Abstract – Sliding Mode Control design with fuzzy parameter adaptation for a simulated robot hand is presented. The fuzzy adaptation uses the Takagi & Sugeno model in order to determine the parameter values which assure system stability. Using this control strategy, the simulated robot hand will be moved without oscillations, in a similar way as human hand. The proposed control method is stable, good behavior to external disturbances, and does not request to know the system parameters (robot hand parameters) exactly. Also, the paper describes the dynamic of the realistic simulator of the simulated robot hand (5 fingers, 15 degrees of freedom).

I. INTRODUCTION

The development of robotic system has been incremented during the last decades, particularly madding system that simulate human behavior, anatomy and physiognomy. Usually robotic systems are complex and non-linear, like human body, so they require non-linear control strategies to maintain robust and stable movements.

Sliding Mode Control (SMC) is a robust non-linear control method [1,2] that has being applied to the control of robotic systems. A dilemma that SMC present is that the definition of a parameter that defines the stability isn’t deterministic, i.e., the theory only indicates a bound it must satisfy. Many works present control strategies using SMC, where auxiliary methods with SMC are incorporated. In [6] a Fuzzy SMC strategy using Mandami model to control a five degree of freedom (DOF) robotic manipulator is described; the results shows that there are bit oscillations around the reference signal. In [7] the development of a control strategy using model equivalent dynamic, switching control and fuzzy control (with Mandami model) to control a hydraulically actuated mini-excavator joint; the results shows good behavior, but the DOF quantity is just one, so this is not a great approach to a mayor DOF manipulator. A SMC with fuzzy logic (with Mandami model) to control a two DOF robotic manipulator is presented in [8]; the results shows good results but there are oscillations around the references. SMC with Madami model to control a two DOF robotic manipulator is described in [9]; the results show bits oscillations around the reference and a bit permanent error. A control strategy using SMC with neural networks (which use fuzzy logic to determine the values of his parameters) is described to control a two DOF robotic manipulator; the results shows oscillations around the reference [10]. SMC with fuzzy logic (using Mandami model) for determinate de sliding surface boundary layer is described in [11]; results show a bit error with oscillations.

This work presents a new control methodology using Sliding Mode Control with parameter adaptation using fuzzy Takagy & Sugeno model. The control strategy development is applied to move the fingers of a robot hand simulator, with four DOF each one.

Section II describes the robot hand simulator and its dynamic model. Section III describes the development of the Sliding Mode Control using Takagi & Sugeno model, and a brief description of the Fuzzy SMC with Mandami approach and the SMC theory. Section IV defines the criteria used to analyze the method control development with the previous works, also presents the results of the simulations using the strategy development, and finally analyzes the obtained results. Section V presents general conclusion about the strategy development.

II. ROBOT HAND: SYSTEM DESCRIPTION

A. Process Description

The simulator considered to the application of the control method proposed is based in a robot hand made at the Electrical Engineering Department of the University of Chile (see Figure 1).
The functional description of the robot hand is found in [12]. The robot hand mechanism resembles the human anatomy and physiology. Each hand joint is controlled by a tendon whose motor (or muscle in the human case) is positioned in a structure similar to a human forearm.

B. Dynamic Modeling

Each robot hand simulator finger is modeled as a robot manipulator with four DOF, which movements are independent to the others fingers. So, all simulator fingers are a robot manipulator with four components, which are third phalange, second phalange, first phalange and knuckle respectively. The finger dynamic models are developed via the Newton-Euler formulation, which is described in [3]. The absolute rotation axis is defined by the movement direction allowed by the joint that units the knuckle and the palm (see Figure 2.a), and the rest axis rotation are defined by the movement direction allowed by the joints that units the rest of the finger components (see Figure 2.b to the rotation axis in the joint that units the first phalange and the second phalange, the rest rotation axis are similar in the rest of the joints with its respective components).

Therefore, the model of the robot hand fingers has the following form:

$$\Gamma = H(q) \ddot{q} + C(q, \dot{q})$$

(1)

with

$$q, \Gamma, \in \mathbb{R}^4$$

$$H(q) = \begin{bmatrix} H_{11}(q) & 0 & 0 & 0 \\ 0 & H_{22}(q) & H_{23}(q) & H_{24}(q) \\ 0 & H_{32}(q) & H_{33}(q) & H_{34}(q) \\ 0 & H_{42}(q) & H_{43}(q) & H_{44}(q) \end{bmatrix}$$

$$C(q, \dot{q}) = \begin{bmatrix} C_1(q, \dot{q}) \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where $\Gamma$ defines the external torques applied to the joints, $q$ defines the angular rotation of the components throws its respective joints, $H(q)$ defines the inertial components of the fingers [1], and $C(q, \dot{q})$ incorporates the gravity force and the internal forces between finger components.

Thus, the previous dynamic model (1) of the fingers allows to determinate the torque applied to the rotation absolute axis only. Therefore, for determining the torque applied to the rest of the joint it is necessary define a new dynamic model that considerate the fingers without the knuckle, i.e., a robot manipulator with three DOF. Also, the new absolute rotation axis is the movement direction defined by the joint between the knuckle and the first phalange of each finger. In this way the fingers model considering the three components mentioned has the following structure:

$$\Gamma = H(q) \ddot{q} + C(q, \dot{q})$$

(2)

with

$$q, \Gamma, \in \mathbb{R}^3$$

$$H(q) = \begin{bmatrix} H_{11}(q) & H_{12}(q) & H_{13}(q) \\ H_{21}(q) & H_{22}(q) & H_{23}(q) \\ H_{31}(q) & H_{32}(q) & H_{33}(q) \end{bmatrix}$$

$$C(q, \dot{q}) = \begin{bmatrix} C_1(q, \dot{q}) \\ C_2(q, \dot{q}) \\ C_3(q, \dot{q}) \end{bmatrix}$$

The component of (1) and (2) are described more detailed in [13].

C. Robot Hand Simulator

The simulator robot hand movements are generated simulated motors in each joint of the fingers. So rotate the
simulator components it is necessary to define the torque applied to each motor axis, i.e., applied to each finger joint.

The robot hand simulator is made using the basis of OpenGL Library, which is used to draw all the components of the robot hand and all the auxiliary components that could be used in the simulation. The robot hand plane was made in VRML format, so to be read in the development platform is used the CyberX3D library, which allows to use OpenGL library to draw the simulator.

To simulate dynamic characteristic of the robot hand is used the Open Dynamics Engine (ODE) library, which allows imitate mass, external torques and forces, inertia, etc. The visual interface of robot hand simulator is shown in Figure 3.

Figure 3. Robot hand simulator.

III. SLIDING MODE CONTROL BASED ON TAKAGI & SUGENO FUZZY MODEL

A. Sliding Mode Control

Sliding Mode Control strategy defines the sliding surface \( s \), given by the following equation \[1\]:

\[
 s = e + λ\dot{e} \quad (3),
\]

where \( e \) represents the error between the reference and the system state (in this case \( e = q - q_{ref} \)), and \( \dot{e} \) represents the error variation. Thus, the Sliding Mode Control goal is get the null surface for the system surface, i.e. \( s=0=S(t) \).

It is important to mention that sliding surface definition (3) is a particular case applicable to the system models described by (1) and (2). Thus, in this case \( s \) and \( e \) are real vectors with dimension 4 if are referred to (1), or 3 if are referred to (2). The parameter \( λ \) is a real matrix of dimension 4x4 if is used by (1), or 3x3 if is used by (2).

To make that the sliding surface \( s \) be equal to the surface \( S(t) \), \( s \) must satisfy the following condition \[1\]:

\[
\frac{1}{2} \frac{d}{dt} s^2 \leq -\eta |s| \quad (4),
\]

with \( \eta \) a real vector with the same dimension of \( e \).

There are various control construction to satisfy equation (4). Filippov’s strategy control construction theory proposes the next input to the system, which had been used by many works to control robots manipulators \[6,8\]:

\[
\Gamma(t) = \hat{C}(q, \dot{q}) + \mathcal{H}(q)\dot{q} + 2λ\dot{e} -
\]

\[
\dot{e} - k(e, \dot{e}) sat\left(\frac{1}{\phi}\right) \quad (5),
\]

where \( \hat{C}(q, \dot{q}) \) is the estimation value of \( C(q, \dot{q}) \), \( \mathcal{H}(q) \) is the estimation of \( H(q) \), and \( k(e, \dot{e}) \) is a real vector with dimension 4 if is used in (1) or dimension 3 if is used in (2). Function \( sat\left(\cdot\right) \) is a real vector that represents the saturation function and has the following expression:

\[
sat_i(x_i) = \begin{cases} -1, & x_i < -1 \\ x_i, & -1 \leq x_i \leq 1 \\ 1, & x_i > 1 \end{cases} \quad (6),
\]

where \( i \in [1,...,n] \), with \( n \) the dimension of \( e \), and operation \( \langle x \ast y \rangle \) is defined by the following expression:

\[
\begin{pmatrix} x_1 \\ ... \\ x_n \end{pmatrix} \ast \begin{pmatrix} y_1 \\ ... \\ y_n \end{pmatrix} = \begin{pmatrix} x_1 \ast y_1 \\ ... \\ x_n \ast y_n \end{pmatrix} \quad (7)
\]

To satisfy SMC stability condition defined by (4), parameter \( k\left(e, \dot{e}\right) \) must satisfy the following condition \[1\]:

\[
(1 - D_s)\mathcal{F} + \sum_{j=1}^{\mathcal{F}} D_j \mathcal{F}^j_{\mathcal{F}^j} \leq -H^{-1}(q)\dot{q} + \hat{C}(q, \dot{q}) \quad (8)
\]

\[
\sum_{j=1}^{\mathcal{F}} D_j \mathcal{F}^j_{\mathcal{F}^j} + \eta_i
\]

with
Matrix $D$ defined the maximum level to the difference between the vector $H(q)$ and its estimation.

**B. Mandami Approach**

As is shown in (7) there is not an analytical equation to determinate the value of parameter $k\left( e, e^* \right)$. In order to determinate this value, a fuzzy Mandami model is used in previous works [6,8]. The development of this method was made using the Lyapunov Theory, concluding that each rule had the next structure:

$$R[i]: \text{if } s \text{ is } L_j \text{ and if } s' \text{ is } L_k, \text{ then } k_i \text{ is } L_{R[i]} \quad (9),$$

where $R[i]$ is the rule number $i$, $L_w$, $w \in \{j, k, R[i]\}$ are the fuzzy sets defined by the system control expert. There are different methods to desfuzzify, which are chosen depending of the desired results and the system to control.

Thus, the parameter value $k\left( e, e^* \right)$ is given by the output of the Mandami fuzzy rules.

**C. Takagi & Sugeno Fuzzy Approach**

This new approach is based on the explicit characteristics of the system model. By replacing (5) in (2), $\ddot{q}$ can be approximately obtained by the following expression:

$$\ddot{q} = q_{\dot{y}} - \left( \frac{k\left( e, e^* \right)}{\varphi} \right) e - \left( \frac{k\left( e, e^* \right)}{\varphi^2 + \lambda \varphi} \right) e^2 \quad (10)$$

It is very important to mention that this work considers parameter $\lambda$ as a diagonal matrix with its diagonal values equals.

On the hand, in order to obtain the behavior of $\ddot{q}$ as function of $e$ and $e^*$, we define the following relations:

$$(e < 0) \land (e < 0) \Rightarrow q > 0$$

$$(e < 0) \land (e = 0) \Rightarrow q > 0$$

$$(e < 0) \land (e > 0) \Rightarrow q > 0$$

$$(e = 0) \land (e < 0) \Rightarrow q > 0$$

$$(e = 0) \land (e = 0) \Rightarrow q = 0 \quad (11)$$

$$(e = 0) \land (e > 0) \Rightarrow q < 0$$

$$(e > 0) \land (e < 0) \Rightarrow q < 0$$

$$(e > 0) \land (e = 0) \Rightarrow q < 0$$

$$(e > 0) \land (e > 0) \Rightarrow q < 0$$

The relations (11) were developed based on the following assumptions: suppose the error is negative and its variation too, so the system controller desires that the second derivates of the system output would be positive, then its control decisions must be designed to satisfy that condition. This case is the first case in relations (11), and the others follow the same reasoning.

Based on this facts it is possible to get simpler expressions to determinate the value of $k\left( e, e^* \right)$, and the relations reasoning mentioned in previous paragraph, in the next paragraph the design of the controller is described to satisfy these requirements.

Using (11) and (10), the following relations are derived:
Defining the following equations:

\[ f_1(e, \dot{e}) = -\frac{2\lambda e + \lambda^2 e - \dot{q}_{ref}}{e + \lambda e} \cdot \beta > 0 \]

\[ f_2(e, \dot{e}) = -\frac{\lambda^2 e - \dot{q}_{ref}}{e} \cdot \beta > 0 \]

\[ f_3(e, \dot{e}) = -\beta \frac{\lambda e + \lambda^2 e - \dot{q}_{ref}}{\dot{e}} \cdot |\dot{e}| > \lambda |e|, \beta > 0 \]

\[ f_4(e, \dot{e}) = -\frac{\lambda e + \lambda^2 e - \dot{q}_{ref}}{e + \lambda e} \cdot \beta < \lambda |e|, \beta > 0 \]

\[ f_5(e, \dot{e}) = -\frac{\lambda e + \lambda^2 e - \dot{q}_{ref}}{e + \lambda e} \cdot \beta > 0 \]

Thus, replacing (13) in (12) the Takagi & Sugeno fuzzy rules are obtained:

\[ R1: \text{if } (e \text{ is NEG and } \dot{e} \text{ is NEG}) \text{ then } k_1 = f_1(e, \dot{e}) \]

\[ R2: \text{if } (e \text{ is NEG and } \dot{e} \text{ is CE}) \text{ then } k_2 = f_1(e, \dot{e}) \]

\[ R3: \text{if } (e \text{ is NEG and } \dot{e} \text{ is POS}) \text{ then } k_3 = f_1(e, \dot{e}) \]

\[ R4: \text{if } (e \text{ is CE and } \dot{e} \text{ is NEG}) \text{ then } k_4 = f_1(e, \dot{e}) \]

\[ R5: \text{if } (e \text{ is CE and } \dot{e} \text{ is CE}) \text{ then } k_5 = f_1(e, \dot{e}) \]

\[ R6: \text{if } (e \text{ is CE and } \dot{e} \text{ is POS}) \text{ then } k_6 = f_1(e, \dot{e}) \]

\[ R7: \text{if } (e \text{ is POS and } \dot{e} \text{ is NEG}) \text{ then } k_7 = f_1(e, \dot{e}) \]

\[ R8: \text{if } (e \text{ is POS and } \dot{e} \text{ is CE}) \text{ then } k_8 = f_1(e, \dot{e}) \]

\[ R9: \text{if } (e \text{ is POS and } \dot{e} \text{ is POS}) \text{ then } k_9 = f_1(e, \dot{e}) \]

Finally, the value of parameter \( k \) is given by the following expression:
\[
k(\dot{e}, e) = \frac{\sum_{i=1}^{n} \mu_i * f_i(\dot{e}, e)}{\sum_{i=1}^{n} \mu_i}
\]  
(15),

where:

\[
\mu_i = \min(\mu_{i1}, \mu_{i2})
\]  
(16)

IV. APPLICATION

A. Evaluation Basis

The fuzzy sets mentioned in Section III for fuzzy rules (14) are showed in Figure 4.

The new control method approach will be compared with the Nonlinear Linearization Control (NLC) method described in [1]. The values to be compared will be the standard deviation of the difference between system output and the reference of the system. This comparison will be done with a common reference to both strategic, and the values of \(\lambda\) for both methods will be the same.

It’s important to mention that parameters \(\beta\), \(\lambda\) and \(\phi\) were defined by trial-error during several simulations tests.

Also, there will be changes in the third phalange mass to simulate an external charge as perturbation.

The references used as system input are a sinusoidal signal and a reference constant.

B. Simulation Results

The best values for parameters \(\beta\), \(\lambda\) and \(\phi\) are 2, 0.5 and 3 respectively. The error between the system output and the reference are shown in Figure 5 and 6 with a constant input as reference, also is shown the sliding surfaces and the variation of the parameter \(k(\dot{e}, e)\). Figure 7 shows the same with a sinusoidal reference of frequency equal to 0.05[rad/s]. Also that Figure 8 shows the results with mass variation and a constant reference.
Figure 7. System output error, sliding surface and the parameter $k(e, \dot{e})$ of the knuckle with a sinusoidal reference.

Figure 8. System output error (e), sliding surface (s) and the parameter $k(e, \dot{e})$ of the knuckle with a constant reference and a variation of the third phalange mass of 0.1%.

To compare the results of the new control strategic approach with a NLC method, Figure 6 shows the error between a constant reference and the system output. The standard deviation comparison of the error (e) between the two methods is shown in Table 1.

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>$\lambda$</th>
<th>$\phi$</th>
<th>Standard Deviation Takagi and Sugeno SMC approach</th>
<th>Standard Deviation NLC</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.5</td>
<td>3</td>
<td>5.1762</td>
<td>6.2583</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>11.3664</td>
<td>19.1019</td>
</tr>
<tr>
<td>3</td>
<td>0.5</td>
<td>3</td>
<td>5.1860</td>
<td>6.2583</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>5</td>
<td>4.8465</td>
<td>6.2583</td>
</tr>
</tbody>
</table>

C. Comparative Analysis

From Figures 5, 6, 7 and 8, the Takagi & Sugeno SMC approach allows a convergence to reference without oscillations, given by the appropriate parameter values. The results strongly depends of the tuning parameters $\beta$, $\lambda$ and $\phi$ of the proposed controller, but the method allows to simplify the method to determinate the value of $k(e, \dot{e})$.

Also, the complexity of the system defines a reference frequency ranges, and a range to external disturbances.

As can be observed in Table 1, the new approach control strategy presents a better performance in error standard deviation than NLC, which indicate the system error with the proposed approach is smaller than the error with NLC.

Fuzzy Sliding Mode controller based on Takagi & Sugeno method allows to represents a better way the non linealities of a robot manipulator in comparison with the NLC.

V. CONCLUSIONS

The new approach for Sliding Mode Control based on Takagi & Sugeno presents good results depending of the parameters values, the external disturbances and the frequency of the reference. Thus, the system follows the reference...
without oscillations, as a human movements do. Also, define a new method to determinate the value of the parameter \( k(e, \dot{e}) \), obtaining a method that simplifies the way to determinate its value.

For this work, the parameters \( \beta \), \( \lambda \) and \( \phi \) are defined using a simple method, that can be improved in futures research applying optimization method as genetic algorithms.

VI. REFERENCES