Teleoperation with Time Delay: Adaptative Control Law Based on High-Order Sliding Modes

Philippe Fraisse, Arnaud Lelevé

Teleoperation with Time Delay : Adaptive Control Law based on High-Order Sliding Modes

P. Fraisse*, A. Lelevé**
* LIRMM, 161 rue Ada, Montpellier, 34000, France
**ICTT Batiment L. de Vinci 20 av A. Einstein Villeurbanne, 69621, France

Abstract

This paper proposes a solution to stabilize a remote manipulator robot (PUMA560) using IP network. A method is described to control this remote manipulator based on High-Order Sliding Mode so as obtain a product time delay bandwidth constant. The stability condition of this algorithm is studied. Some simulation results illustrate the perfomance of the High-Order Sliding Mode to dynamically force the remote system according to the Network Round Trip Time (NRTT).

1 Introduction

There are situations when firms or laboratories have to resort to remote manipulation. Such cases appear when dangerous objects have to be handled [1] or/and when the environment is too aggressive for humans. Typical applications belong to the nuclear domain (for instance in the dismantling of a nuclear plant), deep-sea domain (work on underwater structures of oil rigs) and spatial domain (exploration of distant planets). Teleoperation has the supplementary advantage of giving the possibility of sharing an experiment between several operators located in distinct places. This way, heavy outdoor experimentations could be easily shared between several laboratories and costs could be reduced as much. However, long distance control of a remote system requires the use of different transmission media which causes two main technical problems in teleoperation : limited bandwidth and transmission delays due to the propagation, packetisation and many other events digital links may inflict on data [2]. Moreover bandwidth and delays may vary according to events occurring all along the transmission lines. In acoustic transmission, round-trip delays greater than 10s and bit-rates smaller than 10kbits/s are common. These technical constraints result in one hand in difficulties for the operator to securely control the remote system and, in the other hand, make classical control laws unstable. Many researches have proposed solutions when delays are small or constant [3], but when delays go beyond a few seconds and vary a lot as over long distances asynchronous links, solutions not based on teleprogramation are fewer because such delays make master and slave asynchronous and the control unstable.

Since 1997, our team has begun a research project on teleoperation. At this time, publications about teleoperation methods when delays are small and/or constant were numerous but there was a lack in long distance teleoperation literature, when delays are prohibitive and variable such as through Internet channels. Starting from an initial experiment with our mobile manipulator, our first work consisted in highlighting practical difficulties inherent to teleoperation of mobile manipulator.

We then began to develop a generic teleoperation model which we inserted in a simulation environment using *Matlab/Simulink*. Working with this tool, we proposed an architecture and a method [1] consisting in buffering data at its arrival in both master and slave. Associated with an adaptive predictor we could offer a real-time estimation of slave state over a virtual constant delay transmission link. In order to validate these results, we developed a software based on this model, which we applied on our mobile manipulator. The main drawback was that this buffering method induced a very high global delay in short distance, due to an unsuitable transmission period. Therefore, we completed our model by a real-time network round-trip delay measure (*Network Round Trip Time, NRTT*) and forward prediction in order to adapt the transmission period (*Network Delay Regulator, NDR*) and buffer parameters to small frequency network behavior. First simulation results were presented in [4].

We propose in this paper an original remote decentralized controller on the manipulator based on High-Order Sliding Modes. This controller allows to adapt the manipulator dynamics in closed loop according to the (*NRTT*). We prove in the next section, under some specific assumptions, that same stability level is obtained if the product closed-loop bandwidth of the remote system by the *NRTT* is constant. We then propose to force dynamically each axis of the robot to verify this condition using the High-Order Sliding Modes algorithm whose the coefficients be adapted with respect to the *NRTT*. Stability is also proved into $3rd$ section. Afterwards, some simulations results are provided to illustrate the efficiency of this solution.

2 Study Case

In this section we introduce the model of the experimental platform including the actuators of the manipulator robot, the network and the base station. Some assumptions are presented to reach the stability level condition. We also describe an original method based on *HSOM* allowing to force dynamically each actuator as a first order linear system.

2.1 Teleoperation Platform

We use a terrestrial vehicle equipped with a 6 degree-of-freedom manipulator (d.o.f) [1]. It consists of a 6x6 vehicle fitted with a *PUMA* manipulator. Several control laws are featured: global motion of the whole mobile manipulator, force-driven control laws, PID control law,... In this present work, we use a computed torque method [5] to control the arm. A dSpace © DSP board is dedicated to the low-level control of the 8 axis (6 for the arm and 2 for the vehicle) A laptop PC fitted with a wireless 2Mbps Ethernet network board takes care of the transmission and of a global control of the whole mobile manipulator. The operator sends and receives data to the remote site at least through his lab network. This data roams through the Internet if necessary, it reaches the remote site LAN and, at last, it is sent at 2Mbps through at 802.11b-like radio transmission link. Figure (1) pictures this trip. We have chosen TCP versus UDP because TCP is reliable: the received data is exactly the same as the sent data (no packet loss or disorder). The main drawback is an higher *NRTT* than with UDP, due to packet checking and reordering.

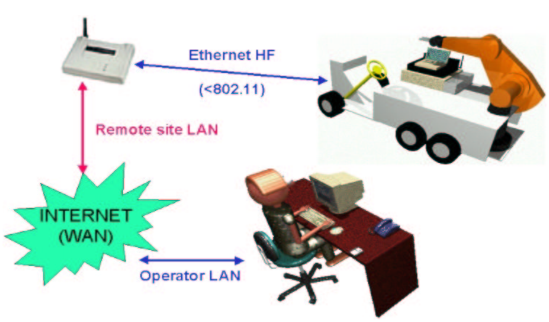

Figure 1: Transmission loop

2.2 Modeling

The modelisation step describes the dynamic model including the network, the remote system and the base station. We consider that the remote manipulator robot as a set of linear subsystems. The nonlinearities (Gravity, Coriolis) are included in the disturbance term.

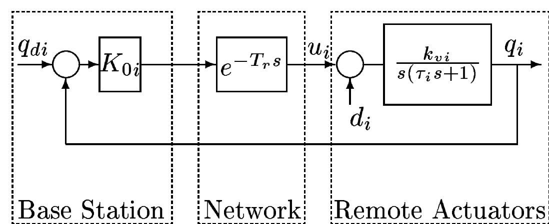

Figure 2: Transmission model

The linear model of the manipulator robot actuators can be expressed as :

$$\tau_i \ddot{q}_i(t) + \dot{q}_i(t) = k_{vi}\left(u_i(t) - d_i(t)\right) \qquad (1)$$

where $\mathbf{u}(t) = [u_1..u_6]^t$ is the control vector, $\mathbf{q}(t) = [q_1..q_6]^t$ the position output vector, $\mathbf{d}(t)$ the disturbance vector and τ_i the time response of each actuator. The base station is always controlled by an human operator whose explicit modeling is complex. However, the stability study needs an explicit model. Also, we assume the base station as a proportionnal loop and the network reacts as a constant delay: either we use a channel that provides constant delay T_r, or we use our *NDR* wich provides a constant virtual delay T_r. We obtain the following equation :

$$\mathbf{u}(t) = \mathbf{K}_0 \mathbf{E}(t) e^{-T_r s} \qquad (2)$$

with $\mathbf{E}(t) = \mathbf{q}_d(t) - \mathbf{q}(t)$ the tracking error and a constant global delay T_r.

2.3 Stability with constant time delay

The stability condition of the second-order system with constant time delay described by the above equations (1) and (2), provides an equation where the bandwidth of the remote system ($\beta_i = \frac{1}{\tau_i}$) according to the delay T_r is nonlinear [6]. However, in the case of the remote system is composed by a set of first-order systems such that :

$$\tau_i \dot{q}_i(t) + q_i(t) = k_{vi} u_i(t) \tag{3}$$

We then find the following stability condition :

$$T_r \leq \frac{\tau_i \arccos(-\frac{1}{K_{0i} k_{vi}})}{K_{0i} k_{vi} \sin \arccos(-\frac{1}{K_{0i} k_{vi}})} \tag{4}$$

Assume that gains K_{0i} and k_{vi} are constant, we can rewrite the above equation (4) as :

$$T_r \beta_i \leq C_i \tag{5}$$

where C_i is a constant positive value. In order to verify the equation (5) when Tr takes different values, the bandwidth β_i has to be adapted according to T_r.

3 Algorithm based on High-Order Sliding Modes

One solution to dynamically force the response of the remote actuators to react as a first-order is the sliding modes control law [7]. This method, when the sliding condition is verified, dynamically force the response of the system such that the sliding surface $s = 0$. However, the sliding condition is obtained with a discontinuous control vector which causes a chattering phenomenon at the equilibrium point. Several works try to vanish this problem, but only one method gives a satisfactory result. This method is called High-Order Sliding Modes *HOSM* [8]. The main goal is the constraint given by the equation $s(x) = 0$ has to be a sufficiently smooth constraint function. The discontinuity does not appear in the first $^{(r-1)th}$ total time derivative (cf. 6).

$$s = \dot{s} = \ddot{s} = ... = s^{r-1} = 0 \tag{6}$$

The relative degree r is defined as :

$$\frac{\partial s^i}{\partial u} = 0, \; \frac{\partial s^r}{\partial u} \neq 0 \tag{7}$$

with $i = 1..(r-1)$. In the case of $r = 1$ in order to avoid the chattering phenomenon, the 2-sliding mode control can be used and the time derivative of the plant control $\dot{u}(t)$ may be considered as the actual control vector. In the 2-sliding mode the control vector $u(t)$ is continuous and the chattering phenomenon disappears.

3.1 Control Law Strategy

Let consider the state representation of the equation (1).

$$\begin{cases} \dot{x}_{1i} = x_{2i} \\ \dot{x}_{2i} = -\frac{1}{\tau_i} x_{2i} + \frac{1}{\tau_i} k_{vi} u_i(t) \end{cases} \tag{8}$$

Where $x_{1i} = q_i$ with $i = 1..6$ is the joint position of the i^{th} actuators, τ_i is the time response of the motor and $u_i(t)$ the control variable. We propose a first-order surface equation with the desired output value $x_{1di} = 0$:

$$s_i(x_i) = -x_{2i} - \lambda_i x_{1i} \tag{9}$$

The 2-sliding mode control considers the time derivative control variable $\dot{u}(t)$ as the control vector. One solution is obtained expressing $\ddot{s}$ with equation (8). The second one is achieved considering the time derivative state model (8) such that the time derivative of the control variable $\dot{u}_i(t)$ appears within $\dot{s}$. We obtain in the second case, the following model:

$$\begin{cases} \dot{x}_{1i} = x_{2i} \\ \dot{x}_{2i} = x_{3i} \\ \dot{x}_{3i} = -\frac{1}{\tau_i} x_{3i} + \frac{1}{\tau_i} k_{vi} \dot{u}_i(t) \end{cases} \tag{10}$$

Working on the new model defined previously (eq.10), we propose a new sliding s_y surface depending on the $s_i(x_i, t)$ function as :

$$s_{yi} = y_{2i} + \alpha_i y_{1i} \tag{11}$$

where $y_{2i} = \dot{s}_i$ and $y_{1i} = s_i$. This sliding surface can be expressed according to the state variables as :

$$s_{yi} = -x_{3i} - (\alpha_i + \lambda_i) x_{2i} - \alpha_i \lambda_i x_{1i} \tag{12}$$

The time derivative control vector $\dot{s}_i(t)$ can be written as the sum of two functions :

$$\dot{u}_i(t) = \dot{u}_{eqi} + \Delta \dot{u}_i \tag{13}$$

where $\dot{u}_{eqi}$ is the equivalent control to achieve $\dot{s}_i = 0$:

$$\dot{u}_{eqi} = \frac{\tau_i}{k_{vi}} \left(\frac{x_{3i}}{\tau_i} - (\lambda_i + \alpha_i) x_{3i} - \alpha_i \lambda_i x_{2i} \right) \tag{14}$$

and $\Delta \dot{u}_i$ is the discontinuous second term used to compensate the unmodelled dynamics of the system :

$$\Delta \dot{u}_i = K_i sgn(s_{yi}) \tag{15}$$

where K_i is the maximum input value of the actuators. The η-reachability equation defined below (16) is a necessary condition for a finite time convergence.

$$s_{yi} \dot{s}_{yi} \leq -\eta |s_{yi}| \tag{16}$$

The equations (14) and (15) verify the condition (16) which ensures the finite time convergence. Integrating equation (16), we obtained the following result :

$$t_{ei} \leq \frac{\alpha_i \lambda_i \tau_i}{k_{vi} K_i} \tag{17}$$

where t_e is the upper bound for the time convergence such that the surface $s_i(t_{ei}) = 0$. Using equations (13), (14) and (15), the global control vector $u_i(t)$ can be expressed as :

$$u_i(t) = u_{eqi}(t) + \int K_i sgn(s_{yi}) dt$$

This control vector is continuous and the chattering phenomenon disappears.

3.2 Stability

Let define a Lyapunov function such that :

$$V(\mathbf{X}, t) = \frac{1}{2} \mathbf{X}^{\mathrm{t}} \mathbf{P} \mathbf{X}$$

with $\mathbf{X} = [x_1, x_2, x_3]^T$ the state vector and $\mathbf{P}$ a matrix defined by :

$$\mathbf{P} = \begin{bmatrix} 0 & 0 & -\alpha_i \lambda_i \\ 0 & \alpha_i \lambda_i - \frac{\alpha_i \lambda_i}{\tau_i} & -(\alpha_i + \lambda_i) \\ -\alpha_i \lambda_i & -(\alpha_i + \lambda_i) & -1 \end{bmatrix}$$

The time derivative term of the Lyapunov function can be written as :

$$\dot{V} = (-\alpha_i - \lambda_i + \frac{1}{\tau_i}) x_{2i}^2 + \frac{\alpha_i \lambda_i}{\tau_i} x_{1i} x_{3i} - \frac{k_{vi} K_i}{\tau_i} |s_{yi}|$$

where V is positive defined and $\dot{V}$ is negative defined under the following condition :

$$\frac{\alpha_i \lambda_i}{\alpha_i + \lambda_i} < \frac{1}{\tau_i} \tag{18}$$

We therefore conclude, under the stability condition (eq. 18) that the system is exponentially stable.

4 Simulation Results

First of all, we validate in simulation the theorical results obtained previously (stability, finite time convergence). Afterwards, we introduce global simulation results made up of base station, network and remote system using the *HOSM* control law whose the sliding surface coefficients are adapted according to *NRTT*, verifying equation (5).

4.1 High-Order Sliding Mode Algorithm

A Simulink model based on the equivalent model of the remote actuators defined by the equation (10) allows to verify the stability equation. This condition achieved in (18) can be rewritten as :

$$\alpha_i < \frac{\lambda_i}{\lambda_i \tau_i - 1} \tag{19}$$

We have performed the model by tuning the coefficients λ_1 and α_1 in order to reach the stability limit. We present the figure (3) these values. This simulation result is achieved with $\tau_1 = 0,1s$, $K_1 = 15V$ and $k_{v1} = 1rd/V$. We also added the theorical points of the stability condition. These curves show a similar behavior between the simulator and the theorical result.

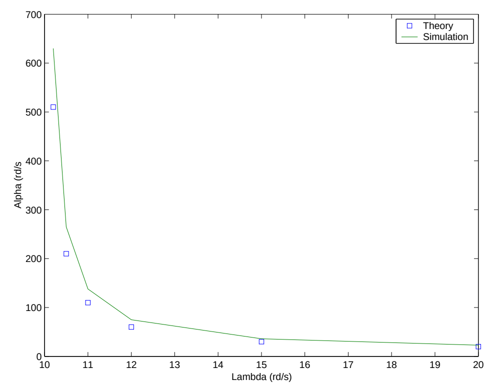

Figure 3: Stability limit

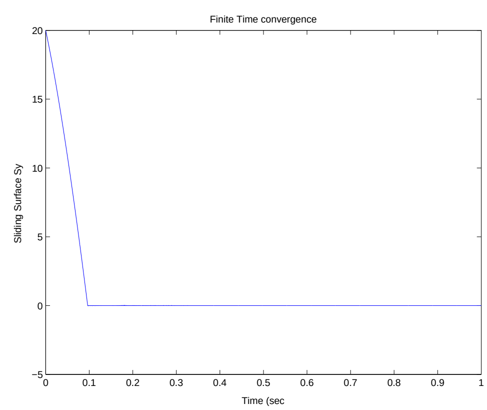

Figure 4: Finite time convergence

The finite time convergence observation is carried out with $\alpha_1 = 1rd/s$ and $\lambda_1 = 20rd/s$. Using previous values of the actuator model and equation (17), we achieve $t_{e1} = 0,13s$. This simulation result is showed figure (4) where $(te \sim 0,1s)$. Figure (5) presents the sliding surface phase trajectory $\dot{s}_1(t) = f(s_1(t))$. This sliding surface describes in the phase trajectory a straight line such that $\dot{s}_1 = -\alpha_1 s_1$, with $\alpha_1 = 2rd/s$.

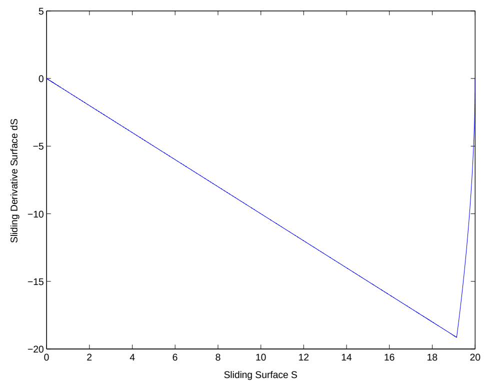

Figure 5: Sliding surface phase trajectory

The time step response of the remote system is then forced dynamically by the sliding surface $s_{y1}(t)$ (eq.(12)). Reaching the sliding straight, the time response is defined by the solution of the ordinary differential equation $s_{y1}(t) = 0$.

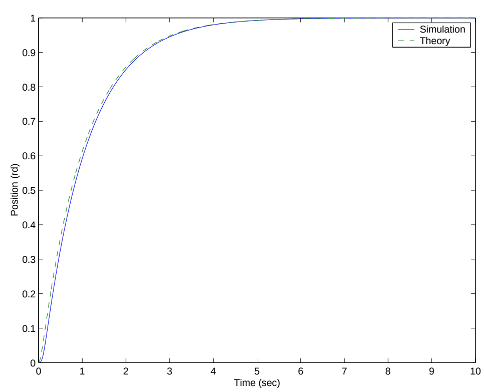

Figure 6: Position response

In the case of this step response $(x_{1d}(t) = u(t))$, the time solution can be expressed as :

$$x_{1T}(t) = u(t)\left(1 - \frac{\alpha_1}{\alpha_1 - \lambda_1}e^{-\lambda_1 t} + \frac{\lambda_1}{\alpha_1 - \lambda_1}e^{-\alpha_1 t}\right)$$

If the coefficient $\alpha_1 << \lambda_1$, the time response of x_{1T} is equivalent to the first-order time response. This theorical time response $x_{1T}(t)$ is showed figure (6) versus the simulation result.

The simulation time response $x_{1T}(t)$ is closed to the theorical time response of the sliding surface. That proves the position trajectory is always defined by the sliding surface when $t > t_e$. It is an important result given in the worst case when the operator sends brutal step orders in a teleoperation task.

4.2 Switching of sliding surfaces

When the constant delay T_r is changed by the NDR, the $HOSM$ algorithm has to modify each α_i. Figure (7) shows a time step response of a sliding surfaces switching with two values of α_1 : $\alpha_1(1) = 0,4rd/s$ and $\alpha_1(2) = 2rd/s$ with $\lambda_1 = 20rd/s$.

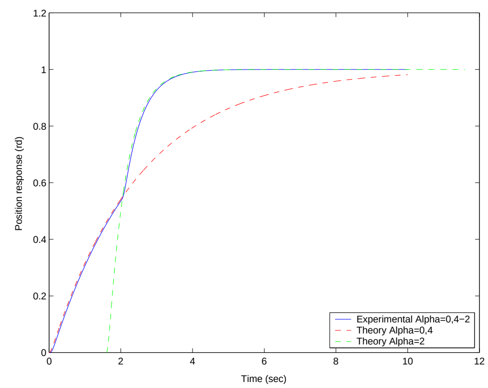

Figure 7: Position response

The experimental curve is very close to theorical responses. This remark allows us to conclude that the sliding condition is always verified even during the switching period.

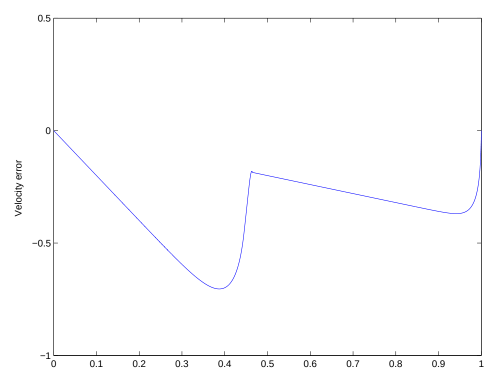

Figure 8: Switching of sliding surfaces

Figure (8) presents the switching of sliding surface with $-x_{21}(t) = f(1 - x_{11}(t))$.

4.3 Teleoperation with time delay

We have performed this control algorithm based on *HOSM* method on a simulation of teleoperation task. The operator sends some desired position steps (see figure (9)) to the remote manipulator.

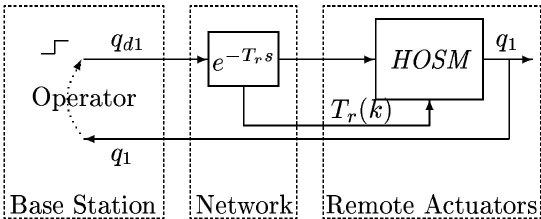

Figure 9: Teleoperation task

We assume that the network delay (T_r) takes two different values during the simulation time : $T_r(1) = 1s$, $t \leq 12s$ and $T_r(2) = 250ms$, $t > 12s$. This can happen when the network behavior changes drastically and the *NDR* has to switch to a new Tr to be able to provide a constant virtual delay anew.

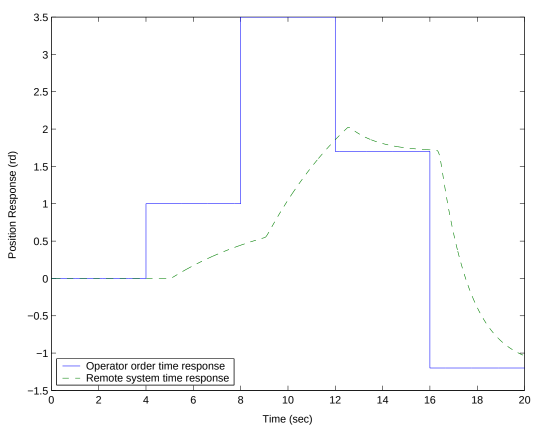

Figure 10: Effects of the *HOSM* on $q_1(t)$

The equation (5) gives a solution to keep a same stability level when the average delay changes. Choosing $C_1 = 0,2$ for the base station, we obtain two values of the sliding surface coefficient $\alpha_1(1) = 0,2rd/s$ and $\alpha_1(2) = 0,8rd/s$. Figure (10) illustrates a teleoperation task using a *HOSM* algorithm. The bandwidth of the remote system is adapted with respect to the network delay. The operator has always a stable system and if the mean delay tends to infinity then the manipulator bandwidth (α) will tend to zero.

5 Conclusion

The method we have proposed in this paper concerns a global stable solution to remote control a manipulator robot via an asynchronous network such Internet. This method assumes a slow step varying delay. This way, it can be used as an upper layer of previous low-level works such as our *NDR*, upon a real network with jitter. This study should be implemented on experimental setup in fews months to validate these results. This control method associated with the previous results (*NDR*), provides an efficient tool for the telerobotics systems using TCP/IP network.

References

[1] A. Lelevé, P. Fraisse, A. Crosnier, P. Dauchez, F. Pierrot, "Towards Virtual Control of Mobile Manipulators", *Proc. of the IEEE ICRA'98*, Leuven, Belgium, pp.2971-2976, 1998.

[2] R. Oboe, P. Fiorini, "Internet-based telerobotics: problems and approches", *Proc. International Conference on Advanced Robotics*, Monterey, USA, pp765-770, July 1997.

[3] G. Niemeyer, J.J. Slotine, "Designing Force Reflecting Teleoperators with large time delays to appears as virtual tools", *Proc. of the IEEE ICRA'97*, pp. 2212-2218, 1997.

[4] A. Lelevé, P. Fraisse, P. Dauchez, "Telerobotics over IP Networks: Towards a low-level Real-Time Architecture", *Proc. of the IROS'01, IEEE/RSJ*, Maui, Hawaii, USA. Oct.2001.

[5] E. Freund, "Fast nonlinear control with arbitrary pole placement for industrial robots and manipulators", *The International Journal of Robotics Research*, Vol. 1(1), pp.65-78, 1982.

[6] A. Lelevé, "Contributions à la téléopération de robots en présence de délais de transmission variables", *Thèse de doctorat, Université Montpellier II, in french*, December 2000.

[7] V. Utkin,"Variable structure systems with sliding modes", *IEEE Trans. on Automatic Control*, Vol. AC-22, No2, pp212-222, 1977.

[8] L. Fridman, A. Levant, "High-Order Sliding Modes", in Sliding Modes control in Engineering, *Ed. W. Perruquetti, J.P. Barbot, Marcel Dekker, Inc., New York*, pp.53-101,2002.