

Intelligent Robot Deburring Using Adaptive Fuzzy Hybrid Position/Force Control

Feng-Yi Hsu and Li-Chen Fu, *Member, IEEE*

Abstract—The overwhelming complexity of the deburring process and imprecise knowledge about robot manipulators leads to a certain control problem. In this paper, a new design of hybrid position/force control of robot manipulators via adaptive fuzzy control approach is proposed to solve these problems.

The control architecture consists of an outer-loop command generator which can automatically determine the robot motion profile to yield the desired chamfering force and an inner-loop adaptive fuzzy hybrid position/force controller which can achieve the desired chamfer depth compliantly as well as the aforementioned command in real time. The proposed adaptive fuzzy controller using B-spline type membership functions can compensate the uncertainties in a much smoother and locally weighted manner and consequently guarantee global stability of closed-loop systems. To demonstrate the effectiveness of the developed work, it is applied to the control of an industrial robot arm for deburring tasks.

Index Terms—Fuzzy control, hybrid position/force control, robot deburring.

I. INTRODUCTION

DEVELOPING automated robot manipulators for deburring is becoming more important because of the high cost of manual deburring, which for some parts makes up 35% of total costs. [1]. In general, deburring tasks are done to remove burrs from the edges and to maintain the final geometry of the deburred part's edges within some allowable excess. To achieve this goal, a cutting tool with a fixed spindle speed has to chamfer the part edges while undergoing contour-following motion. When driving the cutting tool to perform a deburring task, the deburring robot must implement two major motions. One motion applies a suitable chamfering force to the part's edge to remove burrs while avoiding damage to the part. The other motion performs contour following. This ensures that the cutting tool will remain in contact with all the burrs over the chamfer. A heuristic deburring control policy can attain the desired chamfering force by controlling the feedrate of the cutting tool. This heuristic policy simply consists of slowing down the feedrate of the cutting tool when the encountered burr is large, and speeding it up if otherwise [1], [4].

Manuscript received April 27, 1998. This paper was recommended for publication by Associate Editor W. Wilson and Editor S. Salcudean upon evaluation of the reviewers' comments. This paper was presented in part at the IEEE International Conference on Robotics and Automation, Nagoya, Japan, May 1995.

F.-Y. Hsu is with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C.

L.-C. Fu is with the Department of Computer Science and Information Engineering and the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R.O.C.

Publisher Item Identifier S 1042-296X(00)06957-3.

Furthermore, the chamfering force for a constant feedrate and a specific chamfer depth is proportional to the cross-sectional area of the burr [2], and hence, the feedrate of cutting tool is controlled to keep a constant chamfering force. Human operators, however, cannot manipulate the cutting tool to produce the desired chamfer depth as precisely as a deburring robot can. In [3]–[9], impedance control has been applied to compliantly yield an allowable chamfer depth. In [10], hybrid position/force control is applied to yield a desired position subject to contact force control. Since the desired chamfer depth can be defined well by constructing the contact force model, in this paper, we apply hybrid position/force control to the deburring robot. On the other hand, to remove burrs with various sizes, the deburring robot must follow some control policy for the feedrate of the cutting tool. Some research has been done on transferring human skills to control rules. In [4] and [5], the use of the linguistic approach and neural networks to transfer human skills to robot deburring was introduced. Furthermore, an outer-loop intelligent planner was realized by means of a fuzzy neural network [6]. Therefore, in this paper, we propose a control architecture for a deburring robot, consisting of an outer-loop command generator and an inner-loop hybrid position/force controller. The function of the outer-loop command generator is mainly to instruct the inner-loop controller to control the attached cutting tool to produce a desired positional profile. The inner-loop controller drives the robot to perform the aforementioned positional profile command in a force-controlled manner so as to yield a desired chamfer depth.

Fuzzy control has become a very popular approach for performing the task of controller design because it is able to transfer human skills to some linguistic rules. Therefore, fuzzy control is often applied to some ill-defined systems or systems without mathematical models. It is difficult, however, to design fuzzy controllers systematically. With advances in related theories [19]–[21], fuzzy mechanisms can now play a role as a so-called universal approximator in the compact set, which can facilitate parameterization of vague system or uncertain systems. This explains why the current trend in theoretical development in this regard is to combine fuzzy control techniques with adaptive control theories to get adaptive fuzzy control. Studies have been conducted on the application of robot manipulators in the presence of system uncertainties [26]–[28]. Furthermore, fuzzy inference can act as a smooth interpolation operator to soften the control input. This prevents some robot manipulators systems from inducing high frequency noise or high-order unmodeled dynamics terms, in particular when the robot's end effector, equipped with an imprecise force sensor, comes into violent contact with an object. We propose a “soft”

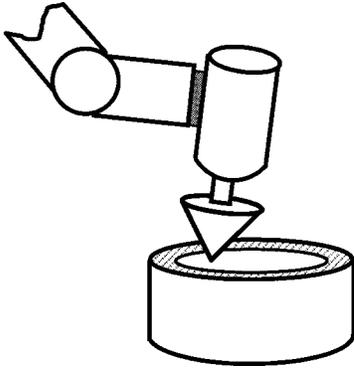


Fig. 1. The deburring robot performing the contour-following task.

adaptive fuzzy controller by choosing membership functions which are smooth B-spline basis functions [12] to achieve the aforementioned goal. To sum up, the proposed controller can perform the deburring control policy from the out-loop command generator and achieve the desired chamfer depth by the inner-loop adaptive fuzzy hybrid position/force controller which can adaptively update fuzzy rules and soften the control input.

This paper is organized as follows. Section II introduces a dynamic model which describes the behavior of a deburring robot in the Cartesian Frame. In Section III, we introduce an adaptive fuzzy hybrid position/force controller for a deburring robot, consisting of an outer-loop generator and an inner-loop hybrid controller. After that, we introduce a refinement algorithm for the desired contact force and adaptive fuzzy hybrid control, and analyze the stability. Section IV gives an example for control of an industrial robot arm to performing deburring tasks. Finally, Section V provides some conclusion.

II. DYNAMIC MODEL OF A DEBURRING ROBOT IN THE CARTESIAN FRAME

Consider an n degree-of-freedom articulated deburring robot, as depicted in Fig. 1. Its dynamic model in Cartesian frame can be derived as follows [13], [14]:

$$M(x)\ddot{x} + C(x, \dot{x})\dot{x} + G(x) + D(\dot{x}) = f_\tau + f \quad (1)$$

where $x = [x_1, \dots, x_6]^T = [x_p^T, x_o^T]^T$, with $x_p \in \mathbb{R}^3$ being its position vector of cutting tool and $x_o \in \mathbb{R}^3$ being the Euler angles representing orientation vector. $M(x) \in \mathbb{R}^{6 \times 6}$ is the inertia matrix, $C(x, \dot{x})\dot{x}$ is the vector representing the centrifugal and Coriolis forces which satisfy that $\dot{M} - 2C$ is a skew-symmetric matrix, $G(x)$ is the vector of gravitational forces, $D(x, \dot{x})$ is the vector of friction forces, f_τ is the vector of control input forces and moments, and f is the vector of cutting forces and moments. To simplify the underlying control problem, we will assume that the cutting tool is in contact with the part only at a single point so that the moment exerted by the cutting tool equals zero. Therefore, the cutting force/moment f can be denoted as $f = [f_n^T + f_t^T, 0, 0, 0]^T$, where $f_n \in \mathbb{R}^3$ is the vector of contact force perpendicular to the chamfer surface of the part, and $f_t \in \mathbb{R}^3$ is the vector of the chamfering force tangential to the chamfer contour of the part.

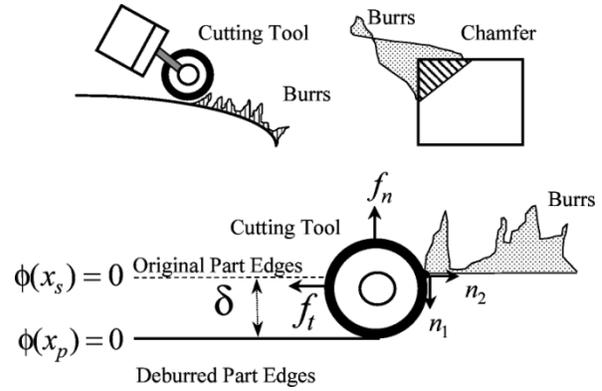


Fig. 2. The geometry of part edges during the deburring process.

To derive the dynamics of the chamfering force and the contact force during the deburring process, we let x_s represent the vector of the cutting tool position on the part's edge when the chamfer depth is zero (see Fig. 2), that is, x_s is subject to a constraint equation $\phi(x_s) = 0$. To express f_n and f_t on the chamfer surface, we define two unit orthogonal vectors, n_1 and n_2 . The former is opposite to the direction of the contact force and can be constructed as $n_1 = (-\nabla\phi(x_s)/\|\nabla\phi(x_s)\|)$, where ∇ represents a gradient operator. However, n_1 cannot be constructed directly from the above equation since x_s is not measurable. Therefore, we approximate n_1 by $n_1 \cong (-\nabla\phi(x_p)/\|\nabla\phi(x_p)\|)$, when the chamfer depth, say, $\delta \in \mathbb{R}^1$, is very small, which is due to the fact that $\nabla\phi(x_p) \times \nabla\phi(x_s) \cong 0$. On the other hand, n_2 is defined as being opposite to the direction of contour-following motion on a chamfer with zero depth, satisfying $n_1 \perp n_2$, and it can be expressed as

$$n_2 = \frac{\dot{x}_p - n_1 n_1^T \dot{x}_p}{\|\dot{x}_p - n_1 n_1^T \dot{x}_p\|} = \frac{(I - n_1 n_1^T) \dot{x}_p}{\|(I - n_1 n_1^T) \dot{x}_p\|} \quad (2)$$

As a result, we can rewrite the contact force and the chamfering force, respectively, as $f_n = -\|f_n\|n_1$ and $f_t = -\|f_t\|n_2$. Furthermore, it can be shown that the magnitude of f_t satisfies the following equation [4]:

$$\begin{aligned} \|f_t\| &= g|n_2^T \dot{x}_p| + h|n_1^T \dot{x}_p| + P'_{th} \\ &= g\|(I - n_1 n_1^T) \dot{x}_p\| + h|n_1^T \dot{x}_p| + P'_{th} \end{aligned} \quad (3)$$

where g and h are coefficients which vary with different burrs and chamfer depths, and P'_{th} is a constant value representing the threshold power of the cutting tool. However, it is very difficult to establish an exact model for the contact force. Here, we assume that the deburring robot is rigid enough, and that the spindle of the cutting tool rotates at high speed, so that the magnitude of the contact force will roughly obey the following relations:

$$\|f_n\| = \begin{cases} c_2 \delta + c_1 \delta + c_0, & \text{as } \delta \geq 0 \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where $c_0 > \epsilon_0 > 0$ is a threshold as the cutting tool comes into contact with the chamfer, $c_1 > \epsilon_1 > 0$ is the stiffness of the part, and $c_2 > \epsilon_2 > 0$ is the damping ratio of the cutting tool moving into the chamfer, where ϵ_0 , ϵ_1 , and ϵ_2 are positive constants.

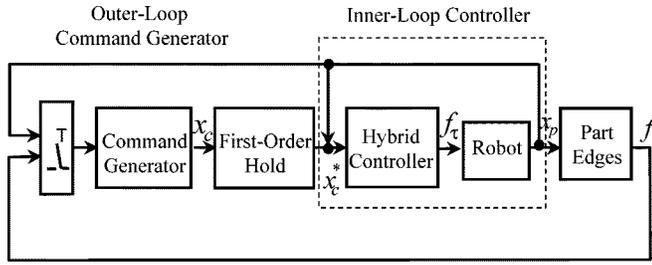


Fig. 3. The control architecture of the deburring robot.

III. ADAPTIVE FUZZY HYBRID POSITION/FORCE CONTROL OF THE DEBURRING ROBOT

In this section, we propose an adaptive fuzzy hybrid position/force control scheme whose architecture consists of an outer-loop command generator and an inner-loop hybrid position/force controller, as depicted in Fig. 3. The former determines a positional profile command such that the desired chamfering force and contour-following motion can be realized, whereas the latter aims to drive the robot to execute that command in a force-controlled manner so as to yield a desired chamfer depth. The positional command from the command generator is then transferred to a continuous input by passing it through a first-order hold mechanism for the inner-loop controller. Generally speaking, the servo-rate of the outer-loop controller is much slower than that of the inner-loop command generator, and, hence, we assume that the outer-loop has a nonzero servo-period T whereas the inner-loop controller is considered as a continuous type for simplicity. In fact the motion commands such as the chamfering force and the locus of the contour-following motion are not so time critical from the viewpoint of the inner-loop controller. Based on this control architecture, we can design the outer-loop command generator and inner-loop hybrid controller, separately.

A. Outer-Loop Command Generator

First, simplifying the model of the chamfering force as in (3), we consider the case where the cutting tool velocity in the direction of n_1 is much smaller than that in the direction of n_2 , i.e., $|n_1^T \dot{x}_p| \ll |n_2^T \dot{x}_p|$. As such, we let $\|f_t\|$ tend to a constant $\|f_{td}\|$ by letting the cutting tool position x_p follow a desired positional profile command x_c during the sampled-period, expressed as follows:

$$x_c(kT) = \begin{cases} x_p(kT) + \frac{\|n_2^T \dot{x}_p(kT)\| (\|f_{td}\| - P'_{th})}{\|f_t(kT)\| - P'_{th}} T n_2(kT), \\ \quad \text{if } \|f_t(kT)\| - P'_{th} > v_\epsilon \\ x_p(kT) + v_0 T n_2(kT), & \text{otherwise} \end{cases}$$

where T is the servo-period of the outer loop, $k = 1, 2, \dots$; v_0 is the default feedrate, set to prevent x_p from changing too quickly and damaging the part as $\|f_t\| - P'_{th}$ becomes excessively small, where $v_\epsilon > 0$ is its offset. Clearly, when $\dot{x}_p \rightarrow (x_c(kT) - x_c(kT - T))/T$, we can get the result that $\|f_t\|$ closely approximates $\|f_{td}\|$.

Finally, passing $x_c(kT)$ through a modified first-order hold filter to the inner-loop controller, we can get a continuous de-

sired position trajectory $x_c^*(t)$ with an initial value $x_c^*(0) = x_c(0)$ as follows:

$$x_c^*(t) = x_c^*(kT) + \frac{x_c(kT) - x_c^*(kT - T)}{T}(t - kT), \quad (5)$$

B. Inner-Loop Hybrid Position/Force Controller

Let us define the desired chamfer depth δ_d^* as a constant, and the desired contact force f_{nd} as follows:

$$f_{nd} = -(c_1 \delta_d^* + c_0) n_1. \quad (6)$$

To achieve δ_d^* under the force-controlled manner $f_n \rightarrow f_{nd}$, we define the desired position trajectory of the cutting tool $\delta_d(t)$, in the direction of the chamfer depth (n_1), which will eventually approach the desired chamfer depth δ_d^* . Furthermore, $\delta_d(t)$ is defined such that its initial condition $\delta_d(0) = 0$ and its time derivative is as follows:

$$\begin{aligned} \dot{\delta}_d &= k_\delta (\|f_{nd}\| - \|f_n\|) \\ &= k_\delta [-c_2 \dot{\delta} + c_1 (\delta_d^* - \delta)] \\ &= k_\delta e_f \end{aligned} \quad (7)$$

where $k_\delta > 0$ and $e_f = \|f_{nd}\| - \|f_n\|$ is defined as the current error of the contact force. Note that when $\dot{\delta} \rightarrow \dot{\delta}_d$ and $\delta \rightarrow \delta_d$, we can rewrite the above equation as

$$(1 + k_\delta c_2) \dot{\delta}_d + k_\delta c_1 \delta_d = k_\delta c_1 \delta_d^* \quad (8)$$

which implies that δ_d approaches δ_d^* exponentially; hence, the error of the contact force e_f can approach zero.

Given $x_c^*(t)$ and δ_d , we can augment the contour-following position command by summing the desired chamfer depth trajectory $\delta_d n_1$ and the desired contour-following profile trajectory x_c^* together. Thus, an augmented desired position trajectory, x_{pd} , can be expressed as

$$x_{pd}(t) = x_c^*(t) + \delta_d(t) n_1 \quad (9)$$

from which we can now define the position tracking error as $e_p(t) = x_{pd} - x_p$.

Similarly, we let the desired orientation of the cutting tool be given as x_{od} so that the collective position/orientation trajectory can be denoted as $x_d = [x_{pd}^T, x_{od}^T]^T$. Based on this definition, we can now define the vector of the tracking position/orientation tracking error vector as $e(t) = x_d(t) - x(t) = [e_p^T, e_o^T]^T$, where $e_o = x_{od} - x_o$, which allows us to rewrite the dynamic model of the manipulators system (1) as follows:

$$M(x)\ddot{e} = M\ddot{x}_d + C(x, \dot{x})\dot{x} + G(x) + D(\dot{x}) - f_\tau - f. \quad (10)$$

Furthermore, we define a sliding variable vector $s = \dot{e} + \lambda e$, where $\lambda \in \mathbb{R}^{6 \times 6}$ is a diagonal positive definite matrix, and reexpress the above dynamic model (10) as follows:

$$\begin{aligned} M\dot{s} + (K + C)s &= M(\ddot{x}_d + \lambda \dot{e}) + G + D \\ &\quad + C(\dot{x}_d + \lambda e) + Ks - f - f_\tau \end{aligned} \quad (11)$$

for some diagonal positive matrix $K \in \mathbb{R}^{6 \times 6}$. When the controller can force the right-hand side (RHS) of (11) to be zero, then the resulting system will be subject to the closed-loop dy-

namics $M_x \dot{s} + (K + C_x)s = 0$, which will not only assure stability of the system, but also guarantee exponential convergence of the tracking error e . As a result, an ideal control law can be designed as

$$f_r^* = M(\ddot{x}_d + \lambda \dot{e}) + G + D + C(\dot{x}_d + \lambda e) + Ks - f. \quad (12)$$

However, it is difficult to implement this ideal controller due to various kinds of model uncertainty. The major problems involved in implementing f_r^* arise from the following facts.

- The unknown parameters, c_0 , c_1 , and c_2 make determination of the actual desired trajectory difficult.
- The robot dynamics are too complex to precisely determine in advance.

Alternatively, a robust controller with the ability to on-line identify these unknown parameters can resolve the above-mentioned problems. Therefore, we propose an adaptive fuzzy hybrid position/force controller (depicted in Fig. 4) as a solution to these problems.

1) *Refinement of the Desired Contact Force*: The parameters associated with the contact force, c_0 , c_1 , and c_2 , are often estimated as values \hat{c}_0 , \hat{c}_1 , and \hat{c}_2 , respectively. The desired contact force virtually derived from these estimated values will be apart from the true desired one. We denote the virtual desired contact force as \hat{f}_{nd} and its magnitude can be expressed as

$$\|\hat{f}_{nd}\| = \hat{c}_0 + \hat{c}_1 \delta_d^*. \quad (13)$$

Moreover, we can get $\delta_d(\infty) = \delta_d^*$, when $\|\hat{f}_{nd}\|$ approach $\|f_n\|$. To achieve this, we build a virtual contact force \hat{f}_n and express it as

$$\|\hat{f}_n\| = \hat{c}_0 + \hat{c}_1 \delta + \hat{c}_2 \dot{\delta} \quad (14)$$

whereby the error between $\|f_n\|$ and $\|\hat{f}_n\|$ can be derived as follows:

$$e_n = \|f_n\| - \|\hat{f}_n\| = \tilde{c}_0 + \tilde{c}_1 \delta + \tilde{c}_2 \dot{\delta} \quad (15)$$

where $\tilde{c}_0 = c_0 - \hat{c}_0$, $\tilde{c}_1 = c_1 - \hat{c}_1$, and $\tilde{c}_2 = c_2 - \hat{c}_2$. To reduce this error, we update the parameters based on the following update law:

$$\dot{\hat{c}}_0 = \begin{cases} 0, & \text{as } \hat{c}_0 \leq \epsilon_0 \text{ and } e_n < 0 \\ r e_n, & \text{otherwise} \end{cases} \quad (16)$$

$$\dot{\hat{c}}_1 = \begin{cases} 0, & \text{as } \hat{c}_1 \leq \epsilon_1 \text{ and } e_n < 0 \\ r e_n, & \text{otherwise} \end{cases} \quad (17)$$

and

$$\dot{\hat{c}}_2 = \begin{cases} 0, & \text{as } \hat{c}_2 \leq \epsilon_2 \text{ and } e_n < 0 \\ r e_n, & \text{otherwise} \end{cases} \quad (18)$$

for some $r > 0$. Given update law (18), the following proposition is valid.

Proposition 1: If an ideal controller and the update law are given by (12) and (18), respectively, then the virtual desired contact force \hat{f}_{nd} described by (13) can approach the actual contact force f_n given in (6), which then yields the desired chamfer depth δ_d^* .

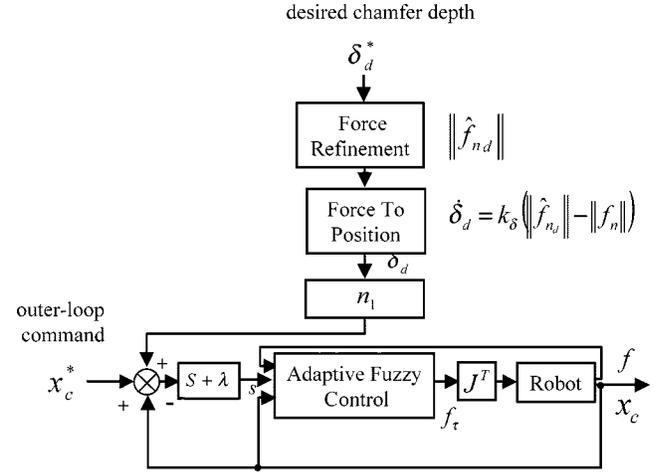


Fig. 4. The block diagram of the inner-loop controller.

Proof: Set a Lyapunov function candidate V as $V = 1/2(\tilde{c}_0^2 + \tilde{c}_1^2 + \tilde{c}_2^2 + s^T Ms)$. By differentiating V , we then get

$$\begin{aligned} \dot{V} &= -\tilde{c}_0 \dot{\tilde{c}}_0 - \tilde{c}_1 \dot{\tilde{c}}_1 - \tilde{c}_2 \dot{\tilde{c}}_2 - s^T K' s \\ &\leq -r e_n (\tilde{c}_0 + \tilde{c}_1 \delta + \tilde{c}_2 \dot{\delta}) - s^T K' s \\ &\leq -r e_n^2 - s^T K' s \leq 0 \end{aligned}$$

which guarantees boundedness of all the signals inside the system. Then, by some arguments about the uniform continuity of e_n and s and the Barbalat lemma [18], we can easily derive that, when $t \rightarrow \infty$, $e_n \rightarrow 0$ and $s \rightarrow 0$. Furthermore, as $s \rightarrow 0$, the tracking error e can be driven to zero exponentially [16]; as a result, we can obtain $\delta \rightarrow \delta_d$ and $\dot{\delta} \rightarrow \dot{\delta}_d$. Finally, we need to show that using this virtual desired contact force \hat{f}_{nd} can indeed lead to the result $\delta_d \rightarrow \delta_d^*$.

By definition of \hat{f}_{nd} in (13), the expression of δ_d in (7) can be reformed as

$$\dot{\delta}_d = k_\delta \left(\|\hat{f}_{nd}\| - \|f_n\| \right) = k_\delta \left(-c_2 \dot{\delta} + \hat{c}_1 \delta_d^* - c_1 \delta - \tilde{c}_0 \right).$$

Furthermore, we define the desired chamfer depth error as $e_{\delta^*} = \delta_d^* - \delta_d$. By differentiating e_{δ^*} , we can get

$$\begin{aligned} \dot{e}_{\delta^*} &= \dot{\delta}_d^* - \dot{\delta}_d \\ &= -\dot{\delta}_d \\ &= -k_\delta \left(-c_2 \dot{\delta} + \hat{c}_1 \delta_d^* - c_1 \delta - \tilde{c}_0 \right) \\ &= -k_\delta \hat{c}_2 \left(-\dot{\delta}_d \right) - k_\delta \hat{c}_2 \left(\dot{\delta}_d - \dot{\delta} \right) - k_\delta \hat{c}_1 \left(\delta_d^* - \delta_d \right) \\ &\quad - k_\delta \hat{c}_1 \left(\delta_d - \delta \right) + k_\delta \left(\tilde{c}_2 \dot{\delta} + \tilde{c}_1 \delta + \tilde{c}_0 \right). \end{aligned}$$

From (15), we can rewrite the above equation as

$$\begin{aligned} (1 + k_\delta \hat{c}_2) \dot{e}_{\delta^*} + k_\delta \hat{c}_1 e_{\delta^*} &= -k_\delta \hat{c}_2 (\dot{\delta}_d - \dot{\delta}) - k_\delta \hat{c}_1 (\delta_d - \delta) - e_n \\ &= f_\delta(t). \end{aligned} \quad (19)$$

Furthermore, we can derive e_{δ^*} as follows:

$$e_{\delta^*}(t) = \psi(t, 0) e_{\delta^*}(0) + \frac{1}{(1 + k_\delta \hat{c}_2)} \int_0^t \psi(t, \tau) f_\delta(\tau) d\tau \quad (20)$$

where $\psi(t, t_0) = \exp[\int_{t_0}^t -(k_\delta \hat{c}_1(\tau)/(1+k_\delta \hat{c}_2(\tau))) d\tau]$. Since \hat{c}_1 , \hat{c}_2 , and k_δ are positive, there exists some positive constant α_0 such that $\psi(t, t_0) \leq \exp[-\alpha_0(t - t_0)]$. Furthermore, since $\delta \rightarrow \delta_d$, $\dot{\delta} \rightarrow \dot{\delta}_d$, and $e_n \rightarrow 0$ exponentially in t , we can get $f_\delta(t) \rightarrow 0$ also exponentially in t , and $|f_\delta(t)| \leq \alpha_1 \exp(-\alpha_2 t)$ for some positive constants α_1 and α_2 . Then, we can get

$$|e_{\delta^*}(t)| \leq \exp(-\alpha_0 t) |e_{\delta^*}(0)| + \frac{1}{(1+k_\delta \hat{c}_2)} \frac{\alpha_1}{(\alpha_0 - \alpha_2)} \cdot [\exp(-\alpha_2 t) - \exp(-\alpha_0 t)]. \quad (21)$$

Therefore, we can conclude $e_{\delta^*}(t) \rightarrow 0$ and $\delta(t) \rightarrow \delta_d^*$ when $t \rightarrow \infty$. \square

Before proceeding to solve the control problems, we consider the robust controller as the following:

$$\begin{aligned} f_\tau &= Ks - f + H(\|x\|, s) \\ H(\|x\|, s) &= (b_1 + b_2\|x\| + b_3\|x\|^2) \operatorname{sgn}(s) + (b_4 + b_5\|x\|)s \end{aligned} \quad (22)$$

where b_1, \dots , and b_5 are defined as in the Appendix. Then, the following theorem is valid.

Theorem: If the control law is given as in (22), then the tracking error e will be driven to zero exponentially.

Proof: See the Appendix.

C. Adaptive Fuzzy Hybrid Control

According to Theorem 1, the control law will depend on the gain parameters b_1, \dots, b_5 of $H(\|x\|, s)$. However, in practice, they are very difficult to obtain, and if they are overestimated, then the system will be easily trapped in a limit-cycle condition for a time-delayed or slowly sampled control servo. Hence, an intelligent control methodology is needed. Here, we introduce an adaptive fuzzy control algorithm as an intelligent control solution. Let the actual control law f_τ be rewritten as follows:

$$f_\tau = Ks - f + f_s(\|x\|, s) \quad (23)$$

where Ks is a linear PD type compensator and f_s is the key adaptive fuzzy compensator. Here, we denote the fuzzy controller $u_f = [u_{f1}, \dots, u_{fn}]^T = f_s$ and consider the fuzzy controller u_f , consisting of n ($n = 6$) two-input single-output (MISO) fuzzy controllers, which are characterized by

$$u_{f_i} \triangleq u_{f_i}(\|x\|, s_i) \equiv u_{f_i}(y_0, y_i): \Omega_0 \times \Omega_i \rightarrow \mathfrak{R}$$

where u_{f_i} is the i th fuzzy controller; $y = [y_0, y_1, \dots, y_n]^T = [\|x\|, s_1, \dots, s_n]^T$ and $\|x\|$ and s_i (i.e., y_0 and y_i) are the i th input fuzzy variables; $\Omega_0 \equiv [-\Upsilon\Delta_0, \Upsilon\Delta_0], \dots, \Omega_n \equiv [-\Upsilon\Delta_n, \Upsilon\Delta_n]$, where Υ is a positive integer, and where it can be set arbitrarily large to obtain large enough compact sets; and $\Delta_0, \dots, \Delta_n$ are some positive real numbers. Here, each of the membership functions is given as an m th-order multiple dimension central B-spline function (as depicted in Fig. 5), the k th dimension of which is defined as follows:

$$\begin{aligned} N_{m_k}(y_k) &= \sum_{l=0}^{m+1} \frac{(-1)^l}{m!} \binom{m+1}{l} \left[\left(y_k + \left(\frac{m+1}{2} - l \right) \Delta_k \right)_+ \right]^m \end{aligned}$$

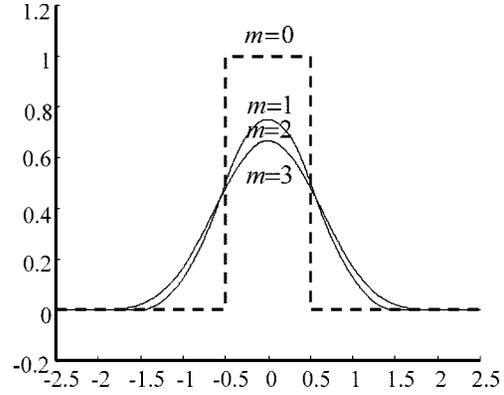


Fig. 5. The m th-order B-spline basis for $m = 0, 1, 2$, and 3.

where we use the notation

$$x_+ := \max(0, x). \quad (24)$$

The m th-order B-spline type of membership function has the following properties.

- It is an $(m - 1)$ th-order continuously differentiable function, i.e., $N_{m_k}(y_k) \in C^{m-1}$.
- It has local compact support, i.e., $N_{m_k}(y_k) \neq 0$ only for $y_k \in [-(m+1)/2\Delta_k, (m+1)/2\Delta_k]$.
- $N_{m_k}(y_k) > 0$ for $y_k \in (-(m+1)/2\Delta_k, (m+1)/2\Delta_k)$.
- It is symmetric with respect to the center point (zero point).
- $\sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} N_{m_0}(y_0 - i\Delta_0) N_{m_k}(y_k - j\Delta_k) = 1$, for $k \in \mathcal{Z}^+$.

Based on the definition of the local compact support, the above property can be rewritten as

$$\sum_{i \in I_{c_0}(y_0)} \sum_{j \in I_{c_k}(y_k)} N_{m_0}(y_0 - i\Delta_0) N_{m_k}(y_k - j\Delta_k) = 1$$

for $k \in \mathcal{Z}^+$, where $I_{c_k}(y_k)$ is an integer set, defined as follows:

$$\begin{aligned} I_{c_k}(y_k) &\equiv \left\{ i: \frac{y_k}{\Delta_k} - \frac{m+1}{2} < i < \frac{y_k}{\Delta_k} + \frac{m+1}{2}, i \in \mathcal{Z}, k \in \mathcal{Z}^+ \right\}. \end{aligned} \quad (25)$$

Then, the membership functions for the k th fuzzy variable y_k are defined as follows:

$$\mu_{k_i}(y_k) = N_{m_k}(y_k - i\Delta_k) \quad (26)$$

whose compact support is given as

$$\Omega_{k_i} = \left[\left(i - \frac{m+1}{2} \right) \Delta_k, \left(i + \frac{m+1}{2} \right) \Delta_k \right] \quad (27)$$

for $k = 0, \dots, n$ and $i = -\Upsilon, \dots, 0, \dots, \Upsilon$, which means that $y_k \in \operatorname{int}(\Omega_{k_i})$ implies that $\mu_{k_i}(y_k) > 0$ and $\Omega_{k_i} \equiv \cup_{i \in \{-\Upsilon, \dots, \Upsilon\}} \Omega_{k_i}$. It is possible that $\Omega_{k_i} \cap \Omega_{k_j} \neq \emptyset$, for some $i \neq j$; i.e., y_k can simultaneously fall into several compact supports. The indices labeling those supports are

equivalent to these in the definition of (25) and, hence, can be rewritten as

$$\begin{aligned} I_{c_k}(y_k) &\equiv \{i: y_k \in \text{int}(\Omega_{k_i}), i \in \mathcal{Z}, -\Upsilon \leq i \leq \Upsilon\} \\ &\equiv \{i: \Omega_{k_i} \subset \Omega_{c_k}(y_k)\} \end{aligned} \quad (28)$$

where $\Omega_{c_k}(y_k)$ is the union set of these compact supports, defined as follows:

$$\Omega_{c_k}(y_k) \equiv \cup_{i \in I_{c_k}(y_k)} \Omega_{k_i} \quad (29)$$

which means that $i \in I_{c_k}(y_k)$ is equivalent to $y_k \in \Omega_{k_i}$.

From [19], we can represent the above fuzzy controllers as follows:

$$\begin{aligned} u_{f_k} &= \frac{\sum_{i=-\Upsilon}^{\Upsilon} \sum_{j=-\Upsilon}^{\Upsilon} \mu_{0_i}(y_0) \mu_{k_j}(y_k) \theta_{k(ij)}}{\sum_{i=-\Upsilon}^{\Upsilon} \sum_{j=-\Upsilon}^{\Upsilon} \mu_{0_i}(y_0) \mu_{k_j}(y_k)} \\ &= \sum_{i=-\Upsilon}^{\Upsilon} \sum_{j=-\Upsilon}^{\Upsilon} \mu_{0_i}(y_0) \mu_{k_j}(y_k) \theta_{k(ij)} \\ &= \sum_{i=-\Upsilon}^{\Upsilon} \sum_{j=-\Upsilon}^{\Upsilon} \nu_{k(ij)}(y_0, y_k) \theta_{k(ij)} \\ &= \sum_{i \in I_{c_0}(y_0)} \sum_{j \in I_{c_k}(y_k)} \nu_{k(ij)}(y_0, y_k) \theta_{k(ij)} \\ &= \theta_k^T \nu_k \end{aligned} \quad (30)$$

where i, j are integer indices, $\nu_{k(ij)}(y_0, y_k)$ is the fuzzy basis function, $\theta_{k(ij)}$ is the parameter

$$\theta_k = [\theta_{k(-\Upsilon-\Upsilon)}, \dots, \theta_{k(00)}, \dots, \theta_{k(\Upsilon\Upsilon)}]^T \in \mathbb{R}^{(2\Upsilon+1)^2}$$

and

$$\nu_k = [\nu_{k(-\Upsilon-\Upsilon)}, \dots, \nu_{k(00)}, \dots, \nu_{k(\Upsilon\Upsilon)}]^T \in \mathbb{R}^{(2\Upsilon+1)^2}.$$

Furthermore, the proposed fuzzy controller (30) can be realized by means of a network implementation, as shown in Fig. 6.

Define the k th element of a new vector $y_{\Delta} = [y_{\Delta_0}, \dots, y_{\Delta_n}]^T$ as follows:

$$y_{\Delta k} = \begin{cases} y_k, & \text{as } y_k < -\Delta_k \text{ or } y_k > \Delta_k \\ 0, & \text{otherwise (i.e., } y_k \in [-\Delta_k, \Delta_k]) \end{cases} \quad (31)$$

so that

$$\dot{y}_{\Delta k} = \dot{y}_k \text{ for } y_{\Delta k} \neq 0.$$

Here, we want to design u_f so as to satisfy the following:

$$\begin{aligned} u_{f_k}(\|x\|, s_k) &= \begin{cases} k_{f_k}(\|x\|) \text{sgn}(s_k), & \text{if } s_k \notin [-\Delta_k, \Delta_k] \\ k_{s_k}(\|x\|, s_k), & \text{otherwise} \end{cases} \\ &\equiv u_{f_k}(y_0, y_k) \\ &= \begin{cases} k_{f_k}(y_0) \text{sgn}(y_k), & \text{if } y_k \notin [-\Delta_k, \Delta_k] \\ k_{s_k}(y_0, y_k), & \text{otherwise} \end{cases} \end{aligned} \quad (32)$$

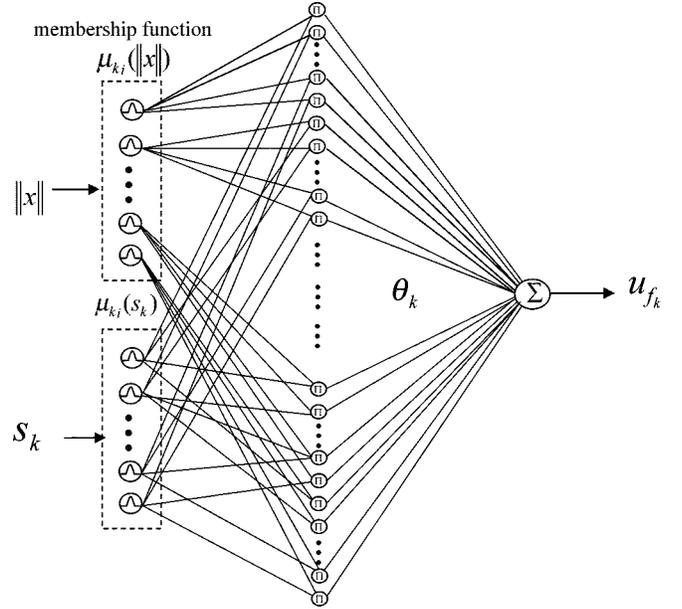


Fig. 6. The network implementation of adaptive fuzzy control.

where $k_f(\|x\|, s) \geq |H(\|x\|, s)|$, $k_s(\|x\|, s) = [k_{s_1}, \dots, k_{s_n}]^T$ is a smooth function vector so as to make u_f smooth and $[-\Delta_k, \Delta_k]$ is regarded as a designated dead-zone range which can be arbitrarily set.

Proposition 2: If the fuzzy control law u_f is given as in (30), then there exists a class of fuzzy controllers which can satisfy expression (32).

Proof: To prove this, we have to assure that u_f satisfies two properties, namely, (a) $\text{sgn}(u_{f_k}) = \text{sgn}(y_k)$ [i.e., $\text{sgn}(u_{f_k}) = \text{sgn}(s_k)$] and (b) $|u_{f_k}| \geq |H_k|$, when $y_k \in \Omega_k \setminus \Omega_{k0}$ (i.e., $s_k \in \Omega_k \setminus \Omega_{k0}$) for $k = 1, \dots, n$, respectively, from variable structure control theory.

Proof of a): From the definition I_{c_k} in (30), when $y_k \in \Omega_k \setminus \Omega_{k0}$, it follows that

$$\begin{cases} I_{c_k}(y_k) \subset \{-\Upsilon, \dots, -1\}, \\ y_{\Delta k} < 0 \text{ (i.e., } y_k < -\frac{m+1}{2}\Delta_k) \\ I_{c_k}(y_k) \subset \{1, \dots, \Upsilon\}, \\ y_{\Delta k} > 0 \text{ (i.e., } y_k > \frac{m+1}{2}\Delta_k) \end{cases}$$

for $k = 1, \dots, n$, respectively, where $\{-\Upsilon, \dots, -1\}$ and $\{1, \dots, \Upsilon\}$ are both integer sets. Then, the representation of the k th fuzzy controller can be rewritten as follows:

$$u_{f_k} = \begin{cases} \sum_{i=-\Upsilon}^{\Upsilon} \sum_{j=-\Upsilon}^{\Upsilon} \nu_{k(ij)} \theta_{k(ij)}, & \text{as } y_{\Delta k} < 0 \\ \sum_{i=-\Upsilon}^{\Upsilon} \sum_{j=-\Upsilon}^{\Upsilon} \nu_{k(ij)} \theta_{k(ij)}, & \text{as } y_{\Delta k} > 0. \end{cases}$$

Since $\nu_{k(ij)}$ is always positive, the sign of u_{f_k} can be determined by that of $\theta_{k(ij)}$. Hence, we set $\theta_{k(ij)} < 0$ for $-\Upsilon \leq j \leq -1$ and $\theta_{k(ij)} > 0$ for $1 \leq j \leq \Upsilon$.

As a result, we can conclude that $\text{sgn}(u_{f_k}) = \text{sgn}(y_k)$ for $k = 1, \dots, n$.

Proof of b): Given the following definitions for $k = 1, \dots, n$:

$$p_{\max k}(y_0, y_k) = \max \left\{ \sup_{x \in \Omega_{c_0}(y_0) \times \Omega_{c_k}(y_k)} |H_k(x)|, x \in \mathbb{R}^2 \right\}$$

$$\theta_{\min k}(y_0, y_k) = \min \{ |\theta_{k(ij)}|, i \in I_{c_0}(y_0), j \in I_{c_k}(y_k) \}$$

by setting $\theta_{\min k} \geq p_{\max k}$, we will obtain the following inequality:

$$|\theta_{k(ij)}| \geq \theta_{\min k} \geq p_{\max k} \text{ for } i \in I_{c_0}(y_0), j \in I_{c_k}(y_k).$$

By virtue of the fact that

$$\sum_{i=-\Upsilon}^{\Upsilon} \sum_{j=-\Upsilon}^{\Upsilon} \nu_{k(ij)} = \sum_{i \in I_{c_0}(y_0)} \sum_{j \in I_{c_k}(y_k)} \nu_{k(ij)} = 1 \quad (33)$$

and in the cases where $y_{\Delta k} \neq 0$ for $k = 1, \dots, n$, we can derive the following result:

$$\begin{aligned} |u_{f k}| &= \sum_{i \in I_{c_0}(y_0)} \sum_{j \in I_{c_k}(y_k)} \nu_{k(ij)} |\theta_{k(ij)}| \\ &\geq \sum_{i \in I_{c_0}(y_0)} \sum_{j \in I_{c_k}(y_k)} \nu_{k(ij)} \theta_{\min k} \\ &= \theta_{\min k} \geq p_{\max k} \\ &\geq |H_k|, \quad \text{as } y_{\Delta k} \neq 0 \end{aligned}$$

for $k = 1, \dots, n$. This completes our proof. \square

Now, we define the optimal parameter vector of the k th fuzzy controller as follows:

$$\theta_k^* = \arg \min \left\{ \sup_{y_0 \in \Omega_0, y_k \in \Omega_k \setminus \Omega_{k_0}} \sum_{i=-\Upsilon}^{\Upsilon} \sum_{j=-\Upsilon}^{\Upsilon} \nu_{k(ij)} \theta_{k(ij)} \cdot \text{sgn}(y_k) \geq |H_k| \right\}. \quad (34)$$

However, θ_k^* may not be available due to the complexity of H_k , $k = 1, \dots, n$. Therefore, the following adaptive law for updating the parameters vector θ_k will be necessary so that the tracking error can be driven toward the dead-zone range:

$$\dot{\theta}_{k(ij)} = r \nu_{k(ij)}(y_0, y_k) y_{\Delta k}, \quad \text{for } y_k \in \Omega_k \quad (35)$$

where $r > 0$ is some positive constant.

Finally, we design f_τ as

$$f_\tau = Ks - f + \text{diag}(\zeta) f_s(\|x\|, s) + [I - \text{diag}(\zeta)] \bar{H}(\|x\|, s) \quad (36)$$

where $\text{diag}(\bar{H}) \text{sgn}(s) > \text{diag}(H) \text{sgn}(s)$ and the k th element of $\zeta \in \mathbb{R}^6$ is defined as follows:

$$\zeta_k = \begin{cases} 1, & \text{as } y_k \in \Omega_k \\ 0, & \text{otherwise.} \end{cases} \quad (37)$$

Then, the following theorem states the condition under which the above-mentioned adaptive fuzzy control law will yield a satisfactory result.

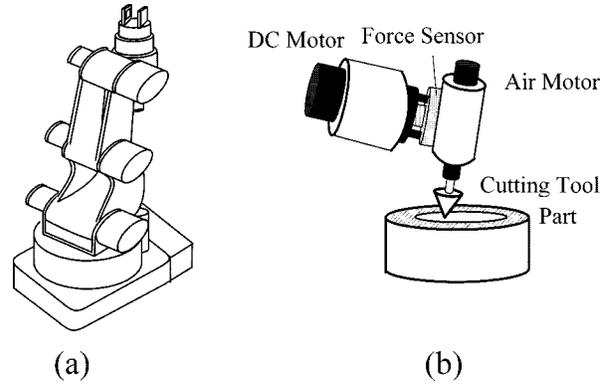


Fig. 7. Setup: (a) A-type robot arm and (b) the end effector for the deburring test.

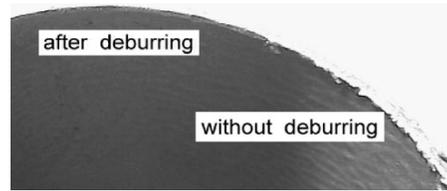


Fig. 8. Profile of burrs.

Theorem 2: If the update law and the control law are given as in (35) and (36), then the tracking errors will asymptotically converge to a neighborhood of zero.

Proof: If we substitute the control law into (11), then the dynamic equation can be rewritten as follows:

$$\begin{aligned} M\dot{s} &= -(K+C)s + M(\ddot{x}_d + \lambda\dot{e}) + C(\dot{x}_d + \lambda e) \\ &\quad + G + D - \text{diag}(\zeta) f_s - [I - \text{diag}(\zeta)] \bar{H}(\|x\|, s). \end{aligned}$$

Define matrices $\theta = [\theta_1, \dots, \theta_n] \in \mathbb{R}^{(2\Upsilon+1)^2 \times n}$, $\theta^* = [\theta_1^*, \dots, \theta_n^*] \in \mathbb{R}^{(2\Upsilon+1)^2 \times n}$, and $\nu = [\nu_1, \dots, \nu_n] \in \mathbb{R}^{2\Upsilon+1 \times n}$. Furthermore, let $\Phi = \theta^* - \theta$, and define a Lyapunov function

$$V = \frac{1}{2} s_\Delta^T M s_\Delta + \frac{1}{2r} \text{Tr}(\Phi^T \Phi) \quad (38)$$

where $s_\Delta = [y_{\Delta 1}, \dots, y_{\Delta n}]^T$ is as previously defined in (31), so that

$$\begin{aligned} \dot{V} &= -s_\Delta^T K s_\Delta + \frac{1}{2} s_\Delta^T (\dot{M} - 2C) + s_\Delta \{M(\ddot{x}_d + \lambda\dot{e}) \\ &\quad + C(\dot{x}_d + \lambda e) + G + D - \text{diag}(\zeta) f_s \\ &\quad - [I - \text{diag}(\zeta)] \bar{H}\} + \frac{1}{r} \text{Tr} \dot{\Phi}^T \Phi \\ &\leq -s_\Delta^T K s_\Delta - s_\Delta^T [I - \text{diag}(\zeta)] \text{diag}(\bar{H} - H) \text{sgn}(s) \\ &\quad + \sum_{k=1}^n \zeta_k s_{\Delta k} (\theta_k^{*T} \nu_k - H_k) \\ &\leq -s_\Delta^T K s_\Delta. \end{aligned}$$

By invoking Barbalat's lemma, it can be easily verified that $s_\Delta \rightarrow 0$ as $t \rightarrow \infty$. This result along with (31) implies that tracking errors are ultimately bounded within the designated intervals. This completes our proof. \square

Remark: The fuzzy controller (30) with the adaptive law (35) has the following advantages.

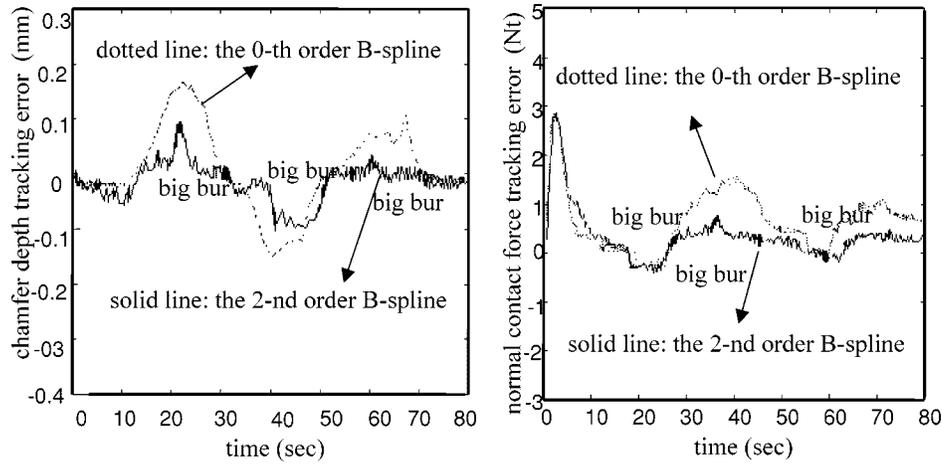


Fig. 9. Position/force tracking error results.

- It is a locally weighted fuzzy controller: only rules supported by compact set Ω_{c_j} need to be updated so that those rules will be locally weighted.
- It is a smooth fuzzy controller: the fuzzy controller (30) can behave as a smooth controller if we can choose membership functions which are smooth high-order B-spline functions. In our case, the order $m \geq 2$ is sufficient to make the controller smooth.

Besides, the computational complexity of the fuzzy controller (30) is discussed as follows [15]. From the above remarks, there are only $(m+1)^2$ rules that need to be computed. Then, it can be verified that the time complexity is $O(n \times (m+1)^2)$ and the space complexity is $O(n \times (2 \times \Upsilon + 1)^2)$.

IV. EXAMPLE

A five degree-of-freedom (DOF) articulated deburring robot arm equipped with a Zebra force sensor and a cutting tool (depicted in Fig. 7) is set up in the Intelligent Robot Laboratory with the Department of Computer Science and Information Engineering at National Taiwan University. The deburring task is performed by controlling the robot arm to drive its cutting tool to chamfer the inner edge of the cylinder part, which has a 30-mm cross section radius as shown in Fig. 8. The same figure also shows the profile of burrs, which include three major burrs (height = 1 mm) on the inner edge. The desired chamfer depth was set to 0.5 mm, and the desired chamfer force is set to 10 Nt. The control architecture based on an industrial PC has a 2.5-ms sampling period for the inner-loop hybrid controller and 25-ms sampling period for the outer-loop command generator. The cylinder is located on the table, expressed as $(x_1^2/30^2) + (x_2^2/30^2) = 1$ and $0 \leq x_3 \leq 50$. To slow down the speed of the contour-following, we set $v_0 = 1$ mm/s [shown in expression (5)] and the average speed is about 2.4 mm/s. Furthermore, to reduce the number of fuzzy rules, the adaptive fuzzy controller is solely applied to control the position/force trajectory in the plane $x_1 - x_2$. We set the membership functions as a class consisting of the zeroth order and the second-order B-spline basis with $\Delta_0 = 0.2x_{\max}$, $\Delta_1 = 0.2s_{1\max}$, and $\Delta_2 = 0.2s_{2\max}$, where x_{\max} , $s_{1\max}$, and $s_{2\max}$ are positive scaling factors used to normalize $\|x\|$, s_1 , and s_2 into the in-

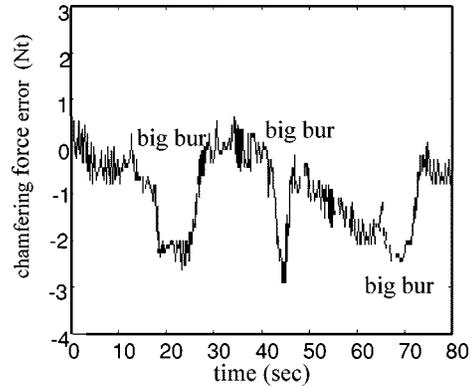


Fig. 10. Chamfering force error results.

terval $[-1, 1]$, respectively. As a result, the fuzzy rule number is equal to $2 \times (2l+1)^2 = 2 \times 11^2 = 242$. At the beginning, the initial parameter matrix θ is set to zero and updated to compensate for the uncertainties.

The chamfer depth error $\delta_d - \delta$ and the normal contact force trajectory error e_f are listed in Fig. 9 for the zeroth-order and the second-order B-spline membership functions, respectively. The chamfering force error and the surface finish result are given in Figs. 10 and 8. It is obvious that the adaptive fuzzy control implemented using the second-order B-spline membership functions has better performance. Figs. 11 and 12 show three-dimensional bar diagrams of the final parameters of the fuzzy controller after updating and three-dimensional plots of the control surface of the fuzzy controller with respect to the zeroth-order and the second-order B-spline membership functions. The fuzzy controller implemented using the second-order B-spline membership functions has a smoother control surface than does that implemented using the zeroth-order B-spline membership functions. To compare with the approach [2], the proposed controller not only has a better performance in the chamfering force and normal contact force but also yields a desired chamfer depth.

V. CONCLUSIONS

We have proposed an adaptive fuzzy hybrid position/force controller, which can update fuzzy rules to compensate for robot

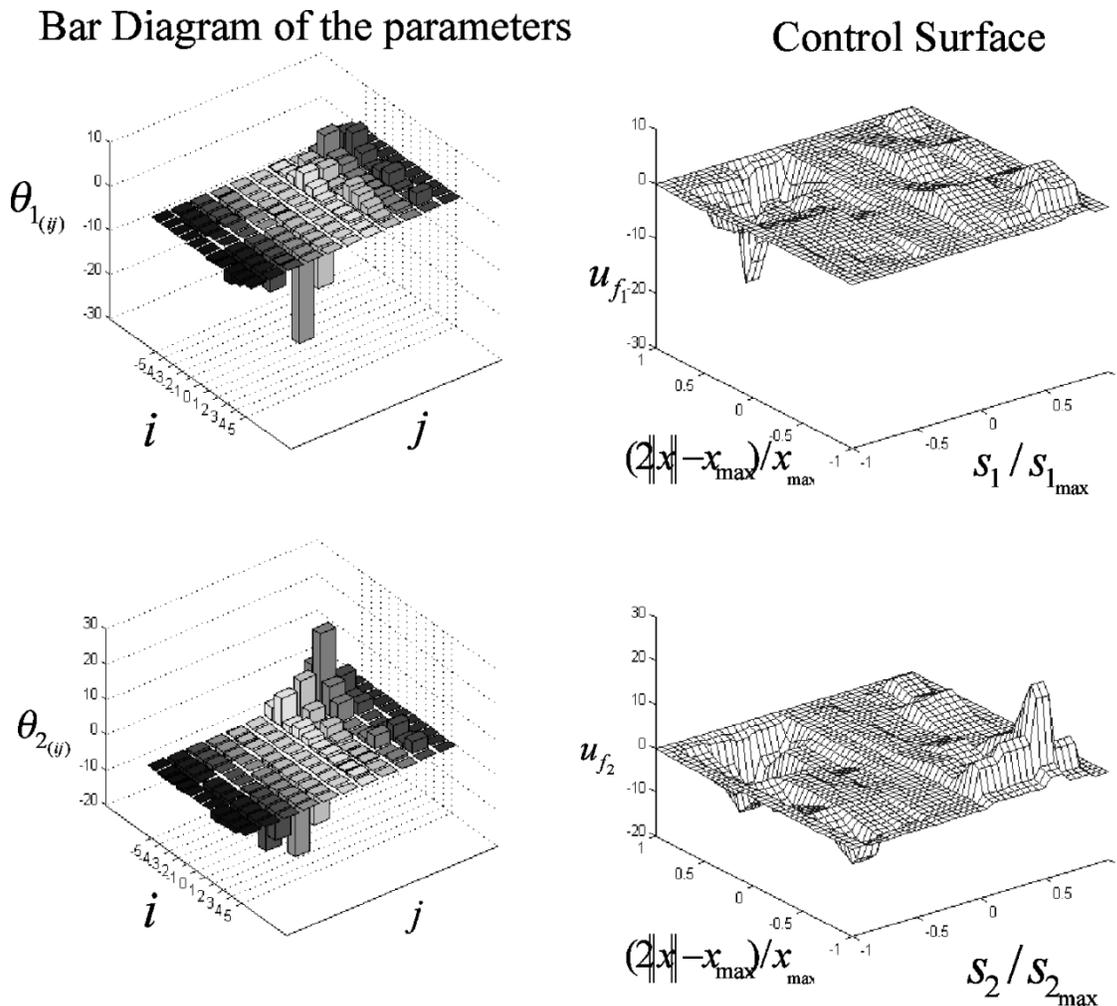


Fig. 11. Features of the zeroth-order B-spline based fuzzy controller.

dynamics along with the force dynamics induced by contact between a cutting tool and a part's edge, and can identify the actual desired contact force. Besides, we showed the smooth fuzzy controller has a better performance than nonsmooth one. Finally, various control results have been provided to verify the effectiveness of the developed work.

APPENDIX

Before proceeding to solve Theorem 1, we summarize the properties of the system dynamics (1) as follows:

- P1) $\underline{m} \leq \|M(x)\|_M \leq \bar{m}$ for some positive constants \underline{m} and \bar{m} ;
- P2) $\|C(x, \dot{x})\| \leq h_0 + h_1\|\dot{x}\|$ for some positive constants h_0 and h_1 ;
- P3) $x^T \dot{M}x = 2x^T Cx$ holds for any 6×1 vector x ;

where $\|\cdot\|$ and $\|\cdot\|_M$ are denoted as a vector norm and a corresponding induced matrix norm, respectively. Besides these properties, to make the control problem contractible, we make the following assumptions:

- A1) $\|G(x) + D(x, \dot{x})\| \leq h_2 + h_3\|\dot{x}\|$ for some positive constants h_2 and h_3 ;

- A2) $\sup_{t>0} \|x_d\| \leq d_0$, $\sup_{t>0} \|\dot{x}_d\| \leq d_1$,
 $\sup_{t>0} \|\ddot{x}_d\| \leq d_2$ for some positive constants d_0 , d_1 and d_2 .

Furthermore, we define the following control parameter:

- D1) $\bar{\lambda} = \max\{\lambda\}$.

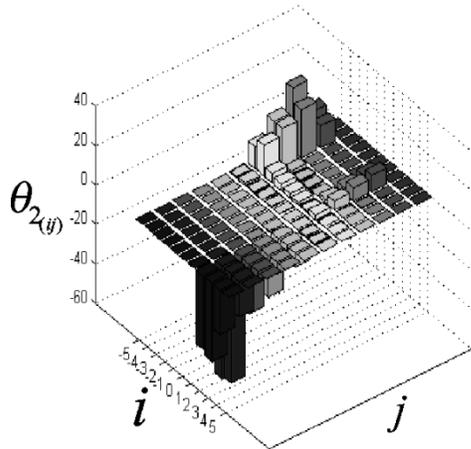
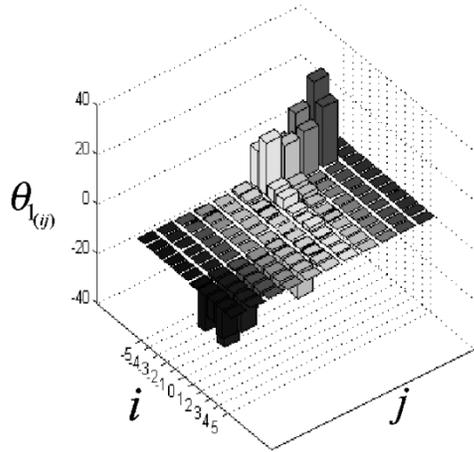
Then, b_1, \dots , and b_5 are defined as follows:

$$\begin{aligned} b_1 &= \bar{m}d_2 + h_2 + d_1h_0 + d_1h_3 + d_1^2h_1 + \bar{\lambda}^2\bar{m}d_0 \\ &\quad + \bar{\lambda}d_0h_0 + \bar{\lambda}d_0h_3 + \bar{\lambda}^2d_0^2h_1 + 2\bar{\lambda}d_0d_1h_1 \\ b_2 &= \bar{\lambda}(h_0 + h_3 + \bar{m} + 2h_1d_1 + 2\bar{\lambda}d_0h_1) \\ b_3 &= \bar{\lambda}^2h_1 \\ b_4 &= h_3 + \bar{m}\bar{\lambda} + \bar{\lambda}d_0h_1 + d_1h_1 \\ b_5 &= \bar{\lambda}h_1. \end{aligned}$$

Proof of Theorem 1: We set a Lyapunov function as $V = (1/2)s^T Ms$ and its time differential as follows:

$$\begin{aligned} \dot{V} &= -x^T Kx + s^T [M(\ddot{x}_d + \lambda\dot{e}) + C(\dot{x}_d + \lambda e) + G + D - H] \\ &\leq -x^T Kx + \|s\| [\bar{m}(d_2 + \bar{\lambda}|\dot{e}|) \\ &\quad + (h_0 + h_1\|\dot{x}\|)(d_1 + \bar{\lambda}|e|) + h_2 + h_3\|\dot{x}\|] - s^T H. \end{aligned}$$

Bar Diagram of the parameters



Control Surface

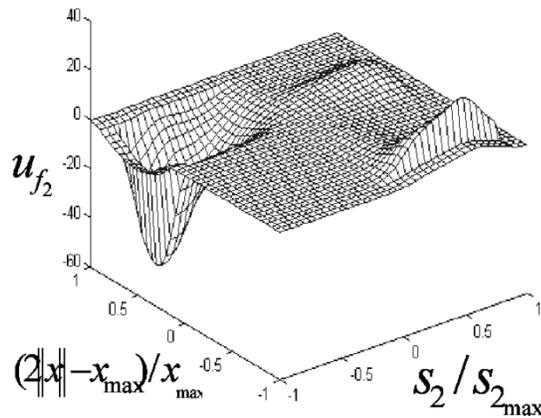
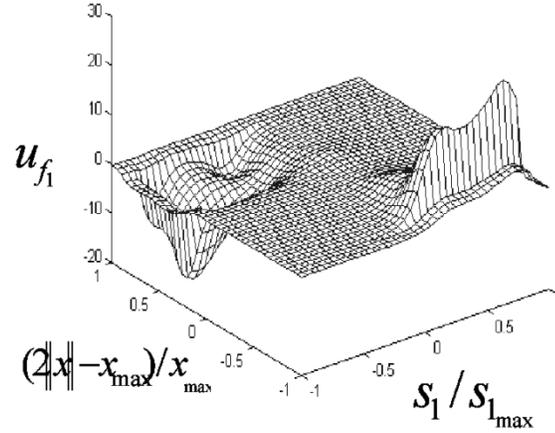


Fig. 12. Features of the second-order B-spline based fuzzy controller.

Based on the inequalities

$$\begin{aligned} \|e\| &\leq d_0 + \|x\| \\ \|\dot{e}\| &\leq \bar{\lambda}d_0 + \|s\| + \bar{\lambda}\|x\| \\ \|\dot{x}\| &\leq d_1 + \bar{\lambda}d_0 + \|s\| + \bar{\lambda}\|x\| \end{aligned}$$

we can get

$$\begin{aligned} \dot{V} &\leq -x^T Kx + \|s\| \left[(\bar{m}d_2 + h_2 + d_1h_0 + d_1h_3 + d_1^2h_1 \right. \\ &\quad \left. + \bar{\lambda}^2\bar{m}d_0 + \bar{\lambda}d_0h_0 + \bar{\lambda}d_0h_3 + \bar{\lambda}^2d_0^2h_1 + 2\bar{\lambda}d_0d_1h_1) \right. \\ &\quad \left. + (\bar{\lambda}h_0 + \bar{\lambda}h_3 + \bar{\lambda}\bar{m} + 2\bar{\lambda}h_1d_1 + 2\bar{\lambda}^2d_0h_1) \|x\| \right. \\ &\quad \left. + \bar{\lambda}^2h_1\|x\|^2 + (h_3 + \bar{m}\bar{\lambda} + \bar{\lambda}d_0h_1 + d_1h_1) \|s\| \right. \\ &\quad \left. + h_1\bar{\lambda}\|x\|\|s\| \right] - \sum_{i=1}^6 |s_i| (b_1 + b_2\|x\| + b_3\|x\|^2) \\ &\quad - (b_4 + b_5\|x\|)\|s\|^2 \leq -s^T Ks. \end{aligned}$$

□

REFERENCES

- [1] H. Kazerooni, J. J. Kramer, and B. M. , "An approach to automated deburring by robot manipulators," *J. Dynam. Syst. Meas. Contr.*, vol. 108, no. 4, pp. 354–359, Dec. 1986.
- [2] M. G. Her and H. Kazerooni, "Automated robotic deburring of parts using compliance control," *J. Dynam. Syst. Meas. Contr.*, vol. 113, pp. 60–66, March 1991.
- [3] H. Kazerooni, "On the robot compliant motion control," *J. Dynam. Syst. Meas. Contr.*, vol. 111, pp. 416–423, Sept. 1989.
- [4] S. Lu and H. Asada, "Transferring manipulative skills to robots: Representation and acquisition of tool manipulative skills using a process dynamics control," *J. Dynam. Syst. Meas. Contr.*, vol. 114, pp. 220–228, June 1992.
- [5] —, "Teaching and learning of deburring robots using neural networks," in *Proc. 1993 IEEE Int. Conf. on Robotics and Automation*, 1993, pp. 339–345.
- [6] K. Kiguchi and T. Fukuda, "Intelligent position/force controller for industrial robot manipulators—Application of fuzzy neural networks," *IEEE Trans. Ind. Electron.*, vol. 44, pp. 753–761, Dec. 1997.
- [7] G. M. Bone and M. A. Elbestawi, "Robotic force for deburring using an active end effector," *Robotica*, vol. 7, pp. 303–308, 1989.
- [8] G. M. Bone, M. A. Elbestawi, R. Lingarkar, and I. Liu, "Force control for robotic deburring," *J. Dynam. Syst. Measure. Contr.*, vol. 113, pp. 395–400, Sept. 1991.
- [9] M. S. Ali, M. N. Noori, and J. Turi, "Automatic deburring utilizing a real-time impedance control strategy," *Comput. Structure*, vol. 46, no. 3, pp. 561–571, 1993.

- [10] D. Jeon and M. Tomizuka, "Learning hybrid force and position control of robot manipulators," *IEEE Trans. Robot. Automat.*, vol. 9, pp. 423–432, Aug. 1993.
- [11] J. Marsden, *Elementary Classical Analysis*. San Francisco, CA: W. H. Freeman, 1974.
- [12] C. K. Chui, *Wavelet Analysis and Its Application*. Boston, MA: Academic, 1991, vol. 1, An Introduction to Wavelets, pp. 80–90.
- [13] J. J. Craig, *Introduction to Robotics Mechanics and Control*. Reading, MA: Addison-Wesley, 1989.
- [14] M. W. Spong and M. Vidyasagar, *Robot Dynamics and Control*. New York: Wiley, 1989.
- [15] J.-C. Latombe, *Robot Motion Planning*. Norwell, MA: Kluwer, 1991, pp. 599–602.
- [16] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ: Prentice-Hall, 1991, pp. 278–284.
- [17] R. M. Sanner and J.-J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. Neural Networks*, vol. 3, pp. 837–863, Nov. 1992.
- [18] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
- [19] L.-X. Wang, *Adaptive Fuzzy Systems and Control: Design and Stability Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1994.
- [20] C. T. Lin and C. S. G. Lee, "Neural-network-based fuzzy logic control and decision systems," *IEEE Trans. Comput.*, vol. 40, no. 12, pp. 1320–1326, 1991.
- [21] H. T. Nguyen and V. Kreinovich, "On approximation of controls by fuzzy systems," in *Proc. 5th IFSA Congress*, Seoul, Korea, 1996, pp. 1414–1417.
- [22] V. S. C. Raviraj and P. C. Sen, "Comparative study of proportional-integral, sliding mode, and fuzzy logic controllers for power converters," *IEEE Trans. Ind. Applicat.*, vol. 33, no. 2, pp. 518–524, 1997.
- [23] J. S. Glower and J. Muunighan, "Designing fuzzy controllers from a variable structures standpoint," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 138–144, Feb. 1997.
- [24] S.-C. Lin and Y.-Y. Chen, "Design of self-learning fuzzy sliding mode controllers based on genetic algorithms," *Fuzzy Sets Syst.*, vol. 86, no. 2, pp. 139–153, Mar. 1997.
- [25] J. C. Wu and T. S. Liu, "Fuzzy control stabilization with applications to motorcycle control," *IEEE Trans. Syst., Man, Cybern. B*, vol. 26, pp. 836–847, Dec. 1996.
- [26] F.-Y. Hsu and L.-C. Fu, "Adaptive robust fuzzy control for robot manipulators," in *Proc. IEEE Conf. on Robotics and Automation*, 1994, pp. 629–634.
- [27] —, "A new design of adaptive fuzzy hybrid force/position controller for robot manipulators," in *Proc. IEEE Conf. on Robotics and Automation*, 1995, pp. 863–868.
- [28] —, "Intelligent robot deburring using adaptive fuzzy hybrid control," in *Proc. 27th Int. Symp. on Industrial Robots*, Milan, Italy, 1996, pp. 847–852.



Feng-Yi Hsu was born in Ping-Tung County, Taiwan, R.O.C., in 1969. He received the B.S. and M.S. degrees in aeronautical and astronautical engineering from National Cheng Kung University, in 1990 and 1992, respectively, and the Ph.D. degree in electrical engineering from National Taiwan University in 2000.

From 1997 to 1999, he was with AverMedia Technologies, Inc., as a Senior Engineer. His areas of research interest include intelligent control and applications of robots, fuzzy control, and nonlinear control.



Li-Chen Fu (S'85–M'88) was born in Taipei, Taiwan, R.O.C., in 1959. He received the B.S. degree from National Taiwan University in 1981 and the M.S. and Ph.D. degrees from the University of California, Berkeley, in 1985 and 1987, respectively.

Since 1987, he has been on the faculty and is currently is a Professor of both the Department of Electrical Engineering and the Department of Computer Science and Information Engineering, National Taiwan University. He now also serves as the Deputy Director of Tjing Ling Industrial

Research Institute of National Taiwan University. His areas of research interest include adaptive control, nonlinear control, induction motor control, visual tracking, control of robots, FMS scheduling, and shop floor control.

Dr. Fu is now a member of both the IEEE Robotics and Automation Society and the IEEE Automatic Control Society. From 1996 to 1998, he was appointed a Member of AdCom of IEEE Robotics and Automation Society, and will serve as the Program Chair of 2003 IEEE International Conference on Robotics and Automation. He is also a Board Member of the Chinese Automatic Control Society and the Chinese Institute of Automation Engineers. He has been the Editor of *Journal of Control and Systems Technology* and is currently an Associate Editor of the prestigious control journal, *Automatica*. Starting in 1999, he became Editor-in-Chief of a new control journal, called *Asian Journal of Control*.

Dr. Fu received the Excellent Research Award in the period of 1990–1993 and Outstanding Research Awards in the years of 1995 and 1998 from National Science Council, R.O.C., respectively, the Outstanding youth Medal in 1991, the Outstanding Engineering Professor Award in 1995, the Best Teaching Award in 1994 from Ministry of Education, and The Ten Outstanding Young Persons Award in 1999 of R.O.C.