Discrete Time Sliding Mode Control of Robotic Manipulators: Development and Experimental Validation

Valentino Fossi, Andrea Giantomassi, Gianluca Ippoliti, Sauro Longhi, Giuseppe Orlando
Dipartimento di Ingegneria Informatica, Gestionale e dell’Automazione,
Università Politecnica delle Marche, Ancona, Italy.
valentino.fossi@gmail.com
{a.giantomassi, gianluca.ippoliti, sauro.longhi, giuseppe.orlando}@univpm.it.

Maria Letizia Corradini
Dipartimento di Matematica e Informatica,
Università di Camerino, via Madonna delle Carceri,
62032 Camerino (MC), Italy
letizia.corradini@unicam.it

Abstract

This paper presents a discrete-time sliding mode control based on prediction compensation of uncertainties for planar robotic manipulators. Autoregressive models, identified on-line by Kalman Filters, are used to learn about uncertainties affecting the system. The analysis of the control stability is given and the controller is evaluated on the ERICC robot arm. Experiments show that the proposed controller produces good trajectory tracking performance and it is robust in the presence of model inaccuracies.

1 Introduction

Robotic manipulators are highly nonlinear dynamic systems with unmodeled dynamics and uncertainties [17] that, being commonly used in industrial tasks, are expected to maintain good dynamic performance. The design of ideal controllers for such systems is a challenge for control engineers, mainly because of the nonlinearities and the coupling effects typical of robotic systems. Different approaches have been followed in order to cope with this problem, such as, for instance, feedback linearization [1, 14], model predictive control [11, 13], and sliding mode control [18]. In general, control approaches which do not guarantee some robustness can make the performance of the system, in terms of convergence, quite poor. As discussed in [3], global feedback linearization is possible in theory, but is difficult to achieve in practice as a consequence of uncertainties coming from incomplete knowledge of the kinematics and dynamics, from joint and link flexibility, actuator dynamics, friction, sensor noise, and unknown loads. This imposes to couple the inverse dynamics approach with robust control methodologies [1]. It is well known that sliding mode methods provide noticeable robustness and invariance properties to matched uncertainties [20] [22], and is computational simpler with respect to other robust control approaches. Recent literature contains a number of results about sliding mode control of manipulators, in some cases coupled with fuzzy control and or neuro-fuzzy techniques [10, 6]. The largest part of these papers, however, uses the continuous-time dynamic model of the manipulator for design, leaving not addressed the issue of digitalization of the control law. Digital control systems are currently receiving considerable credit as a consequence of the recent advances in digital microprocessor technology, and relevant interest is currently growing in the design of controllers based on the discrete model of the system. Nevertheless, the discrete time counterpart of sliding mode control design has received only a limited attention [4, 5, 8, 15]. Indeed, compared with continuous-time sliding-mode strategies, the design problem in discrete-time has received much less coverage in the literature. This is due to its major drawback, consisting in the presence of a sector, of width depending on the available bound on uncertainties, where robustness is lost because the sliding mode condition cannot be exactly imposed any longer. For this reason, only ultimate boundedness of trajectories can be guaranteed, and the larger are the uncertainties affecting the system, the wider is the bound on trajectories which can
be guaranteed. As a possible solution to this problem, this paper proposes the design of the discontinuous control law, within the sector, based on an estimation, as accurate as possible, of uncertainties affecting the system.

The paper is organized as follows. The robot dynamics is presented in Section 2. In Section 3 details on the considered control are discussed. The uncertain predictor is discussed in Section 4. Results on robot arm tests are reported in Section 5. The paper ends with comments on the performance of the proposed controller.

2 Robot Dynamics

From the Euler-Lagrangian formulation, the equations of motion of a robot manipulator can be written as [19]

\[ B(q)\ddot{q} + C(q, \dot{q})\dot{q} + F_s\sgn(\dot{q}) + G(q) = \tau \]

where \( q \in \mathbb{R}^n \) denotes the vector of generalized coordinates (rotational joint configurations), \( B(q) \in \mathbb{R}^{n \times n} \) is the symmetric positive definite inertia matrix which is bounded for any \( q \), \( C(q, \dot{q})\dot{q} \in \mathbb{R}^n \) represents the centrifugal and Coriolis torques, \( F_s \in \mathbb{R}^{n \times n} \) is the diagonal matrix of the viscous friction coefficients, \( \sgn(\dot{q}) \in \mathbb{R}^n \) is the vector of sign function, \( G(q) \in \mathbb{R}^n \) is the vector of gravitational torques and \( \tau \in \mathbb{R}^n \) is the vector of torques acting at the joints. The robot model (1) is characterized by the structural properties given in [19] and it is assumed that it has uncertainties, i.e.:

\begin{align*}
B(q) &= B_0(q) + \Delta B(q) \\
C(q, \dot{q}) &= C_0(q, \dot{q}) + \Delta C(q, \dot{q}) \\
F_s &= F_{s0} + \Delta F_{s0} \\
G(q) &= G_0(q) + \Delta G(q)
\end{align*}

where \( B_0(q), C_0(q, \dot{q}), F_{s0}, \) and \( G_0(q) \) are the nominal terms; \( \Delta B(q), \Delta C(q, \dot{q}), \Delta F_{s0}, \Delta F_{s0}, \) and \( \Delta G(q) \) are uncertain bounded terms.

A planar two-link manipulator with revolution joints [19] is considered in this paper to illustrate the feasibility of the proposed control algorithm. In particular \( q = [q_2 \ q_3]^T \) where \( q_2, q_3 \) denote the joint displacements of the two considered rotational joints 2 and 3 of Fig. 1 and the dynamics is described by (1) with:

\begin{align*}
B(q) &= \begin{bmatrix} b_{11}(q_3) & b_{12}(q_3) \\
        b_{21}(q_3) & b_{22} \\
\end{bmatrix} \\
C(q, \dot{q}) &= h(q_3) \begin{bmatrix} q_3 & (\dot{q}_2 + \dot{q}_3) \\
                   -\dot{q}_2 & 0 \\
\end{bmatrix}
\end{align*}

\[ G(q) = \begin{bmatrix} G_1(q_2, q_3) & G_2(q_2, q_3) \end{bmatrix}^T, \]

where

\begin{align*}
b_{11}(q_3) &= I_{\ell_2} + m_{\ell_2} \ell_2^2 + k_{\ell_2} I_{m_2} + I_{\ell_3} + m_{\ell_3} (a_2^2 + \\
         &\quad + \ell_3^2 + 2a_2 \ell_3 \cos(q_3)) + I_{m_3} + m_{m_3} a_2^2 + k_{m_3} I_{m_3} \\
b_{12}(q_3) &= b_{21}(q_3) = I_{\ell_3} + m_{\ell_3} (\ell_3^2 + a_2 \ell_3 \cos(q_3)) + k_{\ell_3} I_{m_3} \\
b_{22} &= I_{\ell_3} + m_{\ell_3} \ell_3^2 + k_{\ell_3} I_{m_3} \\
h(q_3) &= -m_{\ell_3} \ell_3 \sin(q_3) \\
G_1(q_2, q_3) &= (m_{\ell_2} \ell_2 + m_{\ell_3} a_2) g \cos(q_2) + \\
          &\quad + m_{\ell_3} \ell_3 g \cos(q_2 + q_3) \\
G_2(q_2, q_3) &= m_{\ell_3} \ell_3 g \cos(q_2 + q_3)
\end{align*}

and \( g \) is the gravity acceleration, \( \ell_i, i = 2, 3, \) is the distance from joint to the center of mass of link \( i, a_i, i = 2, 3, \) is the length of link \( i, m_i, i = 1, 2 \) is the mass of link \( i, I_{\ell_i}, i = 2, 3 \) is the moment of inertia relative to a frame attached at the centre of mass of the link \( i \) and aligned with the principle axes of the link, \( I_{m_i}, i = 2, 3 \) is the motor inertia and \( k_{\ell_i}, i = 2, 3 \) is the gear ratio.

Figure 1. ERICC manipulator

3 Control Design

A discrete-time Sliding Mode Controller (SMC) is developed for the trajectory tracking problem in the joint space of the considered planar two-link manipulator. Technical aspects of the proposed solution are stated in the following where details on the design of the control strategy and on its implementation by a predictor for uncertainties compensation are given. In particular AR and ARMA models are used to estimate the uncertainties affecting the robotic system.

Introducing the state vector \( x = [x_1 \ x_2 \ x_3 \ x_4]^T = [q^T \dot{q}^T]^T \), model (1) in the
nominal case can be expressed as:
\[
\dot{x} = f(x) + g(x)u
\]
with:
\[
f(x) = \begin{bmatrix}
x_3 \\
x_4 \\
f_1(x) \\
f_2(x)
\end{bmatrix}, \quad g(x) = \begin{bmatrix}
0 & 0 \\
0 & 0 \\
g_{11}(x) & g_{12}(x) \\
g_{21}(x) & g_{22}(x)
\end{bmatrix}
\]  
\[
\text{where:}
\[
\begin{bmatrix}
f_1(x) \\
f_2(x)
\end{bmatrix} = -B(q)^{-1}[C(q, \dot{q})\dot{q} + F_i \dot{q} + G(q)]
\]
\[
= -B(q)^{-1}S(q, \dot{q})
\]
\[
\begin{bmatrix}
g_{11}(x) & g_{12}(x) \\
g_{21}(x) & g_{22}(x)
\end{bmatrix} = [B(q)]^{-1},
\]
(8)

The discretization of the uncertain model equations with a sampling time \(T_c\) according to standard techniques gives:
\[
q(k+1) = q(k) + T_c \dot{q}(k)
\]
\[
\dot{q}(k+1) = \hat{q}(k) + T_c f(k + \Delta f(k)) + T_c g(k + \Delta g(k))u(k)
\]
(11)

where \(\Delta f(k) = \Delta S(q(k), \dot{q}(k)), \Delta g(k) = \Delta B(q(k))\). \(f(k) = \begin{bmatrix} f_1(k) \\ f_2(k) \end{bmatrix}^T\), with \(f_i(k) = f_i(x(kT_c))T_c, i = 1, 2\), and \(g(k) = \begin{bmatrix} g_{11}(k) & g_{12}(k) \\ g_{21}(k) & g_{22}(k) \end{bmatrix}\), with \(g_{i,j}(k) = g_{i,j}(x(kT_c))T_c, i, j = 1, 2\).

**Assumption A.** \(g(k)\) is an invertible matrix \(\forall q_k, \forall T_c\).

The discretized uncertain model can be expressed equivalently as follows:
\[
q(k+1) = q(k) + T_c \dot{q}(k)
\]
\[
\dot{q}(k+1) = \hat{q}(k) + T_c f(k) + T_c g(k)u(k) + n(k)
\]
(12)

where \(n(k) = T_c(\Delta f(k) + \Delta g(k))u(k)\).

**Assumption B.** The uncertain terms \(\Delta S(q, \dot{q})\) and \(\Delta B(q)\) are bounded by a known constant:
\[
||\Delta S(q, \dot{q})|| \leq \rho_s; \quad ||\Delta B(q)|| \leq \rho_b.
\]

The control law ensuring the robust tracking of a reference variable \(q_d(k) = [x_{1,d}(k) \\ x_{2,d}(k)]^T\) by the sampled position \(q(k) = [x_1(k) \\ x_2(k)]^T\) will be described in the following. Define the discrete-time tracking error as \(\pi(k) = q(k) - q_d(k)\), and consider the following discrete-time sliding surface:
\[
s(k) = \pi(k + 1) - \Lambda \pi(k) = 0
\]
(13)

with \(\text{eig}(\Lambda) = \gamma_i, i = 1, 2\) such that \(|\gamma_i| < 1\). It is easy to verify that the achievement of a quasi-sliding motion on the surface \(s(k) = 0\) implies the ultimate boundedness of the tracking error. The quasi-sliding motion is enforced by the control law \(u(k) = u^{eq}(k) + u^a(k)\), where \(u^{eq}(k)\) is such that \(u^{eq}(k) = 0\) in the nominal case, i.e.
\[
T_c^2 g(k)u^{eq}(k) = -(I - \Lambda)q(k) - T_c(2I - \Lambda)\dot{q}(k) - T_c^2 f(k) - \Delta q_d(k + 1) + q_d(k + 2)
\]
(14)

and \(u^a(k)\) has to be designed in order to account for model uncertainties. To this purpose, the condition \(|s(k + 1)| < |s(k)|\) which enforces a quasi-sliding mode on surface (13) provides, after the insertion of the equivalent control:
\[
||T_c^2 g(k)u^{eq}(k) + T_c^2 \Delta f(k)||
\]
\[
= ||T_c^2 g(k)u^a(k) + n(k)|| = \alpha ||s(k)||,
\]
(15)

where \(0 < \alpha < 1\). Taking the worst case, the term \(u^a(k)\) can be designed as follows:
\[
||u^a(k)|| = \begin{cases}
\theta \left(\frac{||s(k)||}{T_c^2} - \rho_s\right) \frac{1}{\min_4 [||g(k) + \Delta g(k)||]}
\end{cases}
\]
\[
\begin{cases}
\frac{1}{\max_4 [||g(k) + \Delta g(k)||]}
\end{cases}
\]
\[
|\sigma - \alpha s(k))|< \rho_s > 0
\]
(16)

with \(|\theta| < 1, 0 < \alpha < 1\). In other words, when \(||s(k)|| > \rho_s\), i.e. outside the sector, the control law given by (14) and (16) guarantees that \(||s(k)||\) is always decreasing. On the contrary, when \(||s(k)|| \leq \rho_s\), i.e inside the sector, \(||s(k)||\) cannot be made decreasing, and the uncertain term \(n(k)\) is estimated by \(\tilde{n}(k)\), using the procedure described in Section 4. Such an estimation is used in (16) inside the sector, in order to fulfill approximately the condition \(s(k + 1) = \alpha s(k)\). This latter condition guarantees that \(||s(k)||\) decreases with a rate proportional to \((1/\alpha)^k\). Moreover by properly setting \(\alpha\), it is possible to obtain smoother control effort than imposing \(s(k + 1) = 0\).

## 4 Predictor Design

The estimation of \(n(k)\) is performed by a dynamical model, identified online. The following candidate model classes have been chosen [9]:

In equations 17 and 18, 
\[ \hat{n}_i(k) = -a_i^1 n_i(k-1) - \ldots - a_i^d n_i(k-d) \]
i = 1, 2, 
(17)

AutoRegressiveMovingAverage (ARMA):
\[ \hat{n}_i(k) = -a_i^1 n_i(k-1) - \ldots - a_i^d n_i(k-d) + c_i^1 \hat{e}_i(k-1) + \ldots + c_i^d \hat{e}_i(k-d) \]
i = 1, 2, 
(18)
\[ \hat{e}_i(k) = n_i(k) - \hat{n}_i(k), i = 1, 2 \]
is the estimation error.

In equations 17 and 18, \( n(k) = s(k+1) - T_i^2 g(k) u^n(k) \) is the uncertain term expression. To train the model parameters an online identification procedure is chosen, so if disturbances or parametric variations affect the robot, the identification procedure will be able to modify model parameters in order to maintain a proper estimation of uncertain term. Kalman filter is used for this task, as in [12, 2]. In particular, at each sampling interval the model parameters are updated by:
\[ \theta_i(k) = \theta_i(k-1) + K_i(k) (n_i(k-1) - \hat{n}_i(k-1)) \]
i = 1, 2, 
(19)
where \( n_i(k-1), i = 1, 2 \) are the uncertain terms computed at current sampling time by \( n(k-1) = s(k) - T_i^2 g(k-1) u^n(k-1) \). Once the parameters are updated by Kalman Filters, the prediction of \( \hat{n}_i(k), i = 1, 2 \) is computed and used in eq. 16.

Tests are made to tune the covariance matrices of the process and measurements noise. As shown in [7], it is possible to put the process noise covariance, in the form:
\[ Q = q I \]
(20)
where \( I \) is the identity matrix. The measurement noise variance \( r \) is also a scalar. The optimal values for \( q \) and \( r \) are determined by their ratio, \( q/r \), as in [7], instead of the absolute value. This value is determined by trial and error, considering the signal dynamics and the measurement noise. Considering that, from experimental results, the measure uncertainties are very small, determining a big \( q/r \) ratio.

The AR and ARMA model orders can be chosen with an appropriate data analysis and to optimize the accuracy/memory ratio, the MDL index is preferred, as in [16].

5 Experimental Implementation

The proposed controller is implemented on the ERICC robot arms, built by Barras Provence (France). The robot is installed in the Robotics Laboratory at Dipartimento di Ingegneria Informatica, Gestionale e dell’Automazione, Università Politecnica delle Marche. In Fig. 1 is shown the robot with labels indicating the three base joints. Other two joints (not indicated in Fig. 1) are for wrist movements. In this section, the experimental setup and results are discussed.

5.1 Experimental Setup

The considered robot has five degrees of freedom but for the sake of simplicity only links 2 and 3 have been utilized in the experiments. The two considered rotational joints 2 and 3 are actuated by two dc motors with reduction gears. Position measurements are obtained by means of potentiometers and velocity measurements by tachometers. The ERICC command module consists of a power supply module, which provides the servo power for the system; a joint interface module, which contains the hardware to drive the motors and provides sensor feedback from each joint; and a processor module to run user developed software. In order to implement complex control algorithms, a new controller is used in this setup in place of the original ERICC processor module. This system, including hardware and software, combines an experimental apparatus with an easy-to-use software platform based on a dSPACE controller board. In particular the control law is implemented on a dSPACE DS1102 real-time controller board. The sampling time is selected as 0.01 s.

5.2 Implementation and Validation of Predictor

An experiment is performed to determine which model is better for predict the uncertain term \( n(k) \). Data have been acquired on a set of planned trajectories chosen with different shapes and bounds on trajectory derivatives and considering different payload configurations for the robot. From these tests the better fit is obtained with AR models, and the orders are chosen, by means of MDL descriptor [16], of order \( d_1 = 14 \) and \( d_2 = 12 \) for \( n_1(k) \) and \( n_2(k) \) respectively. The Kalman filter for parameter estimation is implemented by setting \( q/r \approx 10^3 \).

As measure of the performance of the proposed estimator the Mean Square of the Error (MSE) \( e(\cdot) = n(\cdot) - \hat{n}(\cdot) = [e_2(\cdot) e_3(\cdot)]^T \) and its Standard Deviation (SD) have been calculated for both links: \( \hat{n}(\cdot) = [\hat{n}_2(\cdot) \hat{n}_3(\cdot)]^T \).

A sample of the performed estimation tests is given in Figs. 2(a) and 3(a) for the estimation of uncertainties \( n_2(k) \) and \( n_3(k) \) related to links 2 and 3, respectively. The Mean Square of the Error (MSE) \( e(\cdot) = [e_2(\cdot) e_3(\cdot)]^T \) and its Standard Deviation (SD) are \( 1.73 \cdot 10^{-5} \) and \( 1.27 \cdot 10^{-4} \) for the joint 2, and \( 1.95 \cdot 10^{-5} \) and \( 1.52 \cdot 10^{-4} \) for the joint 3, respec-
Figure 2. (a) Sample of the uncertain term estimation for joint 2: red line denotes the real value, blue line denotes the predicted value; (b) Sample covariance of the residual $e_2(\cdot)$ obtained by the prediction performed by the AR model. The whiteness test passes with $\varphi = 0.01$.

Figure 3. (a) Sample of the uncertain term estimation for joint 3: red line denotes the real value, blue line denotes the predicted value; (b) Sample covariance of the residual $e_3(\cdot)$ obtained by the prediction performed by the AR model. The whiteness test passes with $\varphi = 0.01$.

5.3 Experimental Results

The task performed in the experiments is a robot joint space trajectory tracking. The smooth trajectories are generated by the nonlinear sliding mode discrete-time system given in [21] that provides smooth trajectories according to user-selectable bounds on trajectories derivatives that can be changed during robot operation. The parameters of the discrete-time SM control for both joints, obtained by trial and error, are reported in Table 1. An experimental result is shown in Figs. 4 and 5. In these figures, the performance produced by the proposed robot controller are illustrated for the

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho_s$</td>
<td>0.0859</td>
</tr>
<tr>
<td>$\min_q |g(k) + \Delta g(k)|$</td>
<td>10.3385</td>
</tr>
<tr>
<td>$\theta_{q2}$</td>
<td>0.1624</td>
</tr>
<tr>
<td>$\theta_{q3}$</td>
<td>0.5019</td>
</tr>
<tr>
<td>$\text{eig}(\Lambda)$</td>
<td>$[0.92 \ 0.91]^T$</td>
</tr>
</tbody>
</table>
robot following a desired trajectory with a payload of 2 Kg. The desired robot motion trajectory has bounds \(|\dot{q}_i| < 35\) deg/s, \(|\ddot{q}_i| < 10\) deg/s\(^2\), \(i = 2, 3\) for joint speed and acceleration, respectively. In particular, the measured trajectories are displayed in Figs. 4(a) and 5(a), the tracking errors are displayed in Figs. 4(b) and 5(b) and the control output voltages are shown in Figs 4(c) and 5(c) for joint \(q_2\) and \(q_3\), respectively. In Fig. 6 norm of the sliding surface are shown for the considered experiment. Fig. 6(a) shows the norm of sliding surface using a classical SMC controller, the figure highlights the residual dynamics not modeled that causes the sliding surface do not remain into the sector. Using the AR model to predict the uncertain term \(\hat{n}(k)\), the sliding surface decreases and remain into the sector for whole experiment as shown in Fig. 6(b).

Comparing with the performance of a robot controller based on a classical discrete-time SMC without uncertain term prediction, i.e. inside the sector the sliding condition is imposed as 0, and based on a standard PID solution, which parameters are given in Table 3 (considering for the robot the same task as in Figs. 4 and 5), the proposed robot controller produces smaller tracking errors as reported in Table 2. In this table the criterion IAE, i.e. the integral of the absolute value of the tracking errors, is used to summarize the above experimental results. The performance improvement of SMC with AR predictor respect PID controller is about 46% and 55% without payload for joint 2 and 3 respectively. This improvement increase in the case of payload experiment to about 50% and 57% for joint 2 and 3 respectively. Improvement in performance are obtained already respect the classical SMC and vary from about 16% to 30%. Another relevant result of the proposed control algorithm is its input disturbance rejection property, that result in IAE performance value of 9.05 and 8.57 for joint 2 and 3 respectively. The PID controller cannot compensate such a disturbance and reach the following IAE values 26.35 and 27.08 for joint 2 and 3 respectively. The considered disturbance amplitude is set to 50% of the controller output for the performed experiment.

<table>
<thead>
<tr>
<th>Joint</th>
<th>Gain</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(q_2)</td>
<td>Proportional</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Integral</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Derivative</td>
<td>1.378</td>
</tr>
<tr>
<td>(q_3)</td>
<td>Proportional</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>Integral</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Derivative</td>
<td>1.425</td>
</tr>
</tbody>
</table>

### 6 Concluding Remarks

In this paper a discrete-time SMC algorithm is proposed for the control of a planar robotic manipulator. A predictor based on AR model, and Kalman filter for on-line parameter estimation, is used to learn about system uncertainties, and ultimate boundedness of the tracking error is guaranteed. The discrete-time SMC control is developed considering a smoothness condition on the sliding surface, that guarantees a smoother control effort than imposing the ideal condition. Moreover using a predictor it is possible to apply the property control input inside the sector and make the sliding surface decreasing. The proposed control law has been implemented on a ERICC robot arm. Experimental evidence shows good trajectory tracking performance as well as robustness in the presence of model inaccuracies, disturbances and payload perturbations, moreover it is shown that the developed controller has a better tracking performance respect to classical formulation of SMC controller and to the most common industrial controller PID.

### References


Figure 4. Joint 2 tracking performance. (a) joint position: red line denotes robot joint 2 trajectory, blue line denotes joint 2 reference trajectory; (b) joint 2 tracking error; (c) joint 2 control output voltage.
Figure 5. Joint 3 tracking performance. (a) joint position: red line denotes robot joint 3 trajectory, blue line denotes joint 3 reference trajectory; (b) joint 3 tracking error; (c) joint 3 control output voltage.

Figure 6. Norm of the sliding surface: red line denotes the threshold $\rho_s$, blue line denotes the norm of the sliding surface $||\bar{x}(k)||$; (a) classical SMC; (b) SMC with AR predictor for uncertainty compensation.