delta \mathbf{x}_2 = \mathbf{x}_2 - \mathbf{x}_d$, $\mathbf{x}_d$ is a desired Cartesian trajectory, γ_p is positive-definite diagonal matrix and function $sgn(*)$ stands for the signum function of $(*)$ and

$$\begin{aligned}\mathbf{S}_e &= \mathbf{S}_x - \mathbf{S}_d \\ \mathbf{S}_x &= \Delta \dot{\mathbf{x}}_2 + \alpha_2 \Delta \mathbf{x}_2 \\ \mathbf{S}_d &= \mathbf{S}_x(t_0) exp^{-\kappa(t-t_0)}, \quad \kappa > 0\end{aligned}$$

Now, substituting equation (13) in (12) we have that

$$\hat{\mathbf{S}}_{r2} = \mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{X}} - \hat{\mathbf{J}}^{-1}(\mathbf{q})(\dot{\mathbf{x}}_d - \alpha_2 \Delta \mathbf{x}_2 + \mathbf{S}_d - \gamma_p \int_{t_0}^{t} sgn(\mathbf{S}_e(\tau))d\tau) \tag{14}$$

Uncertain Open Loop Equation
Using equation (11), the uncertain parameterization of $Y_r \Theta_r$ becomes

$$\mathbf{H}(\mathbf{q})\hat{\ddot{\mathbf{q}}}_r + \left(\frac{1}{2}\dot{\mathbf{H}}(\mathbf{q}) + \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}})\right) \hat{\dot{\mathbf{q}}}_r + \mathbf{g}(\mathbf{q}) = \hat{Y}_r \Theta_r \tag{15}$$

If we add and subtract equation (15) to (1), the uncertain open loop error equation is defined as

$$\mathbf{H}(\mathbf{q})\dot{\hat{\mathbf{S}}}_{r2} + \left(\frac{1}{2}\dot{\mathbf{H}}(\mathbf{q}) + \mathbf{S}(\mathbf{q}, \dot{\mathbf{q}})\right) \hat{\mathbf{S}}_{r2} = \mathbf{u} - \hat{Y}_r \Theta_r \tag{16}$$

Theorem 2: [Local Stability] Assuming that the initial conditions and the desired trajectories are within a space free of singularities. Consider the uncertain open loop error equation (16) in closed loop with the controller given by

$$\mathbf{u} = -\mathbf{K}_{d2}\hat{\mathbf{S}}_{r2} \tag{17}$$

with $\mathbf{K}_{d2}$ an $n \times n$ diagonal symmetric positive-definite matrix. Then, for large enough gain $\mathbf{K}_{d2}$ and small enough error in initial conditions, local exponential tracking is assured provided that $\gamma_p \geq \|\dot{\mathbf{J}}(\mathbf{q})\hat{\mathbf{S}}_{r2} + \mathbf{J}(\mathbf{q})\dot{\hat{\mathbf{S}}}_{r2} + \dot{\mathbf{J}}(\mathbf{q})\Delta J \dot{\mathbf{X}}_{r2} + \mathbf{J}(\mathbf{q})\Delta \dot{J} \dot{\mathbf{X}}_{r2} + \mathbf{J}(\mathbf{q})\Delta J \ddot{\mathbf{X}}_{r2}\|$.

Proof. Substituting equation (17) into (16) we obtain the closed-loop dynamics given as

$$\mathbf{H}(\mathbf{q})\dot{\hat{\mathbf{S}}}_{r2} = -\left(\frac{1}{2}\dot{\mathbf{H}}(\mathbf{q}) + \mathbf{S}(\mathbf{q},\dot{\mathbf{q}})\right)\hat{\mathbf{S}}_{r2} - \mathbf{K}_{d2}\hat{\mathbf{S}}_{r2} - \hat{Y}_r\Theta \tag{18}$$

The proof is organized in three parts as follows.
Part 1: Boundedness of Closed-loop Trajectories. Consider the following Lyapunov function

$$V = \frac{1}{2}\hat{\mathbf{S}}_{r2}^T\mathbf{H}(\mathbf{q})\hat{\mathbf{S}}_{r2} \tag{19}$$

whose total derivative of (19) along its solution (18) leads to

$$\dot{V} = -\hat{\mathbf{S}}_{r2}^T\mathbf{K}_{d2}\hat{\mathbf{S}}_{r2} - \hat{\mathbf{S}}_{r2}^T\hat{\mathbf{Y}}_r\Theta \tag{20}$$

Similarly to [Parra & Hirzinger (2000)], we have that $\hat{\mathbf{Y}}_r\Theta \leq \eta(t)$ with η a functional that bounds $\hat{\mathbf{Y}}_r$. Then, equation (20) becomes

$$\dot{V} \leq -\hat{\mathbf{S}}_{r2}^T\mathbf{K}_{d2}\hat{\mathbf{S}}_{r2} - \|\hat{\mathbf{S}}_{r2}\|\eta(t) \tag{21}$$

For initial errors that belong to a neighborhood ϵ_1 with radius $r > 0$ near the equilibrium $\hat{\mathbf{S}}_{r2} = 0$, we have that thanks to Lyapunov arguments, there is a large enough feedback gain $\mathbf{K}_{d2}$ such that $\hat{\mathbf{S}}_{r2}$ converges into a set-bounded ϵ_1. Thus, the boundedness of tracking errors can be concluded, namely

$$\hat{\mathbf{S}}_{r2} \to \epsilon_1 \qquad \text{as} \qquad t \to \infty \tag{22}$$

then

$$\hat{\mathbf{S}}_{r2} \in \mathcal{L}_\infty \Rightarrow \|\hat{\mathbf{S}}_{r2}\| < \epsilon_1 \tag{23}$$

where $\epsilon_1 > 0$ is a upper bounded.
Since desired trajectories are C^2 and feedback gains are bounded, we have that $(\dot{\mathbf{q}}_r, \ddot{\mathbf{q}}_r) \in \mathcal{L}_\infty$, which implies that $\dot{\mathbf{X}}_{r2} \in \mathcal{L}_\infty$ if $\hat{\mathbf{J}}^{-1}(\mathbf{q}) \in \mathcal{L}_\infty$. Then, the right hand side of (18) is bounded given that the Coriolis matrix and gravitational vector are also bounded. Since $\mathbf{H}(\mathbf{q})$ and $\mathbf{H}^{-1}(\mathbf{q})$ are uniformly bounded, it is seen from (18) that $\dot{\hat{\mathbf{S}}}_{r2} \in \mathcal{L}_\infty$. Hence there exists a bounded scalar $\epsilon_2 > 0$ such that

$$\|\dot{\hat{\mathbf{S}}}_{r2}\| < \epsilon_2 \tag{24}$$

So far, we conclude the boundedness of all closed-loop error signals.
Part 2. Sliding Mode. If we add and subtract $\mathbf{J}^{-1}(\mathbf{q})\dot{X}_r$ to (12), we obtain

$$\begin{aligned}\hat{\mathbf{S}}_{r2} &= \mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{X}} - \hat{\mathbf{J}}^{-1}(\mathbf{q})\dot{\mathbf{X}}_{r2} \pm \mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{X}}_{r2} \\ &= \mathbf{J}^{-1}(\mathbf{q})(\dot{\mathbf{X}} - \dot{\mathbf{X}}_{r2}) + (\mathbf{J}^{-1}(\mathbf{q}) - \hat{\mathbf{J}}^{-1}(\mathbf{q}))\dot{\mathbf{X}}_{r2} \\ &= \mathbf{J}^{-1}(\mathbf{q})\mathbf{S}_x - \Delta J\dot{\mathbf{X}}_{r2}\end{aligned} \tag{25}$$

which implies that $\Delta J = \mathbf{J}^{-1}(\mathbf{q}) - \hat{\mathbf{J}}^{-1}(\mathbf{q})$ is also bounded. Now, we will show that a sliding mode at $\mathbf{S}_e = 0$ arises for all time as follows.
If we premultiply (25) by $\mathbf{J}(\mathbf{q})$ and rearrange the terms, we obtain

$$\mathbf{S}_x = \mathbf{J}(\mathbf{q})\hat{\mathbf{S}}_{r2} + \mathbf{J}(\mathbf{q})\Delta J\dot{\mathbf{X}}_{r2} \tag{26}$$

Since $\mathbf{S}_x = \mathbf{S}_e + \gamma_p \int_{t_0}^{t} sgn(\mathbf{S}_e(\zeta))d\zeta$, we have that

$$\mathbf{S}_e = -\gamma_p \int_{t_0}^{t} sgn(\mathbf{S}_e(\zeta))d\zeta + \mathbf{J}(\mathbf{q})(\hat{\mathbf{S}}_{r2} + \Delta J\dot{\mathbf{X}}_{r2}) \tag{27}$$

Deriving (27), and then premultiplying by $\mathbf{S}_e^T$, we obtain

$$\begin{aligned} \mathbf{S}_e^T\dot{\mathbf{S}}_e &= -\gamma_p|\mathbf{S}_e| + \mathbf{S}_e^T \frac{d}{dt}\left(\mathbf{J}(\mathbf{q})\hat{\mathbf{S}}_{r2} + \mathbf{J}(\mathbf{q})\Delta J\dot{\mathbf{X}}_{r2})\right) \\ &\leq -\gamma_p|\mathbf{S}_e| + \zeta|\mathbf{S}_e| \\ &\leq -(\gamma_p - \zeta)|\mathbf{S}_e| \\ &= -\mu|\mathbf{S}_e| \end{aligned} \tag{28}$$

where $\mu = \gamma_p - \zeta$ and $\zeta = \dot{J}(q)\hat{\mathbf{S}}_{r2} + \mathbf{J}(\mathbf{q})\dot{\hat{\mathbf{S}}}_{r2} + \dot{J}(q)\Delta J\dot{\mathbf{X}}_{r2} + \mathbf{J}(\mathbf{q})\Delta\dot{J}\dot{\mathbf{X}}_{r2} + \mathbf{J}(\mathbf{q})\Delta J\ddot{\mathbf{X}}_{r2}$. Therefore, we obtain the sliding mode condition if

$$\gamma_p > \zeta \tag{29}$$

in such a way that $\mu > 0$ guarantees the existence of a sliding mode at $\mathbf{S}_e = 0$ at time $t_e \leq \frac{|\mathbf{S}_e(t_0)|}{\mu}$. However, notice that for any initial condition $\mathbf{S}_e(t_0) = 0$, and hence $t \equiv 0$ implies that a sliding mode in $\mathbf{S}_e = 0$ is enforced for all time without reaching phase.
Part 3: Exponential Convergence. Sliding mode at $\mathbf{S}_e = 0$ implies that $\mathbf{S}_x = \mathbf{S}_d$, thus

$$\Delta\dot{\mathbf{x}}_2 = -\alpha_2\Delta\mathbf{x}_2 + \mathbf{S}_x(t_0)e^{-k_p t} \tag{30}$$

which decays exponentially fast toward $[\Delta\mathbf{x}_2, \Delta\dot{\mathbf{x}}_2] \to (0, 0)$, that is

$$\mathbf{x}_2 \to \mathbf{x}_d \quad \text{and} \quad \dot{\mathbf{x}}_2 \to \dot{\mathbf{x}}_d \tag{31}$$

it is locally exponential. □

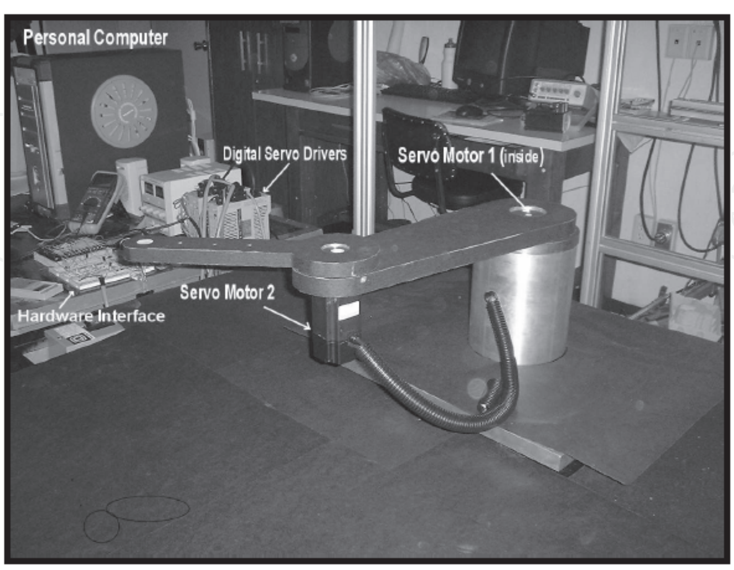
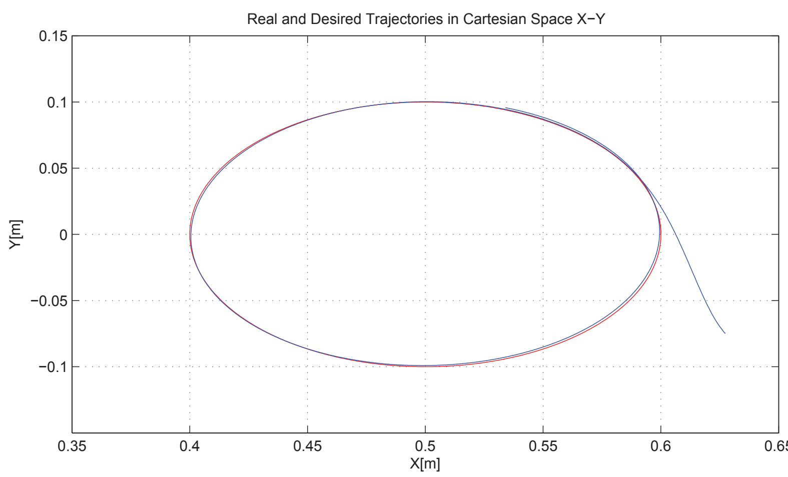
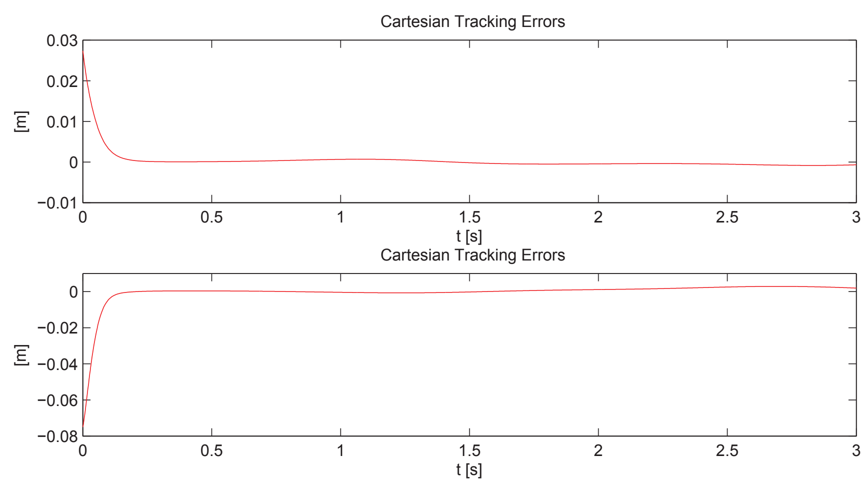


Fig. 1. Planar Manipulator of 2-DOF.

The properties of this controller can be numbered as



(a) Theorem 1: Plane Phase



(b) Theorem 1: Cartesian Tracking Errors

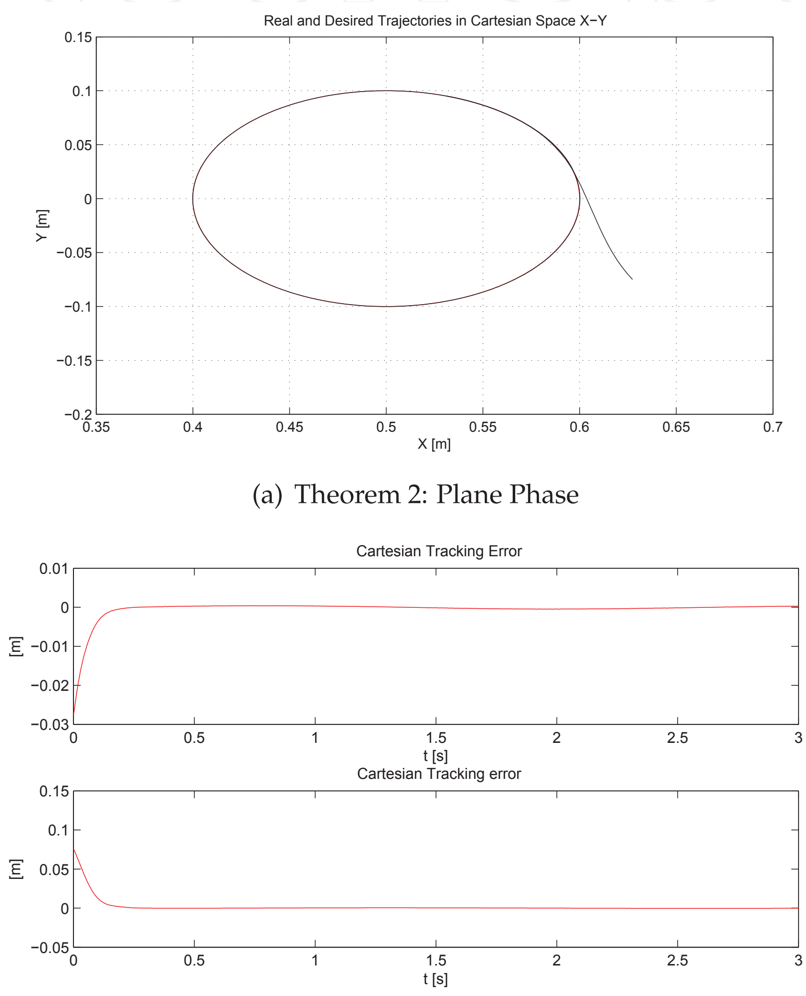
Fig. 2. Cartesian Tracking of the Robot Manipulator using Theorem 1.

a) The sliding mode discontinuity associated to $\hat{\mathbf{S}}_{r2} = 0$ is relegated to the first order time derivative of $\dot{\hat{\mathbf{S}}}_{r2}$. Then, sliding mode condition in the closed loop system is induced by the $sgn(\mathbf{S}_e)$ and an exponential convergence of the tracking error is established. Therefore, the closed loop is robust due to the invariance achieved by the sliding mode, robustness against unmodeled dynamics, and parametric uncertainty. A difference of this approach from others [Lee & Choi (2004)], [Barambones & Etxebarria (2002)], [Jager (1996)], [Stepanenko et.al. (1998)], is that the closed loop dynamics does not exhibit chattering. Finally, notice that the discontinuous function $sgn(\mathbf{S}_e)$ is only used in the stability analysis.

c) The control synthesis does not depend on any knowledge of the robot dynamics: it is model free. In addition, a smooth control input is guaranteed.

d) Taking $\gamma_p = 0$ in equation (13), it is obtained the joint error manifold $\mathbf{S}_{r1}$ defined in the Case No.1, which is commonly used in several approaches. However under this sliding surface it is not possible to prove convergence in finite time as well as reaching the sliding condition. Then, a dynamic change of coordinates is proposed, where for a large enough feedback gain K_d in the control law, the passivity between η_1 and $\hat{\mathbf{S}}_{r2}$ is preserved with $\eta_1 = \dot{\hat{\mathbf{S}}}_{r2}$ [Parra & Hirzinger (2000)]. In addition, for large enough γ_p the dissipativity is established between $\mathbf{S}_e$ and η_2 with $\eta_2 = \dot{\mathbf{S}}_e$.

e) In order to differentiate from other approaches where the parametric uncertainty in the Jacobian matrix is expressed as a linear combination of a selected set of kinematic parameters [Chea et.al. (1999)], [Chea et.al. (2001)], [Huang et.al. (2002)], [Chea et.al. (2004)], [Chea et.al. (2006)], [Chea et.al. (2006)], in this chapter the Jacobian uncertainty is parameterized in terms of a regressor times as parameter vector. To get the parametric uncertainty, this vector is multiplied by a factor with respect to the nominal value.



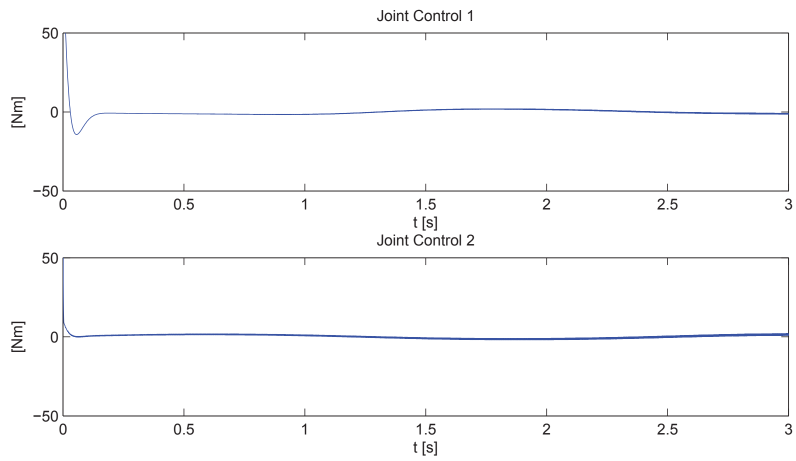
(a) Theorem 2: Plane Phase

(b) Theorem 2: Cartesian Tracking Errors

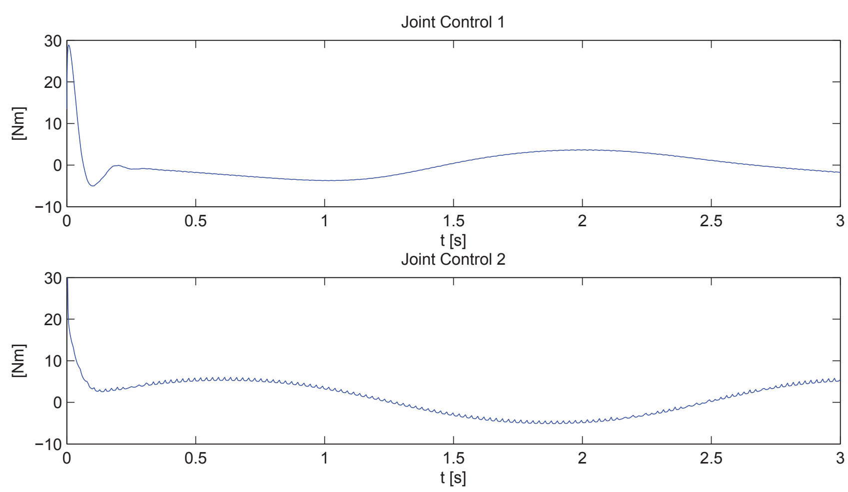
Fig. 3. Cartesian Tracking of the Robot Manipulator using Theorem 2.

4. Simulation results

In this section we present simulation results carried out on 2 degree of freedom (DOF) planar robot arm, Fig. 1. The experiments were developed on Matlab 6.5 and each experiment has an average running of 3 [s]. Parameters of the robot manipulator used in these simulations are shown in Table 1.



(a) Theorem 1: Control Inputs



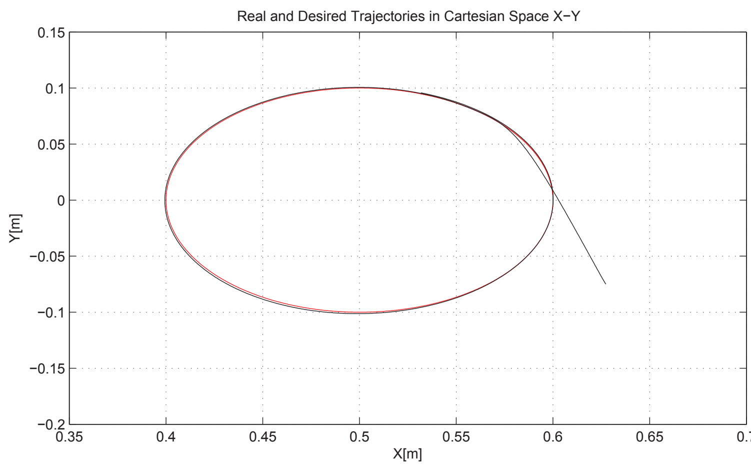
(b) Theorem 2: Control Inputs

Fig. 4. Control Inputs applied to Each Joint.

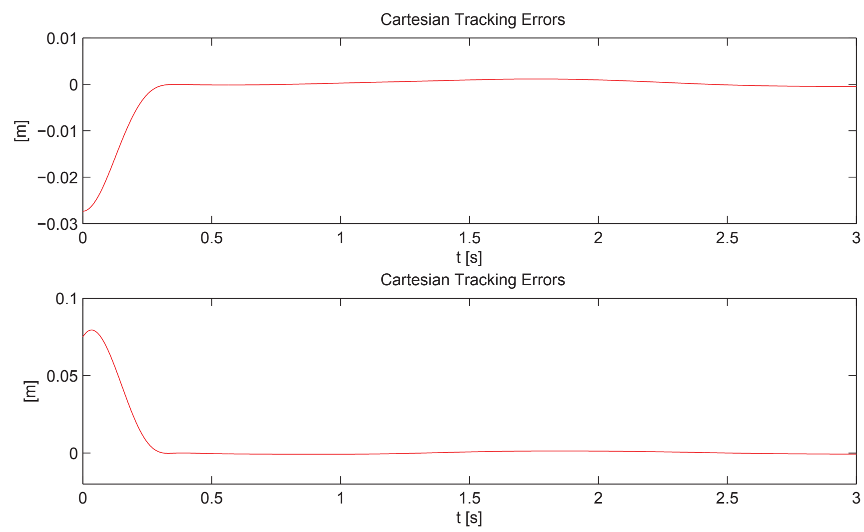
Parameters	m_1	m_2	l_1	l_2
Value	8 Kg	5 Kg	0.5 m	0.35 m
Parameters	l_{c1}	l_{c2}	I_1	I_2
Value	0.19 m	0.12 m	0.02 Kgm^2	0.16 Kgm^2

Table 1. Robot Manipulator Parameters.

The objective of these experiments is to given a desired trajectory, the end effector must follow it in a finite time. The desired task is defined as a circle of radius 0.1 [m] whose center located at X=(0.55,0) [m] in the Cartesian workspace. The initial condition is defined as $[q_1(0) = -0.5, q_2(0) = 0.9]^T$ [rad]. which is used for all experiments. In addition, we consider zero initial velocity and 95% of parametric uncertainty.
The performance of the robot manipulator using equations (9) and (10) defined in theorem 1 are presented in Fig. 2. In this case, the end-effector tracks the desired Cartesian trajectory once the Cartesian error manifold is reached, Fig. 2(a). In addition, as it is showed in Fig. 2(b),



(a) TBG: Plane Phase



(b) TBG: Cartesian Tracking Errors

Fig. 5. Cartesian Tracking of the Robot Manipulator using TBG

the Cartesian tracking errors converge asymptotically to zero in few seconds. However, for practical applications it is necessary to know exactly the regressor and the inverse Jacobian.
Now, assuming that the Jacobian is uncertain, there is no knowledge of the regressor, and there cannot be any overparametrization, then a Cartesian tracking of the robot manipulator using control law defined in equation (17) is presented in Fig 3(a). As it is expected, after a very short time, approximately 2 [s], the end effector of the robot manipulator follows the desired trajectory, Fig. 3(a) and Fig. 3(b). This is possible because in the proposed scheme all the time it is induced a sliding mode. Thus, it is more faster and robust.
On the other hand, in Fig. 4 are shown the applied input torques for each joint of the robot manipulator for the cases 1 and 2. It can be see that control inputs using the controller defined in equation (17) are more smooth and chattering free than controller defined in equation (9).
Given that in several applications, such as manipulation tasks or bipedal robots, it is not enough the convergence of the errors when t tends to infinity. Finite time convergence faster that exponential convergence has been proposed [Parra & Hirzinger (2000)]. To speed up the

response, a time base generator (TBG) that shapes a feedback gain α_2 is used. That is, it is necessary to modify the feedback gain α_2 defined in equation (13) by

$$\alpha_2(t) = \alpha_0 \frac{\dot{\xi}}{1 - \xi + \delta} \tag{32}$$

where $\alpha_0 = 1 + \epsilon$, for small positive scalar ϵ such that α_0 is close to 1 and $0 < \delta \ll 1$. The time base generator $\xi = \xi(t) \in \mathcal{C}^2$ must be provided by the user so as to get ξ to go smoothly from 0 to 1 in finite time $t = t_b$, and $\dot{\xi} = \dot{\xi}(t)$ is a bell shaped derivative of ξ such that $\dot{\xi}(t_0) = \dot{\xi}(t_b) \equiv 0$ [Parra & Hirzinger (2000)]. Accordingly, given that the convergence speed of the tracking errors is increased by the TBG, a finite time convergence of the tracking errors is guaranteed.

In the Fig. 5 are shown simulation results using a finite time convergence at $t_b = 0.4$ [s]. As it is expected, the end effector follows exactly the desired trajectory at $t_b \geq 0.4$ [s], as shown in Fig. 5(a). At the same time, Cartesian tracking errors converge to zero in the desired time, Fig. 5(b).

The feedback gains used in these experiments are given in Table 2 where the subscript ji represents the joint of the robot manipulator with $i = 1, 2$.

K_{dj1}	K_{dj2}	α_{j1}	α_{j2}	γ_{pj1}	γ_{pj2}	k_p	Γ	t_b	Case
60	60	25	25	-	-	-	0.01	-	1
50	20	30	30	0.01	0.01	20	-	-	2
60	60	2.2	2.2	0.01	0.01	20	-	0.4s	TBG

Table 2. Feedback Gains

5. Conclusion

In this chapter, two Cartesian controllers under parametric uncertainties are presented. In particular, an alternative solution to the Cartesian tracking control of the robot manipulator assuming parametric uncertainties is presented. To do this, second order sliding surface is used in order to avoid the high frequency commutation. In addition, closed loop renders a sliding mode for all time to ensure convergence without any knowledge of robot dynamics and Jacobian uncertainty. Simulation results allow to visualize the predicted stability properties on a simple but representative task.

6. References

Arimoto, S. (1996). *Control Theory of Non-linear Mechanical Systems*, Oxford University Press.

Barambones, O. & Etxebarria, V. (2002). *Robust Neural Network for Robotic Manipulators*, Automtica, Vol. 38, pp. 235-242.

Brogliato, B., Landau, I-D. & Lozano-Leal, R. (1991).*Adaptive Motion Control of Robot Manipulators: A unified Approach based on Passivity*, International Journal of Robust and Nonlinear Control, Vol. 1, No. 3, pp. 187-202.

Cheah, C.C., Kawamura, S. & Arimoto, S. (1999). *Feedback Control for Robotics Manipulator with Uncertain Jacobian Matrix*, Journal of Robotics Systems, Vol. 16, No. 2, pp. 120-134.

Cheah, C.C., Kawamura, S., Arimoto, S. & Lee, K. (2001). *A Tuning for Task-Space Feedback Control of Robot with Uncertain Jacobian Matrix*, IEEE Trans. on Automat. Control, Vol. 46, No. 8, pp. 1313-1318.

Cheah, C.C., Hirano, M., Kawamura, S. & Arimoto, S. (2004). *Approximate Jacobian Control With Task-Space Damping for Robot Manipulators*, IEEE Trans. on Autom. Control, Vol. 19, No. 5, pp. 752-757.

Cheah, C.C., Liu, C. & Slotine, J. J. E. (2006). *Adaptive Jacobian Tracking Control of Robots with Uncertainties in Kinematics, Dynamics and Actuator Models*, IEEE Trans. on Autom. Control, Vol. 51, No. 6, pp. 1024-1029.

Cheah, C.C. & Slotine, J. J. E. (2006). *Adaptive Tracking Control for Robots with unknown Kinematics and Dynamics Properties*, International Journal of Robotics Research, Vol. 25, No. 3, pp. 283-296.

DeCarlo, R. A., Zak, S. H. & Matthews, G. P. (1988). *Variable Structure Control of Nonlinear Multivariable Systems: A Tutorial*, Proceedings of the IEEE, Vol. 76, No. 3, pp. 212-232.

Ge, S.S. & Hang, C.C. (1998). *Structural Network Modeling and Control of Rigid Body Robots*, IEEE Trans. on Robotics and Automation, Vol. 14, No. 5, pp. 823-827.

Huang, C.Q., Wang, X.G. & Wang, Z.G. (2002). *A Class of Transpose Jacobian-based NPID Regulators for Robot Manipulators with Uncertain Kinematics*, Journal of Robotic Systems, Vol. 19, No. 11, pp. 527-539.

Hung, J. Y., Gao, W. & Hung, J. C. (1993).*Variable Structure Control: A Survey*, IEEE Transactions on Industrial Electronics, Vol. 40, No. 1, pp. 2-22.

Jager, B., (1996). *Adaptive Robot Control with Second Order Sliding Component*, 13th Triennial World Congress, San Francisco, USA, pp. 271-276.

Lee, M.-J. & Choi, Y.-K. (2004). *An Adaptive Neurocontroller Using RBFN for Robot Manipulators*, IEEE Trans. on Industrial Electronics, Vol. 51 No. 3, pp. 711-717.

Lewis, F.L. & Abdallaah, C.T. (1994). *Control of Robot Manipulators*, New York: Macmillan.

Liu, C., Cheah, C.C. & Slotine, J.J.E. (2006). *Adaptive Jacobian Tracking Control of Rigid-link Electrically Driven Robots Based on Visual Task-Space Information*, Automatica, Vol. 42, No. 9, pp. 1491-1501.

Miyazaki, F. & Masutani, Y. (1990). *Robustness of Sensory Feedback Control Based on Imperfect Jacobian*, Robotic Research: The Fifth International Symposium, pp. 201-208.

Moosavian, S. A. & Papadopoulus, E. (2007). *Modified Transpose Jacobian Control of Robotics Systems*, Automatica, Vol. 43, pp. 1226-1233.

Parra-Vega, V. & Hirzinger, G. (2000). *Finite-time Tracking for Robot Manipulators with Singularity-Free Continuous Control: A Passivity-based Approach*, Proc. IEEE Conference on Decision and Control , 5, pp. 5085-5090.

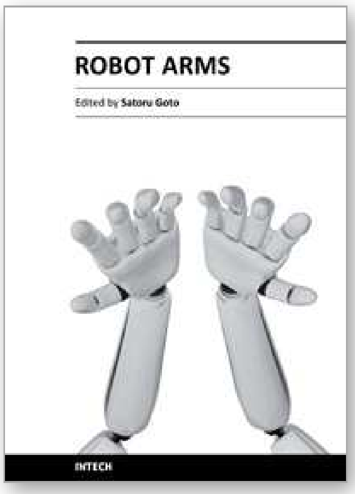
Slotine, J.J.E. & Li, W. (1987). *On the Adaptive Control of Robot Manipulator*, Int. Journal of Robotics Research, Vol. 6, No. 3, pp. 49-59.

Stepanenko, Y., Cao, Y. & Su, A.C. (1998). *Variable Structure Control of Robotic Manipulator with PID Sliding Surfaces*, Int. Journal of Robust and Nonlinear Control, Vol. 8, pp. 79-90.

Utkin, V.J. (1977). *Variable Structure Systems with Sliding Modes*, IEEE Trans. on Autom. Control, Vol. 22, No. 1, pp. 212-222.

Yazarel, H.& Cheah, C.C. (2001). *Task-Space Adaptive Control of Robots Manipulators with Uncertain Gravity Regressor Matrix and kinematics*, IEEE Trans. on Autom. Control, Vol. 47, No. 9, pp. 1580-1585.

Zhao, Y., Cheah, C.C. & Slotine, J.J.E. (2007). *Adaptive Vision and Force Tracking Control of Constrained Robots with Structural Uncertainties*, Proc. IEEE International Conference on Robotics and Automation, pp.2349-2354.



Robot Arms

Edited by Prof. Satoru Goto

ISBN 978-953-307-160-2
Hard cover, 262 pages
Publisher InTech
Published online 09, June, 2011
Published in print edition June, 2011

Robot arms have been developing since 1960's, and those are widely used in industrial factories such as welding, painting, assembly, transportation, etc. Nowadays, the robot arms are indispensable for automation of factories. Moreover, applications of the robot arms are not limited to the industrial factory but expanded to living space or outer space. The robot arm is an integrated technology, and its technological elements are actuators, sensors, mechanism, control and system, etc.