Two Adaptive Robust Sliding Mode Controllers for Robot Manipulators*

Takayuki FURUTA** and Ken TOMIYAMA**

Sliding mode controllers are known to be capable of achieving robustness against modeling errors, plant disturbances and plant parameter variations. They, however, require upper-bounds of uncertainties in plant parameters to specify feedback gains that realize robustness. These upper-bounds are often chosen conservatively and, as a result, high feedback gains and oscillations, referred to as chattering, occur. Recently, on-line identification of the upper-bounds of uncertainties have been successfully incorporated into sliding mode regulators to alleviate the chattering. Here, we propose to extend this on-line identification technique to two types of controllers, called adaptive robust sliding mode controllers, for trajectory control of robot manipulators. The stability properties of the proposed controllers are demonstrated using Lyapunov functions and are implemented using a PUMA-type manipulator. The performance of these controllers is compared with that of regular sliding mode controllers. The result of comparison reveals excellent robustness and chattering suppression of the proposed controllers.

Key Words: Adaptive Controller, Sliding Mode Controller, Robot Manipulator, VSS

1. Introduction

Robust controllers have been a subject of intensive study for some time. They possess numerous advantages compared with conventional optimal controllers. For example, they are 'robust' against system parameter variations and unknown perturbations. Among typical robust controllers is the VSS (Variable Structure System) controller. The idea of VSS controllers is to achieve robust characteristics by discontinuously altering their control laws based on some criterion. This controller gained popularity after Utkin proposed it in his paper⁽¹⁾ in 1977. The sliding mode controller is one of the more successful VSS-type controllers. It constricts the system trajectory to a specified hyperplane in the state space and, as a result, makes the system response insensitive to parameter variations and external disturbances. The hyperplane is defined in such a way that the state

trajectory on the hyperplane possesses some desirable characteristics such as asymptoticity to the zero state as in the case of a regulator.

One of the major shortcomings of sliding mode controllers is the high frequency oscillation, referred to as 'chattering', near the hyperplane due to the use of high gains to force the system to traverse along the hyperplane. This is highly undesirable for both plant mechanisms and control objectives such as the path control of robot manipulators.

Adaptive controllers, on the other hand, achieve excellent performance by adjusting model parameters to changing system characteristics on-line. Recursive identification of the unknown parameters is the key technology in those controllers.

Attempts have been made to combine the two approaches in order to maximize their advantages (Ref. (2) for example). One attempt is to adaptively identify upper bounds of all the system uncertainties and set the feedback gains of a sliding mode controller accordingly⁽³⁾. This approach is valid for all nonlinear systems and has the effect of lowering feedback gains and, as a result, decreasing chattering. Application of this approach to a manipulator with this controller,

* Received 5th January, 1996.

** Department of Mechanical Engineering, Aoyama Gakuin University, 6-16-1 Chitosedai, Setagaya, Tokyo 157, Japan

however, does not take full advantage of the special features of manipulator dynamics. Another approach is based on the fact that a sliding mode controller can be constructed as a combination of two parts; one based on the Lagrangian dynamics and the other sliding mode generator. The burden of the sliding mode control part can be reduced by improving the accuracy of the Lagrangian model of the manipulator. This can be achieved by using recursive identification of parameters in the Lagrangian dynamic model⁽⁴⁾.

A clever approach is to identify the parameters in the Lagrangian dynamics equation and the upper bounds of other unknown dynamics of a manipulator and use both to construct an adaptive robust regulator⁽⁵⁾. This controller can achieve robustness without chattering and possesses advantages over other controllers. Here, we extend the results in Chen⁽⁵⁾ from a regulator problem to a trajectory tracking problem and implement the resulting control system on a robot manipulator to demonstrate its expected characteristics.

Another control approach we will try on our manipulator is based on state space descriptions of nonlinear systems⁽⁶⁾. We assume that the state space description can be divided into a linear part, a nonlinear part that is linear in unknown parameters, and an unknown dynamic part including disturbances. This controller has an advantage over the one suggested above in that it is applicable to a wider class of nonlinear systems.

Derivations of two controllers are presented first and the results of trajectory following experiments using the two controllers as well as traditional robust controllers are presented and compared in the sequel.

2. Lagrangian Dynamics of Manipulators

Assuming that the joint angles are taken as generalized coordinates, the dynamics of a robot manipulator can be expressed by the Lagrangian dynamics model given by

$$M(\theta) \cdot \ddot{\theta} + C(\theta, \dot{\theta}) \cdot \dot{\theta} + D \cdot \dot{\theta} + G(\theta) = \tau \quad (1)$$

where θ and τ are the joint angle and joint torque vectors and the terms on the lefthand side are the moment of inertia, the combined centrifugal and Coriolis force, the combined viscous and Coulomb friction term, and the gravitation term, respectively. It is noted that the time dependencies of variables are not indicated unless deemed necessary for the sake of simplicity.

There are several known properties of this equation that are exploited in later steps. First, M is a symmetric positive definite matrix regardless of the value of θ . Second, M and C are not independent and satisfy the following relationship,

$$x^T [\dot{M}(\theta) - 2C(\theta, \dot{\theta})] x = 0 \quad (2)$$

for an arbitrary vector x . Furthermore, Eq.(1) can be expressed as a linear equation of system parameters by suitably defining them as,

$$Y(\theta, \dot{\theta}, \ddot{\theta}) \cdot a = \tau \quad (3)$$

where a is the vector of system parameters and Y is a coefficient matrix whose elements are (nonlinear) functions of θ and its first and second derivatives.

The control of the manipulator begins with the trajectory planning to generate a reference trajectory x_d of the end-effector in the world coordinate and then translates it to the joint coordinate θ_d through the use of inverse kinematics. Then, the inverse dynamics of the manipulator is used to compute the reference torque τ . This portion of the control constitutes the feed-forward control which is common to all the control schemes used in this paper.

3. Sliding Mode Controller based on Lagrangian Dynamics

A sliding mode controller for a robot manipulator whose dynamic equation is given by Eq.(1) is presented in this section. First, we define the link angle error e by,

$$e = \theta - \theta_d \quad (4)$$

where θ_d is the desired value of θ . Next, we define the sliding variable vector $s(t)$ that specifies a sliding hyperplane.

$$s = \dot{e} + \lambda e \quad (5)$$

where a positive number λ is a design parameter. Note that, on the hyperplane defined by the equation $s = 0$, the error $e(t)$ is asymptotic to 0 since $\lambda > 0$.

The objective of the sliding mode controllers is to steer the state to the hyperplane. Once on the hyperplane, the system dynamics are solely governed by the parameter λ and become robust against parameter variations and external disturbances. The control law proposed by Slotine⁽⁷⁾ is as follows:

$$\tau = Y(\theta, \hat{\theta}, \hat{\theta}_r, \hat{\ddot{\theta}}_r) \cdot \hat{a} - Ks \quad (6)$$

where $(\hat{\cdot})$ indicates estimated quantities, K is an arbitrary positive definite matrix and θ_r and Y are defined as follows.

$$\hat{\theta}_r = \hat{\theta}_d - \lambda e \quad (7)$$

$$Y(\theta, \hat{\theta}, \hat{\theta}_r, \hat{\ddot{\theta}}_r) \cdot \hat{a} = \hat{M}(\theta) \hat{\ddot{\theta}}_r + \hat{C}(\theta, \hat{\theta}) \hat{\dot{\theta}}_r + \hat{D} \cdot \hat{\dot{\theta}} + \hat{G}(\theta) \quad (8)$$

It is noted that this control law does not require the measurement of the angular acceleration because of the introduction of the variable θ_r .

The corresponding parameter adaptation law is given as follows.

$$\frac{d\hat{a}}{dt} = -\Gamma Y^T(\theta, \hat{\theta}, \hat{\theta}_r, \hat{\ddot{\theta}}_r) s \quad (9)$$

where Γ is a positive definite weighting matrix that governs the speed of adaptation. Slotine showed the

stability of this sliding mode controller using a Lyapunov's V-function⁽⁷⁾.

4. Adaptive Robust Sliding Mode Controller-I

We derive one of the two proposed robust controllers here. It is based on Slotine's sliding mode controller and has an added estimator of the upper-bounds of system uncertainties. The problem considered here is a tracking problem rather than a regulator problem⁽⁵⁾ with application of trajectory control of a PUMA-type manipulator in mind.

The manipulator dynamics given by Eq. (1) are modified here to include a term h that represents system uncertainties such as unknown dynamics caused by system anomalies and external disturbances.

$$M(\theta) \cdot \ddot{\theta} + C(\theta, \dot{\theta}) \cdot \dot{\theta} + D \cdot \dot{\theta} + G(\theta) + h(\theta, \dot{\theta}, t) = \tau \quad (10)$$

or

$$Y(\theta, \dot{\theta}, \ddot{\theta}) \cdot a + h(\theta, \dot{\theta}, t) = \tau \quad (11)$$

Let us introduce an upper-bound function ρ for h as follows.

$$|h(\theta, \dot{\theta}, t)| + \phi \leq \rho(\theta, \gamma, t) \quad (12)$$

where ϕ is a positive constant.

The upper-bound function ρ is a concave positive differentiable function of γ . This can be expressed in equation form as follows.

$$(\hat{\gamma} - \gamma) \frac{\partial \rho}{\partial \gamma} \Big|_{\hat{\gamma}} \leq [\rho(\theta, \hat{\gamma}, t) - \rho(\theta, \gamma, t)] \quad (13)$$

$$\rho(\theta, \hat{\gamma}, t) > 0 \quad (14)$$

Both h and ρ can be time varying functions. The upper-bound is estimated by identifying the value of γ . In this study, we use the following function as the upper-bounded function ρ

$$\rho(\gamma) = K_1 \gamma^2 + K_2 \quad (15)$$

where K_1 and K_2 are positive constants.

With these preparations, the first proposed adaptive robust sliding mode controller is stated in the following theorem.

Theorem 1

The system with the following control law, system parameter adaptation law, and upper-bound estimate adaptation law is asymptotic to the buffer region defined by $\|s\| < \lambda\epsilon$. Furthermore, e approaches the buffer region defined by $\|e\| \leq \epsilon$ simultaneously.

Control Law:

$$\tau = Y(\theta, \dot{\theta}, \ddot{\theta}_r) \hat{a} - \left[\frac{\rho(\theta, \hat{\gamma}, t)}{\lambda \cdot \epsilon} \right] \cdot s \quad (16)$$

Parameter Adaptation Law:

$$\frac{d\hat{a}}{dt} = \begin{cases} -\Gamma_1 \cdot Y^T(\theta, \dot{\theta}, \ddot{\theta}_r) \cdot s, & \text{for } \|s\| \geq \lambda \cdot \epsilon \\ 0, & \text{for } \|s\| < \lambda \cdot \epsilon \end{cases} \quad (17)$$

Upper-Bound Adaptation Law:

$$\frac{d\hat{\gamma}}{dt} = \begin{cases} \Gamma_2 \cdot \frac{\partial \rho}{\partial \gamma} \Big|_{\hat{\gamma}} \cdot \|s\|, & \text{for } \|s\| \geq \lambda \cdot \epsilon \\ 0, & \text{for } \|s\| < \lambda \cdot \epsilon \end{cases} \quad (18)$$

where Γ_1 and Γ_2 are appropriate positive definite weighting matrices and ϵ is a parameter to define a buffer region in which the adaptation of parameters and upper-bounds are temporarily sustained as shown in the above equations.

Proof: First, we consider the behavior of s . Second, we consider that $\|e\|$ approaches a region that is determined by $\|s\|$. Finally, we show that this region approaches the buffer region defined by $\|e\| < \epsilon$.

Case 1. $\|s\| \geq \lambda\epsilon$ (Outside of the buffer region)

Consider the following candidate of the Lyapunov's V-function.

$$V(s) = \frac{1}{2} [s^T M(\theta) s + \tilde{a}^T \Gamma_1^{-1} \tilde{a} + \tilde{\gamma}^T \Gamma_2^{-1} \tilde{\gamma}] \quad (19)$$

where $(\hat{\cdot}) = (\hat{\cdot}) - (\cdot)$. The time derivative of this V-function can be shown to be negative using the above three laws and equations (2), (8), (11), (12), and (13).

$$\begin{aligned} \dot{V}(s) &= s^T [M\dot{s} + C s] + \tilde{a}^T \Gamma_1^{-1} \dot{\tilde{a}} + \tilde{\gamma}^T \Gamma_2^{-1} \dot{\tilde{\gamma}} \\ &= s^T [\tau - Y(\theta, \dot{\theta}, \ddot{\theta}_r) \cdot \hat{a} - h(\theta, \dot{\theta}, t)] \\ &\quad + \tilde{a}^T \Gamma_1^{-1} \dot{\tilde{a}} + \|s\| \tilde{\gamma}^T \frac{\partial \rho}{\partial \gamma} \Big|_{\hat{\gamma}} \\ &\leq -s^T \left[\frac{\rho(\theta, \hat{\gamma}, t)}{\lambda \epsilon} \right] s + \|s\| [\rho(\theta, \gamma, t) - \phi] \\ &\quad + \|s\| \tilde{\gamma}^T \frac{\partial \rho}{\partial \gamma} \Big|_{\hat{\gamma}} \\ &\leq -\|s\| \left\{ \frac{\rho(\theta, \hat{\gamma}, t)}{\lambda \epsilon} \|s\| - \rho(\theta, \gamma, t) + \phi \right\} \\ &\quad + \|s\| \tilde{\gamma}^T \frac{\partial \rho}{\partial \gamma} \Big|_{\hat{\gamma}} \\ &\leq -\|s\| \{ \rho(\theta, \hat{\gamma}, t) - \rho(\theta, \gamma, t) + \phi \} \\ &\quad + \|s\| \tilde{\gamma}^T \frac{\partial \rho}{\partial \gamma} \Big|_{\hat{\gamma}} \\ &\leq -\|s\| \phi \end{aligned} \quad (20)$$

Since, ϕ is positive, the time derivative of the V-function is non-positive outside of the boundary layer in the region defined by $\|s\| \geq \lambda\epsilon$. This guarantees that the state approaches the boundary layer.

Next, we show that $\|e\|$ is asymptotic to a region and that this region shrinks to $\|e\| < \epsilon$ in infinite time. First, we consider the region at a time t_1 defined by $\|s(t_1)\| < \lambda\epsilon + \eta$, where $\eta > 0$. By calculating the time derivative of the Lyapunov's V-function given by Eq. (21), we can show the following result.

$$V(e) = \frac{1}{2} e^T e \quad (21)$$

$$\begin{aligned} \dot{V}(e) &= e^T \dot{e} \\ &= e^T (s - \lambda e) \\ &\leq \|e\| (\|s\| - \lambda \|e\|) \\ &< \|e\| (\lambda \epsilon + \eta - \lambda \|e\|) \\ &< \lambda \|e\| \left(\frac{\lambda \epsilon + \eta}{\lambda} - \|e\| \right) \end{aligned} \quad (22)$$

This result shows that in the region $\|s(t)\| < \lambda\epsilon + \eta$, e is asymptotic to the region defined by $\|e\| \leq (\lambda\epsilon + \eta)/\lambda$. Furthermore, since s goes to the region $\|s\| < \lambda\epsilon$, η goes to zero. This completes the proof of Theorem 1.

Q.E.D.

5. Adaptive Robust Sliding Mode Controller-II

The adaptive sliding mode regulator proposed in [Chen, 1993]⁽⁶⁾ is extended to a tracking problem in this section. The controller relies on a particular state space representation of nonlinear systems given by,

$$\dot{x} = A(t)x(t) + Bf(t, x) + Bh(t, x) + Bu(t) \quad (23)$$

where $f(t, x)$ is a known function that is linear in unknown parameters σ as,

$$f(t, x) = g(t, x)\sigma \quad (24)$$

and $h(t, x)$ represents the unknown nonlinear dynamics. It is assumed that $h(t, x)$ is bounded by a concave positive differentiable function ρ as shown.

$$|h(t, x)| \leq \rho(t, x, \gamma) \quad (25)$$

$$(\bar{\gamma} - \gamma) \frac{\partial \rho}{\partial \gamma} \Big|_{\bar{\gamma}} \leq [\rho(t, x, \bar{\gamma}) - \rho(t, x, \gamma)] \quad (26)$$

$$\rho(t, x, \bar{\gamma}) > 0 \quad (27)$$

where the parameter γ specifies the upper-bound. Both the system parameters σ and the upper-bound parameter γ are estimated in this regulator. It is noted that typical Lagrangian dynamics of robot manipulators can be expressed in this format.

First, we define the error vector e and the switching variable s .

$$e_x = x - x_d \quad (28)$$

$$s = C[x - x_d] = Ce_x \quad (29)$$

where x_d is the desired state trajectory and C is an arbitrary matrix of proper dimension to make the square matrix $C \cdot B$ non-singular and positive definite. C can also be viewed to specify the normal direction of the sliding hyperplane.

Theorem 2

The system in Eq.(23) with the following control law, system parameter adaptation law, and upper-bound estimate adaptation law and with a proper choice of C is asymptotically stable, that is, the error vector in Eq.(28) is asymptotic to 0.

Control Law :

$$\tau = -Fx - g(t, x)\hat{\sigma} - \left[\frac{\rho(t, x, \bar{\gamma}) + k(t, x)}{\|s\|} \right] s + (CB)^{-1}C\dot{x}_d \quad (30)$$

$$F = (C \cdot B)^{-1}CA(t) \quad (31)$$

where $C \cdot B$ is a positive definite matrix and $k(t, x)$ is positive for any non-zero x .

Parameter Adaptation Law :

$$\frac{d\hat{\sigma}}{dt} = W_1 \cdot g^T(t, x) \cdot s \quad (32)$$

where W_1 is a nonsingular weighting matrix of a

Fig. 1 Block diagram of the manipulator control system

proper dimension.

Upper-Bound Adaptation Law :

$$\frac{d\bar{\gamma}}{dt} = W_2 \cdot \frac{\partial \rho}{\partial \gamma} \Big|_{\bar{\gamma}} \cdot \|s\| \quad (33)$$

where W_2 is a nonsingular weighting matrix of a proper dimension.

Proof: The stability is proven by the Lyapunov stability theorem. Consider the following V -function candidate.

$$V = \frac{1}{2} [s^T (C \cdot B)^{-1} s + \bar{\sigma}^T W_1^{-1} \bar{\sigma} + \bar{\gamma}^T W_2^{-1} \bar{\gamma}] \quad (34)$$

Computing the time derivative of this function using Eq.(23) through Eq.(29) gives

$$\dot{V} \leq -k(t, x) \|s\| \leq 0 \quad (35)$$

for non-zero s . This shows that the system is asymptotic to the sliding hyperplane defined by $s=0$. The asymptoticity of e to 0 on the hyperplane depends on the choice of matrix C . Note that we can choose C in such a way that this asymptoticity is guaranteed. An example of such a C matrix is given in the next section.

Q.E.D.

The block diagram of the manipulator control system is shown in Fig. 1.

6. System Configuration

The experimental system consists of a PUMA-type robot manipulator, a servo-driver, a VME-bus based board computer, and a UNIX work-station. The robot manipulator is a two link type with link lengths of 301.5 mm (Link 1) and 227.0 mm (Link 2). Each link weighs 2.0 kg. The joints are activated by DC servomotors with incremental encoders (50,000 pulses per 360 degrees). The board computer consists of a CPU board with a Motorola 68040 CPU and a VxWorks real-time operating system, D/A converter board, and a counter board. It is connected to a network with multiple UNIX workstations, one of them is used as a host and data processing station. The schematic diagram is shown in Fig. 2.

The dynamic equation of the manipulator in the state space description requires a four-dimensional

Fig. 2 Schematic diagram of the robot system

state vector and is defined as,

$$x(t) = [\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2]^T \quad (36)$$

The C matrix to define the sliding hyperplane can be chosen as,

$$C = \begin{bmatrix} \lambda_1 & 0 & 1 & 0 \\ 0 & \lambda_2 & 0 & 1 \end{bmatrix} \quad (37)$$

with positive parameters λ_1 and λ_2 to insure the asymptoticity of e_x to zero. It is mentioned that this choice of the C matrix makes $C \cdot B$ positive definite.

7. Experiment-I, Simulation

Tracking performance of a regular robust controller (Slotine type) with moderate upper-bound estimates, termed as SMC⁽⁴⁾, the same controller with high upper-bound estimates, termed as SMCH, as well as the proposed controllers I, termed as ARSC-R, and II, termed as ARSC-S are simulated and the performances of these four controllers are compared. The difference between the two controllers, SMC and SMCH, lies in the feedback gain value. The feedback gain value is governed by the upper-bound estimates. The SMCH has a larger feedback gain value than that of the SMC. SMCH is expected to have better robustness than SMC, but would be more susceptible to chattering. In the experiment, we use the following set of values as the controller's design parameters.

For ARSC-R and ARSC-S

K_1	K_2	$\gamma(0)$
1.0	100.0	10.0

For ARSC-S

$C = \begin{bmatrix} 20 & 0 & 1 & 0 \\ 0 & 20 & 0 & 1 \end{bmatrix}$	W_1	W_2	k
	$I \times 9.0 \times 10^2$	2.0×10^{-3}	2.5×10^4

For ARSC-R

$\Gamma_1(1,1)$	$\Gamma_1(2,2)$	$\Gamma_1(3,3)$	$\Gamma_1(4,4)$	$\Gamma_1(5,5)$
5.0	7.0	10.0	10.0	10.0
$\Gamma_1(6,6)$	$\Gamma_1(7,7)$	$\Gamma_1(8,8)$	$\Gamma_1(9,9)$	Γ_2
1.0	1.0	0.1	0.1	0.002

Table 1 Tracking accuracy in simulation without disturbance

	Link1 Error	Link2 Error	Position Error
SMC	2.48 %	2.27 %	6.18 %
SMCH	2.70 %	2.98 %	5.39 %
ARSC-R	0.46 %	0.42 %	0.85 %
ARSC-S	1.17 %	1.33 %	2.11 %

Table 2 Tracking accuracy in simulation with disturbance

	Link1 Error	Link2 Error	Position Error
SMC	10.2 %	2.84 %	15.1 %
SMCH	2.96 %	2.50 %	5.49 %
ARSC-R	1.19 %	0.85 %	1.74 %
ARSC-S	2.10 %	1.38 %	3.49 %

These values are determined through test runs. For ARSC-R, we introduced artificial errors to initial values by multiplying them by 1.2 in the simulation. On the other hand, for the initial value of $\bar{\sigma}$ we use a nominal value of σ that is computed in Eq. (20) using system parameters. The trajectory to follow is chosen to be a circle of radius 150.0 mm centered at the middle of the robot's operational field. The robot accelerates from 0 to 2 revolutions per second speed in the first one second interval. Then the speed is kept constant for four seconds and decelerated to a stop in one second. The total simulation time is six seconds. The simulation is repeated twice, once with no external disturbances and once with a disturbance which is simulated by doubling the weight of Link 2 at the two second mark.

The accuracy of target tracking is measured by two kinds of errors, joint angle error rate and end-effector position error rate defined by

$$\epsilon_1 = \text{Average of } \left| \frac{\theta - \theta_d}{\theta_d} \right| \quad (38)$$

$$\epsilon_2 = \text{Average of } \frac{\sqrt{(x-x_d)^2 + (y-y_d)^2}}{r} \quad (39)$$

where r is the desired radius (=150.0 mm). The results are summarized in Table 1 for the no disturbance case and in Table 2 for the disturbance case.

The results show the obvious advantages of the proposed controllers, especially of ARSC-R, over conventional ones. Among the two proposed controllers, ARSC-R consistently outperformed ARSC-S. This endorsed the observation that, while ARSC-S would enjoy wider applicability, ARSC-R more fully takes advantage of the special structure of the Lagrangian dynamics of robot manipulators.

The tracking error performance of SMCH is

Fig. 3 Link 1 position error of simulation

clearly improved over SMC. It did not degrade with disturbance and demonstrated strong robustness. As expected, however, the SMCH controller exhibits significant chattering as is obvious in Fig. 3 where position errors of Link 1 with four controllers were plotted for the no disturbance case.

8. Experiment-II, Robot Implementation

With the simulation results in mind, we implemented and tested four types of controllers. They were a feed-forward controller, termed as FFC, a Slotine-type adaptive sliding mode controller, termed as SAC, an upper-bound adaptation controller, termed as UBAC, and the proposed ARSC-R.

SAC has a parameter adaptation part, but not the upper-bound adaptation part (it is the same as ARSC-R with $\Gamma_2=0$), while UBAC has the upper-bound estimator, but not the parameter identifier ($\Gamma_1=0$ in ARSC-R). We have chosen ARSC-R rather than ARSC-S to be implemented on our robot because the former consistently outperformed the latter in the simulation study. FFC (designed based on Eq. (1) with nominal system plant parameters) is added to the experiment to provide a frame of reference. In implementation, the sampling time of each controller is set at 2 [msec].

In the experiment, we use the following sets of values as the controller's design parameters.

λ	ϵ	K_1	K_2	$\gamma(0)$
110.0	0.005	$\lambda \times \epsilon \times 0.7$	$\lambda \times \epsilon \times 0.3$	4.0×10^{-1}

$\Gamma_1(1,1)$	$\Gamma_1(2,2)$	$\Gamma_1(3,3)$	$\Gamma_1(4,4)$	$\Gamma_1(5,5)$
1.2×10^{-4}	2.5×10^{-3}	0.4×10^{-4}	0.9×10^{-3}	0.9×10^{-4}
$\Gamma_1(6,6)$	$\Gamma_1(7,7)$	$\Gamma_1(8,8)$	$\Gamma_1(9,9)$	Γ_2
0.03×10^{-3}	0.03×10^{-3}	0.01×10^{-3}	0.01×10^{-3}	4.0×10^{-1}

$$\bar{a}(0) = [M_1 \ M_2 \ \beta \ D_1 \ D_2 \ E_1 \ E_2 \ G_1 \ G_2]^T$$

$M_1 \ (V \cdot s^2/rad)$	1.7×10^{-2}
$M_2 \ (V \cdot s^2/rad)$	8.7×10^{-3}
$\beta \ (V \cdot s^2/rad)$	2.9×10^{-3}
$D_1 \ (V \cdot s/rad)$	3.3
$D_2 \ (V \cdot s/rad)$	1.7
$E_1 \ (V)$	5.4×10^{-3}
$E_2 \ (V)$	3.8×10^{-3}
$G_1 \ (V)$	4.0×10^{-2}
$G_2 \ (V)$	3.0×10^{-3}

Before each trajectory tracking experiment, a set of preliminary measurements was taken to identify nominal values for system parameters that were used as initial values. The controllers were also tuned, namely design parameters such as λ and various weighting matrices were adjusted, through this phase.

As in the simulation, the task of those controllers is to follow a circular target trajectory of radius 85.0 mm centered at the middle of the robot's operational field. The experiments are performed with each controller under two conditions that are

Case 1: With no external disturbance.

Case 2: With a 2 kg load attached at the tip of Link 2.

Note that, the 2 kg load in case 2 corresponds to 100% weight disturbance for Link 2. The trajectory tracking errors of Link 1 with three controllers, SAC, UBAC, and ARSC-R, are plotted in Fig. 4 for case 1 and in Fig. 5 for case 2. The time histories of sliding variable $s(t)$ for Link 1, corresponding to Fig. 4, are plotted in Fig. 6. The sliding variable, the control, and the upper-bound estimate of the ARSC-R controller for Link 2 are plotted together in Fig. 7.

The performance of controllers are again compared using the two error rates defined in Eqs. (38) and (39). The results are summarized in Table 3 for case 1 and Table 4 for case 2. The results summarized in Table 3 and Fig. 4 clearly indicate the relative merit of the three controllers. The performance is best with ARSC-R then UBAC and finally SAC. Especially, the tracking error converges faster with ARSC-R as compared to the other two controllers. This result confirms the advantage of having both system parameter and upper-bound adaptation schemes in the controller.

It also shows that the performance of a sliding mode controller depends more heavily on the upper-bounds than the system parameters (UBAC outperformed SAC). However, this point must be studied more carefully because, although the model parameters were identified and controllers were tuned prior to the experiment, the relative "well-tunedness" among the three controllers was not evaluated. Therefore, it

Fig. 4 Link 1 tracking error(actual implementation)

Fig. 5 Link 1 tracking error(actual implementation)

Fig. 6 Sliding variable for link 1(actual implementation)

Fig. 7 Sliding variable, compensation input, and upper-bound(link 2)

Table 3 Tracking accuracy in actual implementation

	Link1 Error	Link2 Error	Position Error
FFC	1.74 %	7.56 %	21.3 %
SAC	1.29 %	0.050 %	1.23 %
UBAC	0.92 %	0.060 %	0.97 %
ARSC-R	0.86 %	0.045 %	0.81 %

Table 4 Tracking accuracy in actual implementation with load

	Link1 Error	Link2 Error	Position Error
SAC	1.71 %	0.055 %	1.27 %
UBAC	0.60 %	0.047 %	0.73 %
ARSC-R	0.59 %	0.045 %	0.71 %

Fig. 8 Sliding variable for link 1 in ARSC-R(actual implementation)

may be that the a priori parameter settings are not adequate for SAC to perform to its full potential. This result suggests the value of taking more care in estimating the upper-bounds of the system uncertainties.

Figures 6 and 7 indicate that the degree of chattering is low in these experiments. This suggests that the feedback gain used in SAC could have been increased to enhance robustness and that this might be the cause of the poorer performance of SAC compared to UBAC indicated in Table 3. Figure 7 shows the interdependence of the three plotted variables. The upper-bound estimate is increased as it should be whenever the sliding variable surges apart from zero. The control also takes high values accordingly at those moments to account for the increased separation from the sliding hyperplane.

Next, let us pay attention to Fig. 8 which shows the comparison of the sliding variables for case 1 and case 2. It can be observed that the ARSC-R is robust against disturbances caused by payloads.

9. Concluding Remarks

Two types of adaptive robust sliding mode

controllers are proposed and implemented on a robot manipulator. One of them, termed ARSC-R, is based on the Lagrangian dynamic model of manipulators and takes full advantage of its special features. The other one, termed ARSC-S, adopts a general state space description that is decomposed to a linear part, a nonlinear part with unknown parameters and an unknown nonlinear disturbance. The former is aimed at application to robot manipulators while the latter is aimed at wider applicability. Both of them have upper-bound adaptation as well as system parameter adaptation to limit chattering without jeopardizing robustness.

The results of simulation and actual implementation both showed improved performance of the proposed controllers over conventional sliding mode controllers.

Acknowledgment

It is acknowledged that this research was partially supported by the SONY Corporation and the

Research Institute of Aoyama Gakuin.

References

- (1) Utkin, V.I., Variable Structure System with Sliding Modes, IEEE Tran. Auto. Con., Vol. AC-22, No. 2(1977), p. 212.
- (2) Furuta, K., VSS-type Self-Tuning Control, IEEE Tran. on Ind. Elect., Vol. IE-40, No. 1(1993), p. 37.
- (3) Chen, Y.H., Design of Robust Controller for Uncertain Dynamical System, IEEE Tran. on Auto. Con., Vol. AC-33, No. 109(1988), p. 209.
- (4) Slotine, J.J. and Li, W., On the Adaptive Control of Robot Manipulators, Int. J. Robotic Res., Vol. 6, No. 3(1987), p. 49.
- (5) Chen, Y.F. and Mita, T., Adaptive Robust Sliding Mode Control, T. IEE Japan, (in Japanese), Vol. 113-C, No. 3(1993), p. 203.
- (6) Chen, Y.F. and Mita, T., Sliding Mode Controller with Adaptive VSS Observer, T. IEE Japan, (in Japanese), Vol. 111-C, No. 10(1991), p. 514.
- (7) Slotine, J.J. and Li, W., Adaptive Manipulator Control: A Case Study, IEEE Tran. Auto. Con., Vol. AC-33, No. 11(1988), p. 995.