Proceedings of the 2006 IEEE International Conference on Robotics and Automation
Orlando, Florida - May 2006

Proxy-Based Sliding Mode Control For Accurate and Safe Position Control

Ryo Kikuuwe and Hideo Fujimoto

Nagoya Institute of Technology, Nagoya, Aichi 466-8555, Japan
kikuuwe@ieee.org

***Abstract*— High-gain PID position control, which is widely used with industrial robots, involves some risks in cases of abnormal events, such as unexpected environment contacts and temporal power failures. This paper proposes a new position control method to achieve both accurate, responsive tracking during normal operation and smooth, overdamped recovery from a large positional error after abnormal events. The proposed method, which we call proxy-based sliding mode control, is a modified version of sliding mode control adapted to discrete-time systems, and also is an extension of PID control. The validity of the proposed method is demonstrated through experiments.**

I. Introduction

Industrial robots contain many components whose dynamics are difficult to be modeled, such as joint frictions. To suppress the influence of unmodeled dynamics, industrial robots usually require a stiff position controller as the lowest level controller. Low-level position controllers are necessary even for implicit force control [1] and admittance control [2]. One of the most widely used methods for position control is PID (proportional-integral-derivative) control.

Stiff position control can be unsafe when the desired position determined by a higher-level controller is far separated from the actual end-effector position. This unsafe situation occurs in cases of unexpected environment contacts, temporal power failures to the actuators, software failures in the higher-level controller, and so on. In such situations, the actuator force, determined by the position controller, can become excessively large. This can cause damage to the environment and the robot, and can cause excessive speed and overshoots during recovery to the desired position.

It is not a straightforward problem with PID control method to ensure both the accuracy during normal operation and the safety in cases of abnormal events. For safety in cases of abnormal events, the magnitude of the actuator force should be limited. Besides, recovering motions from abnormal positional errors should be ensured to be adequately slow, moderate, and overdamped to prevent excessive speed and overshoots. One imaginable way to achieve overdamped dynamics is to use a very high velocity feedback gain. However, this can deteriorate the responsiveness and accuracy during normal operation. In addition, a high velocity feedback gain can magnify the measurement noise in the velocity signal especially in low-velocity motion. Therefore, both accuracy and safety cannot be achieved by conventional PID control with a force limiter.

The concept of sliding mode control provides an approach to avoid the above problem. Sliding mode control is a control method to realize the desired dynamics defined by a manifold (called a sliding surface) in a state space. The overdamped dynamics, which is necessary for a safe recovering motion, can be represented as a sliding surface. The actuator force is determined to attract the system state toward the sliding surface, and the system state can be robustly constrained to the surface by infinitely fast switching of the actuator force. In addition, the magnitude limit of the actuator force can be explicitly set as a user parameter for the controller. Therefore, sliding mode control is, in principle, capable of achieving both accuracy and safety. However, this idealized concept of sliding mode control is based on the unrealistic assumption that there are no time delays in the feedback loop. When the sliding mode control is directly implemented to a real control system with a discrete-time controller, repetition of delayed switching on the sliding surface causes high-frequency oscillation, which is called chattering.

In this paper, we propose a *proxy-based sliding mode control*, which is a modified version of sliding mode control adapted to discrete time and, at the same time, is an extension of force-limited PID control. This method can achieve smooth, overdamped recovering motion after abnormal events without sacrificing accurate, responsive tracking capability during normal operations.

The rest of this paper is organized as follows. Section II discusses the capability of ideal sliding mode control to achieve both accuracy and safety under the assumption of infinitely fast response. Section III presents a new proxy-based sliding mode control to preserve advantages of ideal sliding mode control in discrete-time systems. Section IV presents experimental results. Section V provides the concluding remarks.

II. Sliding Mode Control in Continuous Time

We start our discussion from a 1-DOF continuous-time system to clarify the capability of ideal sliding mode control to achieve both accuracy and safety. Let M denote the inertia of a controlled object, and p and v denote the object's position and velocity, respectively. Let p_d and v_d denote the desired position and velocity, respectively. The controller accepts p, v, p_d, and v_d as inputs, and produces the actuator force f_a as an output. The force f_a from the actuator and an external force h are applied to the controlled object. Then, the equation of motion of the controlled object is written as $M\dot{v} = f_a + h$.

One of the simplest examples of the sliding mode control

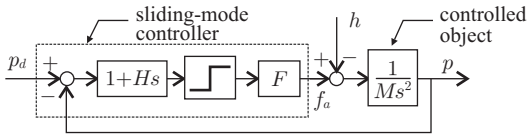

Fig. 1. A simple example of sliding mode control system.

scheme determines the actuator force f_a as follows:

$$f_a = F\mathrm{sgn}\,(s)\,, \tag{1a}$$

where

$$s = p_d - p + H(v_d - v). \tag{1b}$$

Here, F and H are positive real numbers, and $\mathrm{sgn}(\cdot)$ is the signum function, which is defined by

$$\mathrm{sgn}(x) = \begin{cases} 1 & \text{if } x > 0 \\ -1 & \text{if } x < 0. \end{cases} \tag{2}$$

Fig. 1 shows the block diagram of this system.

The condition $s = 0$, i.e.,

$$p_d - p + H(v_d - v) = 0 \tag{3}$$

defines a manifold (in this case, a line) in the state space (in this case, the p-v plane). This manifold is called the sliding surface, and the controller output f_a is switched on this surface, as indicated in (1a). If this switching is infinitely fast and the parameter F is sufficiently large, the system state $\{p, v\}$ reaches the sliding surface in finite time. After reaching the surface, the system state is constrained to the sliding surface as long as the external force h is sufficiently small. During the system state is constrained to the surface, the system state develops according to the differential equation $s = 0$, that is,

$$\dot{p} = v = (p_d - p)/H + v_d. \tag{4}$$

As long as the system obeys (4), the system exhibits an overdamped convergence to the desired position and velocity. This system is said to be in the sliding mode when $s = 0$.

In the control law (1), F is the magnitude limit of the actuator force, and H can be interpreted as the time constant of the recovering motion. These values can be determined by users to ensure safety at or after abnormal events. The tracking during normal operation is perfect (when F is sufficiently large and h is sufficiently small) under the assumption of infinitely fast switching. Thus, we can see that the idealized concept of sliding mode control meets our purpose in principle. However, it cannot be directly used in real control systems because time delays in switching are inevitable.

Time delays in switching cause chattering. One of the common methods to remove chattering is to approximate the discontinuous signum function by a continuous function [3], [4]. Typical example is to use

$$f_a = \begin{cases} Fs/C & \text{if } |s| \le C \\ F\mathrm{sgn}(s) & \text{if } |s| > C \end{cases} \tag{5}$$

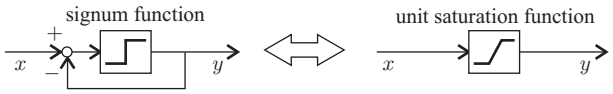

Fig. 2. Two equivalent block diagrams including the signum function and the unit saturation function.

instead of (1a). The parameter C is a positive real number that determines the width of a boundary layer which is laid around the sliding surface. The drawbacks of this approach will be discussed in section III-E.

III. Proxy-Based Sliding Mode Control

The difficulty in realizing the ideal sliding mode control in discrete time is rooted in the discontinuity in the signum function. In this section, we present a method to remove this difficulty and propose a modified version of sliding mode control, which we call proxy-based sliding mode control. This new control scheme is an extension of the conventional PID control scheme to ensure an overdamped recovery motion from a large positional error without sacrificing tracking accuracy.

A. Signum Function and Unit Saturation Function

First, we prepare a mathematical tool to deal with the discontinuity of the signum function. In stead of using (2), we use the following definition of the signum function:

$$\mathrm{sgn}(x)\begin{cases} = 1 & \text{if } x > 0 \\ \in [-1, 1] & \text{if } x = 0 \\ = -1 & \text{if } x < 0. \end{cases} \tag{6}$$

In this definition, $\mathrm{sgn}(x)$ can take an arbitrary value in $[-1, 1]$ when $x = 0$. This definition has been sometimes used in the literature [5]. With this definition, we have the following theorem:

Theorem 1. *With two real numbers x and y, the following statement holds true:*

$$y = \mathrm{sgn}(x - y) \iff y = \mathrm{sat}(x), \tag{7}$$

where $\mathrm{sat}(\cdot)$ is the unit saturation function, which is defined as

$$\mathrm{sat}(x) = \begin{cases} x & \textit{if } |x| \le 1 \\ \mathrm{sgn}(x) & \textit{if } |x| > 1. \end{cases} \tag{8}$$

Proof. The statement $y = \mathrm{sgn}(x - y)$ is equivalent to

$$\begin{aligned} &(y = 1 \ \wedge \ x - y > 0) \\ &\vee (y \in [-1, 1] \ \wedge \ x - y = 0) \\ &\vee (y = -1 \ \wedge \ x - y < 0)\,. \end{aligned} \tag{9}$$

Each term in (9) can be rewritten as follows:

$$\begin{aligned} (y = 1 \ \wedge \ x - y > 0) &\iff (y = 1 \ \wedge \ x > 1) \\ (y \in [-1, 1] \ \wedge \ x - y = 0) &\iff (y = x \ \wedge \ x \in [-1, 1]) \\ (y = -1 \ \wedge \ x - y < 0) &\iff (y = -1 \ \wedge \ x < -1)\,. \end{aligned}$$

Therefore, (9) is equivalent to $y = \mathrm{sat}(x)$. This means that $y = \mathrm{sgn}(x - y)$ is equivalent to $y = \mathrm{sat}(x)$. □

Theorem 1 can be illustrated as a pair of two equivalent block diagrams in Fig. 2. This means that if the discontinuous function $\mathrm{sgn}(\cdot)$ is enclosed within a closed loop without time delay, it can be converted into a system including the unit saturation function $\mathrm{sat}(\cdot)$, which is continuous.

B. Introducing a Proxy

If the feedback loop surrounding the sgn element is passing through actual physical devices such as sensors and actuators, as shown in Fig. 1, time delays are inevitable. To remove this time delay, we must make a closed loop surrounding the sgn element within the controller software. For this purpose, we consider an imaginary *proxy* object in the software.

A proxy, also referred to as a god-object [6], [7], is a concept that is frequently used in the field of haptic display. It is a point-like object that represents the position of the end-effector of a haptic device in a simulated virtual world. The proxy in the virtual world and the actual end-effector are connected to each other by a virtual coupling [8], which is an imaginary spring-like element that exerts a force to maintain its length to be zero. Due to its effect, the actual position of the end-effector is servo-controlled to follow the proxy position. The advantage of a proxy is its capability of simulating ideal motion that can satisfy idealized physical constraints. In the case of this paper, the proxy is used to simulate an object controlled with an ideal sliding mode controller, which does not suffer from chattering.

Fig 3 shows a system including a proxy whose position is controlled using an ideal sliding mode controller. The proxy is connected to the actual controlled object via a virtual coupling. In the field of haptic display, a virtual coupling is usually considered as a spring-dashpot system [8], which exhibits PD-type control action on its length. In order to maintain extensibility of this discussion, we here consider to use a more general case, a PID-type virtual coupling, though the integral action has no physical analogy.

The block diagram of this system is shown in Fig. 4. Here, p_d, p_s, and p are the desired position, the proxy position, and the actual position of the controlled object, respectively. The mass of the proxy is denoted by m. The block $L/s+K+Bs$ represents the virtual coupling, where L, K, and B are the integral, proportional, and derivative gains, respectively. The block diagram Fig. 4 clearly shows that the sgn element is enclosed in a closed loop within the controller software.

We move on to a mathematical description of the system of Fig. 3 and Fig. 4. Let us denote the time derivatives of p_d, p_s, and p by v_d, v_s, and v, respectively. Then, the sliding mode controller determines the force f_a, which is applied to the proxy, as follows:

$$f_a = F\mathrm{sgn}\left(p_d - p_s + H(v_d - v_s)\right). \tag{10}$$

On the other hand, the force produced by the PID-type virtual coupling is given as follows:

$$f_c = L\int(p_s - p)dt + K(p_s - p) + B(v_s - v). \tag{11}$$

The force f_c is applied to the real controlled object (through the real actuator), and its reaction force is applied to the proxy.

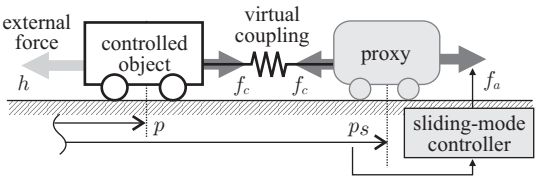

Fig. 3. Physical interpretation of a feedback system including a proxy.

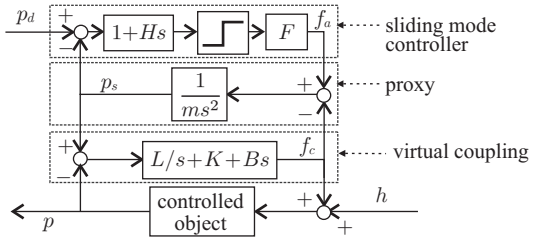

Fig. 4. Block diagram of a feedback system including a proxy.

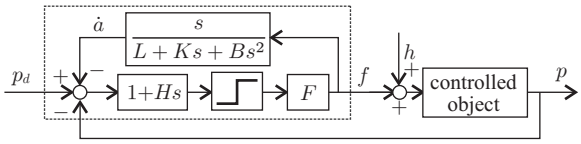

Fig. 5. Block diagram of proxy-based sliding mode control. This is equivalent to Fig. 4 if $m = 0$.

Thus, the equation of motion of the proxy can be written as

$$m\ddot{p}_s = f_a - f_c. \tag{12}$$

Here, we must notice that the proxy does not need to have a mass. Substituting equations (10)(11)(12) by $m = 0$ yields

$$f_a = F\mathrm{sgn}\left(p_d - p - \dot{a} + H(v_d - v - \ddot{a})\right) \tag{13}$$
$$f_c = La + K\dot{a} + B\ddot{a} \tag{14}$$
$$0 = f_a - f_c, \tag{15}$$

where

$$a = \int(p_s - p)dt. \tag{16}$$

By replacing both f_a and f_c by f (because $f_a = f_c$), we have a simpler expression:

$$\sigma = p_d - p + H(\dot{p}_d - \dot{p}) \tag{17a}$$
$$f = F\mathrm{sgn}\left(\sigma - \dot{a} - H\ddot{a}\right) \tag{17b}$$
$$f = La + K\dot{a} + B\ddot{a}. \tag{17c}$$

Fig. 5 shows the block diagram of this system. Notice that f is determined so as to satisfy both of (17b) and (17c). We can interpret a as an internal state variable.

C. Discrete-Time Representation

We consider to derive a discrete-time control law that satisfies (17). A discrete-time representation of (17) based on the Euler approximation is given as follows:

$$\sigma(k) = p_d(k) - p(k) + H(v_d(k) - v(k)) \tag{18a}$$
$$f(k) = F\mathrm{sgn}\left(\sigma(k) - \nabla a(k)/T - H\nabla^2 a(k)/T^2\right) \tag{18b}$$
$$f(k) = La(k) + K\nabla a(k)/T + B\nabla^2 a(k)/T^2 \tag{18c}$$

Here, the arguments in parentheses, such as k, are integers indicating discrete time indices. The operator ∇ is the backward difference operator, which is defined as $\nabla x(k) = x(k) - x(k-1)$, and satisfies $\nabla^2 x(k) = \nabla x(k) - \nabla x(k-1) = x(k) - 2x(k-1) + x(k-2)$. The timestep size is denoted by T.

Note that the expression (18) is not considered as a computational procedure, but as a set of algebraic constraints. The unknowns in (18) are $f(k)$ and $a(k)$. To determine these values, we have to algebraically solve (18b) and (18c). First, (18c) can be rewritten as

$$a(k) = \frac{(B+KT)a(k-1) + B\nabla a(k-1) + T^2 f(k)}{B + KT + LT^2}. \quad (19)$$

Substituting (18b) by (19) yields

$$\begin{aligned} & f(k) = F\mathrm{sgn}\left(\frac{H+T}{B+KT+LT^2}(f^*(k) - f(k))\right) \\ \iff & f(k) = F\mathrm{sgn}\left(f^*(k) - f(k)\right), \end{aligned} \quad (20)$$

where

$$\begin{aligned} f^*(k) = & \frac{B+KT+LT^2}{H+T}\sigma(k) + La(k-1) \\ & + \frac{(K+LT)H - B}{(H+T)T}\nabla a(k-1). \end{aligned} \quad (21)$$

Because of Theorem 1, (20) is equivalent to

$$\begin{aligned} f(k) &= F\mathrm{sat}\left(f^*(k)/F\right) \\ &= \begin{cases} f^*(k) & \text{if } |f^*(k)| \le F \\ F\mathrm{sgn}(f^*(k)) & \text{if } |f^*(k)| > F. \end{cases} \end{aligned} \quad (22)$$

Thus, $f(k)$ is obtained as (22). The other unknown $a(k)$ can be calculated by substituting (19) by (22). In conclusion, the algebraic constraint (18) can be solved by using the following computational procedure:

$$\sigma(k) := p_d(k) - p(k) + H(v_d(k) - v(k)) \quad (23a)$$

$$\begin{aligned} f^*(k) := & \frac{B+KT+LT^2}{H+T}\sigma(k) + La(k-1) \\ & + \frac{(K+LT)H - B}{(H+T)T}\nabla a(k-1) \end{aligned} \quad (23b)$$

$$f(k) := \begin{cases} f^*(k) & \text{if } |f^*(k)| \le F \\ F\mathrm{sgn}(f^*(k)) & \text{if } |f^*(k)| > F \end{cases} \quad (23c)$$

$$a(k) := \frac{(B+KT)a(k-1) + B\nabla a(k-1) + T^2 f(k)}{B + KT + LT^2}. \quad (23d)$$

The proposed control law (23) of proxy-based sliding mode control drives the proxy state $\{p_s(k), v_s(k)\}$ to reach the sliding surface in finite time. Let us define

$$s(k) = p_d(k) - p_s(k) + H(v_d(k) - v_s(k)). \quad (24)$$

The proxy state is said to be in the sliding mode if and only if $s(k) = 0$. We can rewrite $s(k)$ as follows:

$$\begin{aligned} s(k) &= \sigma(k) - \nabla a(k)/T - H\nabla^2 a(k)/T^2 \\ &= \frac{H+T}{B+KT+LT^2}(f^*(k) - f(k)) \end{aligned} \quad (25)$$

Here, $f^*(k)$ is dependent on the current inputs ($p(k)$, $v(k)$, $p_d(k)$, and $v_d(k)$) and previous state variables ($a(k-1)$ and $a(k-2)$), as indicated by (21). Equation (22) determines the output force $f(k)$ so as to minimizes $|s(k)|$ subject to (25) and $|f(k)| \le F$. This $f(k)$ does not cause a signum change in $s(k)$ and makes the proxy state arrive in the sliding mode ($s(k) = 0$) when $|f^*(k)| \le F$ is satisfied. This means that the output force $f(k)$ does not causes chattering in the proxy position. The end-effector position, which is servo-controlled to follow the proxy position, does not suffer from chattering as long as the PID control gains, K, B, and L, are appropriately chosen.

D. Relation with PID Control Laws

Again, the proposed control law (23) is the solution for the algebraic constraints (18). Setting $H = 0$ and $F \to \infty$ with (18) yields

$$\sigma(k) = p_d(k) - p(k) \quad (26a)$$

$$0 = \sigma(k) - \nabla a(k)/T \quad (26b)$$

$$f(k) = La(k) + K\nabla a(k)/T + B\nabla^2 a(k)/T^2, \quad (26c)$$

which is equivalent to the conventional PID control law. (Note that in (18b), $|f(k)| < F$ implies $\sigma(k) - \nabla a(k)/T - H\nabla^2 a(k)/T^2 = 0$ from the definition of $\mathrm{sgn}(\cdot)$ in (6).) This means that the proposed control law (23) is an extension of PID control.

By setting $L = 0$ and $H = B/K$ in (23), we have

$$f^*(k) := K(p_d(k) - p(k)) + B(v_d(k) - v(k)) \quad (27a)$$

$$f(k) := \begin{cases} f^*(k) & \text{if } |f^*(k)| \le F \\ F\mathrm{sgn}(f^*(k)) & \text{if } |f^*(k)| > F, \end{cases} \quad (27b)$$

or equivalently,

$$\sigma(k) := p_d(k) - p(k) + H(v_d(k) - v(k)) \quad (28a)$$

$$f(k) := \begin{cases} K\sigma(k) & \text{if } |\sigma(k)| \le F/K \\ F\mathrm{sgn}(\sigma(k)) & \text{if } |\sigma(k)| > F/K. \end{cases} \quad (28b)$$

Equation (27) can be seen as force-limited PD control, while (28) can be seen as sliding mode control with a boundary layer [3], [4]. The equivalency of these two methods has been pointed out in the literature [9]. The proposed method is an extension of these conventional methods.

E. Choice of Parameter Values

The advantage of proxy-based sliding mode control (23) over force-limited PD control (27) and sliding mode control with a boundary layer (28) lies in the separation of "global" dynamics from "local" dynamics. The local dynamics can be defined as the response characteristics to small positional errors between the actual and desired positions. Small positional errors always occur in normal operation of trajectory tracking. Therefore, the local dynamics should be set as stiff as and as responsive as possible to maintain tracking accuracy. The local dynamics is determined by the parameters of the virtual coupling: K, B, and L.

The global dynamics, on the other hand, can be defined as the response characteristics to large positional errors. Such

positional errors can be produced in abnormal events, such as environment contacts, power failures, and software failures in a higher-level controller that determines the desired position. The global dynamics is determined by H. It should be set so as to achieve adequately smooth and slow recovering motion without overshoots.

In order to maintain the tracking accuracy during normal operation and the safety during abnormal operation, it is desirable to set $H \gg B/K$. (Remember that setting $H = B/K$ and $L = 0$ makes the proposed method (23) equivalent to conventional methods, (27) and (28).) The proportional gain K should be set as high as possible to maintain the tracking accuracy. The derivative gain B should be chosen as small as possible but large enough to suppress oscillation. A large B value will deteriorate responsiveness and tracking accuracy, and can magnify undesirable influence of the noise in the velocity signal. The H value should be large enough to achieve smooth and slow recovering motion. A small H value can cause excessive speed and overshoots.

The integral gain L should be chosen high enough to eliminate steady state errors due to the gravitational force and friction forces during normal operation, but not so high as to cause overshoots. The torque limit F should be chosen so as to overcome the joint friction and link inertia, but so as not to damage environment and human operators.

F. Extensions to Multidimensional Space

Proxy-based sliding mode control, (23), can easily be extended into multidimensional cases. A vectorial version of (23) can be written as follows:

$$\boldsymbol{s}(k) := \boldsymbol{p}_d(k) - \boldsymbol{p}(k) + H(\boldsymbol{v}_d(k) - \boldsymbol{v}(k)) \tag{29a}$$

$$\boldsymbol{f}^*(k) := \frac{B + KT + LT^2}{H + T}\boldsymbol{s}(k) + L\boldsymbol{a}(k-1) + \frac{(K + LT)H - B}{(H + T)T}\nabla\boldsymbol{a}(k-1) \tag{29b}$$

$$\boldsymbol{f}(k) := \begin{cases} \boldsymbol{f}^*(k) & \text{if } \|\boldsymbol{f}^*(k)\| \le F \\ F\boldsymbol{f}^*(k)/\|\boldsymbol{f}^*(k)\| & \text{if } \|\boldsymbol{f}^*(k)\| > F \end{cases} \tag{29c}$$

$$\boldsymbol{a}(k) := \frac{(B + KT)\boldsymbol{a}(k-1) + B\nabla\boldsymbol{a}(k-1) + T^2\boldsymbol{f}(k)}{B + KT + LT^2}. \tag{29d}$$

Here, boldface symbols indicate vectors correspondent to scalars in (23). In this case, the force limit is specified in terms of the magnitude of the force vector.

If the force limit should be specified in terms of the torque of each actuator in a multi-linkage mechanism, (29) should be slightly modified. In (29), assume that the vectors are represented in the Cartesian space. Let $\boldsymbol{J}$ be the Jacobean matrix to transform the joint angular velocity to the end-effector velocity in the Cartesian space. Then, the joint actuator torque statically equivalent to $\boldsymbol{f}$ is described as $\boldsymbol{\tau} = \boldsymbol{J}^T\boldsymbol{f}$. By using this, we can replace (29c) by

$$\boldsymbol{\tau}^*(k) := \boldsymbol{J}^T\boldsymbol{f}^*(k) \tag{30a}$$

$$\boldsymbol{\tau}(k) := \begin{cases} \boldsymbol{\tau}^*(k) & \text{if } \|\boldsymbol{\tau}^*(k)\|_\infty \le F_\tau \\ F_\tau\boldsymbol{\tau}^*(k)/\|\boldsymbol{\tau}^*(k)\|_\infty & \text{if } \|\boldsymbol{\tau}^*(k)\|_\infty > F_\tau \end{cases} \tag{30b}$$

$$\boldsymbol{f}(k) := \boldsymbol{J}^{-T}\boldsymbol{\tau}(k), \tag{30c}$$

where F_τ is the positive real number that indicate the limit of the actuator torque, and $\|\boldsymbol{x}\|_\infty$ denotes the L-infinity norm, which returns $\max_i |x_i|$, where x_i is the i-th element of $\boldsymbol{x}$.

More general description to impose force limits can be formally described as follows:

$$\boldsymbol{f}(k) = \begin{cases} \boldsymbol{f}^*(k) & \text{if } \boldsymbol{f}^*(k) \in \mathcal{F} \\ \left(\max_{\kappa \in [0,1] \wedge \kappa\boldsymbol{f}^*(k) \in \mathcal{F}} \kappa\right)\boldsymbol{f}^*(k) & \text{if } \boldsymbol{f}^*(k) \notin \mathcal{F}. \end{cases} \tag{31}$$

Here, $\mathcal{F}$ is the set of allowable forces. Equation (29c) can be obtained by specializing (31) by $\mathcal{F} = \{\boldsymbol{f} \mid \|\boldsymbol{f}\| \le F\}$. Equation (30) is obtained by specializing (31) by $\mathcal{F} = \{\boldsymbol{f} \mid \|\boldsymbol{J}^T\boldsymbol{f}\|_\infty \le F_\tau\}$. The choice of the allowable force set $\mathcal{F}$ can be made according to the specification of the hardware or to safety regulations at a user's environment.

IV. Experiments

The proposed method was experimentally tested by using the 2-DOF planar parallel link manipulator shown in Fig. 6. This manipulator had two actuators, which were AC servo motors integrated with harmonic drive gearings. Each actuator had optical encoders, with which the position of the end-effector was measured. The joint friction forces in each joint were about 10 N·m at the output axes.

The manipulator were controlled using the control law (29). The position $\boldsymbol{p}$ of the end-effector in the Cartesian coordinate system was measured by the optical encoders, and the velocity $\boldsymbol{v}$ was calculated from the time difference of $\boldsymbol{p}$. The output of the control law, $\boldsymbol{f}$, was the force vector that should be exerted at the end-effector. Its statically equivalent joint torque was commanded to the actuators.

Table I shows the parameter settings used for the control law (29). The settings A and B satisfy $H = B/K$, with which the control law (29) becomes equivalent to a force-limited PD control law (and equivalently, a sliding mode control law with a boundary layer). The settings C and D are those suggested by the discussion in section III-E, satisfying $H \gg B/K$. No attempts were made to include the original sliding mode control law (without boundary layer) into the comparison because it is well known to be unpractical due to its tendency to exhibit chattering. The timestep size (sampling interval) was $T = 0.001$ sec.

The desired trajectory given to the controller was as follows:

$$\boldsymbol{p}_d(t) = \begin{cases} -[A_x/2, A_y]^T & \text{if } t < -\Lambda/8 \\ [A_x/2, A_y]^T \sin(4\pi t/\Lambda) & \text{if } -\Lambda/8 \le t < 0 \\ [A_x \sin(2\pi t/\Lambda), A_y \sin(4\pi t/\Lambda)]^T & \text{if } 0 \le t < \Lambda \\ [A_x/2, A_y]^T \sin(4\pi(t-\Lambda)/\Lambda) & \text{if } \Lambda \le t < 9\Lambda/8 \\ [A_x/2, A_y]^T & \text{if } 9\Lambda/8 \le t, \end{cases}$$

where $A_x = 0.25$ m, $A_y = 0.15$ m, and $\Lambda = 3$ sec. This trajectory was a figure-eight-like motion with smooth acceleration and deceleration periods. The desired velocity $\boldsymbol{v}_d$ was calculated by the time difference of $\boldsymbol{p}_d$.

To observe the recovering motion from a large positional error, we set the actuator torque to be zero between time $t = 0.3\Lambda = 1.8$ sec and $t = 0.7\Lambda = 4.2$ sec.

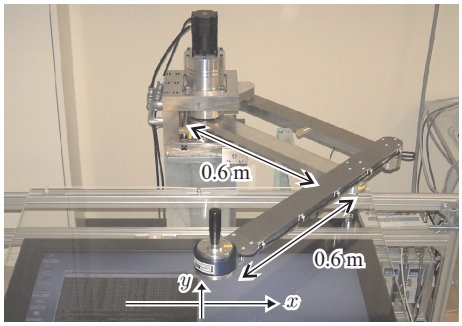

Fig. 6. Experimental setup: a parallel-link manipulator.

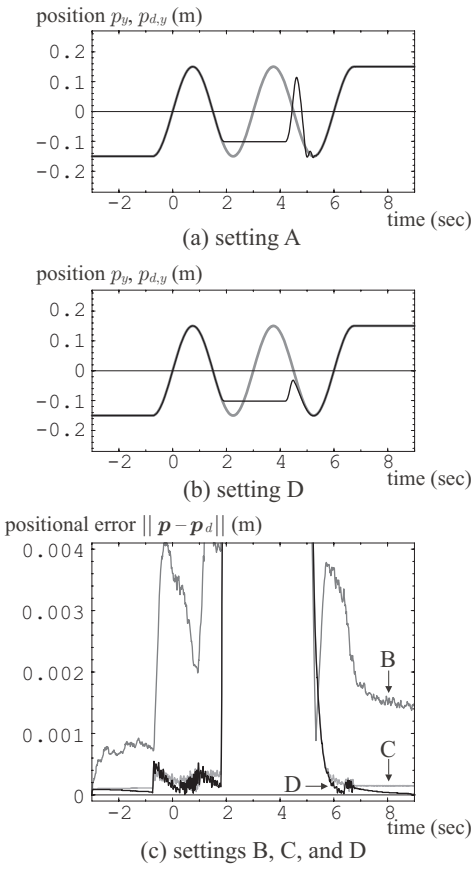

Fig. 7. Experimental results.

Fig. 7 shows the results. In Fig. 7(a) and (b), the gray curves and the black curves represent the y component of the desired position $\boldsymbol{p}_d$ and that of the end-effector position $\boldsymbol{p}$, respectively. Fig. 7(c) shows the magnitude of the positional error $\|\boldsymbol{p}-\boldsymbol{p}_d\|$. Because the results of settings B, C, and D appear almost the same in the scale of Fig. 7(a) and (b), those of the settings B and C are not shown in that scale.

The setting A is intended to achieve tracking accuracy with force-limited PD control. Thus, Fig. 7(a) shows a large overshoots and a subsequent oscillation after the power recovery ($t = 4.2$ sec). The setting B is intended to achieve smoothness with force-limited PD control. With this setting, the manipulator exhibited smooth, overdamped recovering motion after the power recovery, which is very similar to that in Fig. 7(b). However, its tracking accuracy was poor, as shown in Fig. 7(c). Even worse, with the setting B, the actuators created undesirable noisy sound presumably due to the magnified noise in the velocity signal. Fig. 7(b) and Fig. 7(c) show the capability of the settings C and D of achieving both accurate tracking and safe recovery. Especially, the integral term of the setting D contributes to decreasing steady state error in the resting state (after $t = 7$ sec).

TABLE I

PARAMETER SETTINGS A, B, C, AND D.

parameter (unit)	A	B	C	D
F (N)	40	40	40	40
L (N/ms)	0	0	0	100000
K (N/m)	80000	80000	80000	80000
B (Ns/m)	300	16000	300	300
H (s)	0.00375	0.2	0.2	0.2

V. CONCLUSIONS

This paper has proposed proxy-based sliding mode control to be used for position control of industrial robots. The proposed method is an extension of PID control scheme to achieve safer overdamped recovery from large positional errors without sacrificing tracking accuracy during normal operation.

We expect that the treatment of the discontinuous switching in the presented method can have some insights to deal with more general VSSs (variable structure systems), including nonlinear switching manifold, in discrete-time systems. Application of proxy-based methods to general VSSs will be the subject of future studies.

REFERENCES

[1] P. Rocco, G. Ferretti, and G. Magnani, "Implicit force control for industrial robots in contact with stiff surfaces," *Automatica*, vol. 33, no. 11, pp. 2041–2047, 1997.

[2] R. Q. Van Der Linde, P. Lammertse, E. Frederiksen, and B. Ruiter, "The HapticMaster, a new high-performance haptic interface," in *Proc. of The 2nd European Conf. on Haptics*, 2002, pp. 1–5.

[3] J.-J. E. Slotine and M. W. Spong, "Robust robot control with bounded input torques," *J. of Robotic Systems*, vol. 2, no. 4, pp. 329–352, 1985.

[4] K. S. Yeung and Y. P. Chen, "A new controller design for manipulators using the theory of variable structure systems," *IEEE Trans. on Automatic Control*, vol. 33, no. 2, pp. 200–206, 1988.

[5] Y. J. Lootsma, A. J. van der Schaft, and M. K. Çamlıbel, "Uniqueness of solutions of linear relay systems," *Automatica*, vol. 35, pp. 467–478, 1999.

[6] P. Dworkin and D. Zeltzer, "A new model for efficient dynamic simulation," *Proc. of the 4th Eurographics Workshop on Animation and Simulation*, pp. 135–147, 1993.

[7] C. B. Zilles and J. K. Salisbury, "A constraint-based god-object method for haptic display," in *Proceedings of the 1995 IEEE/RSJ International Conf. on Intelligent Robots and Systems*, 1995, pp. 3146–3151.

[8] J. E. Colgate, M. C. Stanley, and J. M. Brown, "Issues in the haptic display of tool use," in *Proc. of the 1995 IEEE/RSJ International Conf. on Intelligent Robots and Systems*, vol. 3, 1995, pp. 140–145.

[9] S.-T. Wu, "On digital high-gain and sliding-mode control," *Int. J. of Control*, vol. 66, no. 1, pp. 65–84, 1997.