Fuzzy Logic Cross-Coupling Controller for Precision Contour Machining

Sungchul Jee*

(*Received September 18, 1997*)

This paper introduces a new cross-coupling controller with a rule-based fuzzy logic control. It is asserted that (i) fuzzy logic controllers provide a better transient response (which is essential for better contour accuracy during transient motions) than the conventional controllers, such as PID controllers, and (ii) cross-coupling controllers perform better than axial controllers in trajectory tracking by machine tools. In this paper, a fuzzy logic controller and a cross-coupling controller are combined to reduce contour errors. A simulation of the FLCCC was performed and the FLCCC was implemented on a CNC milling machine. The simulation and the experimental results show improved contour accuracy over the conventional cross-coupling controller.

Key Words: CNC, Machine Tool, Contouring Accuracy, Cross-Coupling Control

1. Introduction

In order to achieve high precision machining, many efforts have been made to develop more accurate computerized numerical control (CNC) systems. In particular, advanced servo-control algorithms for feed drives such as feedback control, feedforward control, and adaptive control have been implemented (Koren and Lo, 1992).

In conventional CNC machines, each axis has an individual axial position error. The axial position error is the difference between the desired position and the actual one : the former is the output from an interpolator in a CNC system, and the latter is available through a position feedback device such as an encoder. Since the control loop is separate for each axis, contour errors (i.e., deviations from the desired path) can be caused by a mismatch in the loop parameters and a difference in the load and external disturbances on each axis. In addition, a nonlinear contour shape can cause large contour errors, especially at high feedrates.

* Department of Mechanical Engineering, Dankook University, 8 Hannam-Dong, Yongsan-Ku, Seoul 140-714, Korea

Since the individual axial controllers such as PID and feedforward controllers do not guarantee small contour errors (which are more important than the axial position errors in contour machining), it is necessary to utilize sophisticated multi-axis controllers. One of the methodologies addressing this problem is cross-coupling control (Koren, 1980; Kulkarni and Srinivasan, 1989, 1990; Chuang and Liu, 1991; Masory and Wang, 1991; Koren and Lo, 1991, 1992). The control objective of the cross-coupling controllers (CCC) is the reduction of contour errors rather than axial position errors, thereby considerably improving the contouring accuracy. However, the existing cross-coupling controllers cannot overcome machine tool hardware deficiencies, such as backlash and friction, and their transient responses, which are extremely important in high-feedrate machining, still needs some improvements.

Cutting tool motion during the machining of each segment of a part may be divided into transient and steady-state periods. The tool accelerates to its steady-state feedrate during the transient period, and cuts at a constant velocity during the steady-state period. Then the second transient occurs at the end of the segment as the tool

decelerates. In contour machining, each axis motion consists of aforementioned these periods. In low-feedrate machining, the distance moved during each transient period is negligible. However, in high-feedrate machining, the distance traveled during each transient period becomes significant. Consequently, in high-feedrate machining, the transient periods dominate, and a segment may be cut without tool motion even reaching the steady-state (i.e., purely with the acceleration and deceleration periods).

To improve the transient contouring accuracy and to reduce the contour errors due to disturbances, a new cross-coupling controller with rule-based fuzzy logic control has been developed. Fuzzy logic controllers may provide a better transient response (which is essential for better contouring accuracy during transient periods) than conventional controllers such as the PID controller. In addition, it is asserted that the fuzzy logic control is robust for disturbances such as friction which causes large contour errors in the low-velocity range. Consequently, this FLCCC can be applied to a wide range of feedrates in contour machining. A simulation analysis has been performed and the FLCCC has been implemented on a CNC milling machine. The simulation and the experimental results show that this controller is able to achieve high contour accuracies.

2. Fuzzy Logic Cross-Coupling Controller (FLCCC)

In contour machining, a cutting tool is instructed to track a reference point that moves along a desired contour. However, the machine dynamics such as the inertia of machine slides, the friction in the guideways, and the cutting force cause a lag between the reference and the actual points of the cutting tool. This position lag error and the contour error, which is the shortest distance between the cutting tool position and the desired contour, are illustrated in Fig. 1. The position lag error increases with respect to the increase in feedrate and depends on the curvature of the contour as well as on the friction and cutting force. The position lag error is composed of axial position error components (denoted by E_x and E_y in Fig. 1). The objective of most CNC control systems, where each axis is controlled independently, is to reduce the position lag error by reducing each axial position error component. As mentioned previously, small axial position errors do not always guarantee small contour errors, which are more important from the viewpoint of

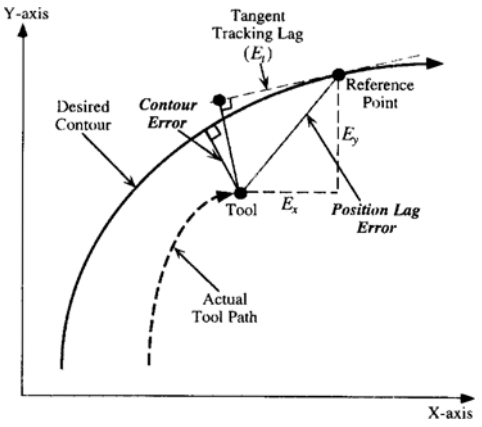
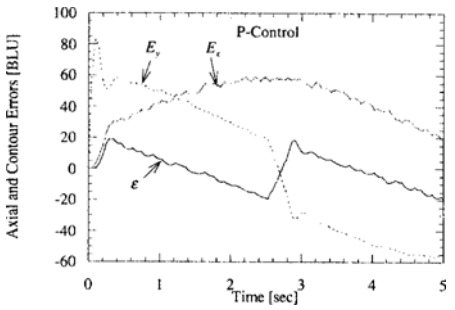
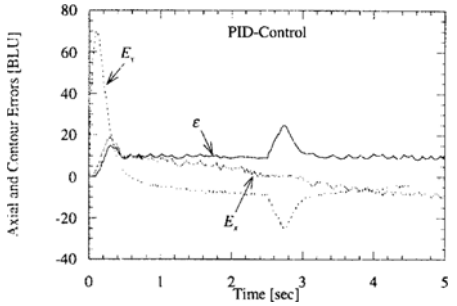


Fig. 1 Position lag error and contour error.



(a) P-Control



(b) PID-Control

Fig. 2 The relationship between axial position errors and contour errors.

the contour accuracy. Figure 2 shows simulation results which represent the relationship between the axial position errors (E_x and E_y) and the contour errors (ε) in the case of half-cycle biaxial circular motion with conventional P and PID controls for each axis. The PID controller results in much smaller axial position errors than the P controller, but does not effectively reduce the contour errors. From this example, it is obvious that the contouring accuracy does not necessarily depend on the axial position tracking accuracy. Here, the basic length unit (BLU) which corresponds to a system resolution is 10 μm.

In contrast to the individual axial control methods, the control objective of the cross-coupling approach is to eliminate the contour error thereby reducing the control-caused dimensional errors. The cross-coupling controller employs a new control architecture by which the servo-control level functions as one unit rather than separate loops. The CCC utilizes the error information of all axes simultaneously to produce accurate contours and reduces contour errors by a large factor compared with the traditional CNC controllers.

A block diagram of the proposed cross-coupling control for two axes is shown in Fig. 3. The contour error ε is calculated based on a mathematical contour error model of Koren and Lo (1991). In their contour error model, it was assumed that (i) the contour error is much smaller than the tracking lag error in the instantaneous tangent direction to the desired contour (denoted by E_t in Fig. 1), which is satisfied especially when the cross-coupling control is applied, and (ii) the tangent tracking lag error is much smaller than the instantaneous radius of curvature of the contour at the reference point. Under these assumptions, the contour error for a general nonlinear contour is approximated by Taylor series expansion:

$$\varepsilon = -E_x C_x + E_y C_y \tag{1}$$

where variable gains C_x and C_y are functions of contour geometry and axial position errors E_x and E_y, respectively.

For each axis, a proportional axial controller has been used with the same gain K_p. In Fig. 3, K_x and K_y represent the system open-loop gains multiplied by the encoder gains for the x and y axes, respectively, and τ_x and τ_y are the time constants of the axial drives. Each axial position error is calculated in real time as the difference between a reference position command and a position feedback from an encoder, and subsequently fed into the above contour error model.

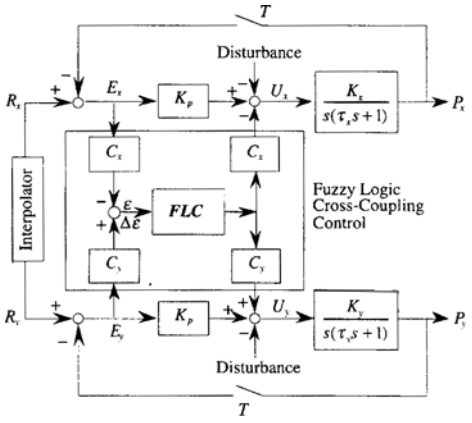
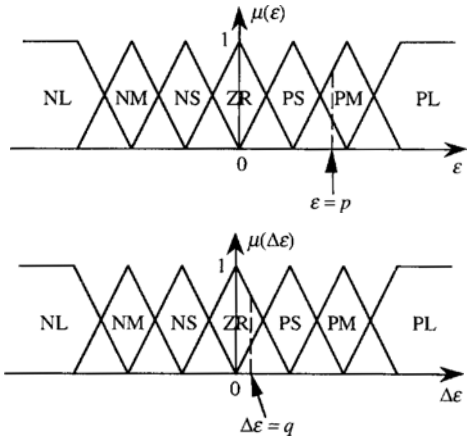
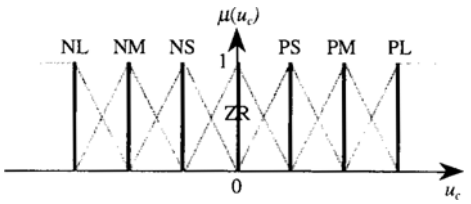


Fig. 3 Overall structure of the FLCCC.



(a) Controller inputs



(b) Controller output

Fig. 4 The membership functions.

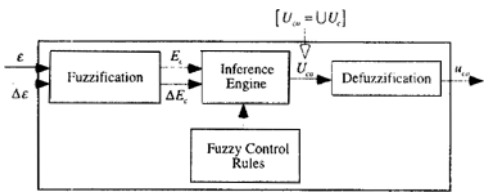


Fig. 5 Structure of the fuzzy logic controller in the FLCCC.

Then, through the fuzzy logic control law, the contour error correction command is determined. This command is multiplied by the gains C_x and C_y, and is decomposed into two components, which are fed to each loop to be combined with the axial control commands. Finally, the combined control commands U_x and U_y are generated and sent to the power amplifiers to drive the motors.

For the fuzzy logic control law, a proportional and differential (PD) type of fuzzy logic control (FLC) (its structure is shown in Fig. 5) is used because of its better transient contouring behavior, when applied to the CCC, compared with other types of fuzzy logic controls (i.e., proportional (P) and proportional-integral (PI) types of fuzzy logic controls). Thus, it is necessary to add an integral controller to eliminate steady-state contour errors. Accordingly, an integral controller is used in parallel with the FLC and included in the FLC block in Fig. 3.

2.1 Fuzzy logic control

This section briefly describes the elements of the fuzzy logic controller (inside the proposed cross-coupling controller) and explains its structure.

Membership Functions: A fuzzy set is characterized by a membership function whose value (i.e., truth value) represents a degree of membership to the fuzzy set having a value between 0 and 1. In general, an error (which is a difference between a desired process state and an actual process output) and the change in the error are used as inputs to a fuzzy logic controller. In this study, seven fuzzy sets have been defined for each controller input (contour error and the change in the contour error) and for the controller output, respectively. Figure 4 shows the defined membership functions. Two kinds of shapes are used in the proposed controller for the input membership functions (see Fig. 4A): a triangle (five membership functions in the middle) and a trapezoid (which covers both ends of the fuzzy input ranges) denoted by NL and PL in Fig. 4A. In general, each value of ε corresponds to (i.e., intersects) two membership functions. Similarly, each value of $\Delta\varepsilon$ corresponds to two functions. For example, in Fig. 4A, the value of $\varepsilon = p$ corresponds to two membership functions PS and PM, and the value of $\Delta\varepsilon = q$ corresponds to two functions ZR and PS. For the output membership functions, fuzzy singletons are used (denoted by solid lines in Fig. 4B).

Control Rules: The fuzzy control rules, which are composed of fuzzy conditional statements and utilize the linguistic values of fuzzy sets for the contour error (A_i), the change in the contour error (B_j) and the control action (C_{ij}), have the following form:

$$R_{ij}: \text{If } E_c = A_i \text{ and } \Delta E_c = B_j \text{ then } U_c = C_{ij} \tag{2}$$

In the proposed controller, for each A_i, B_j and C_{ij}, one of the following seven linguistic labels is assigned: Negative Large (NL), Negative Medium (NM), Negative Small (NS), Nearly Zero (ZR), Positive Small (PS), Positive Medium (PM), and Positive Large (PL). Since in the controller there are two control inputs (ε and $\Delta\varepsilon$) and seven fuzzy sets are defined for each input, there are a total of 49 control rules that have been determined and stored in the rule base. The control rules used in the proposed controller are shown in Table 1.

The fuzzy control rule base in the cross-coupling control was established based on the following principle. If the contour error at the current time step is closer to the zero contour error than the error at the previous time step, the engine inside the FLC infers that the machine is heading in the right direction, tending to reduce the contour error, and consequently only a relatively small control command is required. If the opposite is true, the engine infers that the machine is

Table 1 The fuzzy control rule base inside the proposed cross-coupling controller.

Control		If ΔE_c is						
	Actions	NL	NM	NS	ZR	PS	PM	PL
If E_c is	NL	NL	NL	NL	NL	NL	NL	NL
	NM	NL	NL	NM	NM	NS	NS	NS
	NS	NL	NM	NM	NS	NS	NS	ZR
	ZR	ZR	ZR	ZR	ZR	ZR	ZR	ZR
	PS	ZR	PS	PS	PS	PM	PM	PL
	PM	PS	PS	PS	PM	PM	PL	PL
	PL	PL	PL	PL	PL	PL	PL	PL

tending to increase the error and a relatively large command is required. In other words, if the contour error is small but not moving toward zero, a larger control action is needed than if the contour error is large but indicating a rapid movement toward zero error.

Structure: The structure of the FLC, which is the core of the FLCCC, is shown in Fig. 5. The FLC is composed of three main parts: fuzzification, inference engine with a rule base, and defuzzification. The inputs to the FLC are (i) the contour error at the current time step (ε), and (ii) the change in the contour error between the previous and current sampling time steps ($\Delta\varepsilon$). Thus, the rate of change in the contour error and its direction as well as the magnitude of the contour error are associated with determining the control actions.

Through fuzzification, the controller inputs are converted to fuzzy variables (E_c and ΔE_c), where each fuzzy variable has a corresponding linguistic label such as "positive small", "positive large", and "negative medium". After fuzzification, the converted fuzzy input variables are transformed into a fuzzy output variable (U_c) through the inference engine aided with a control rule base. The inference engine produces the overall fuzzy output (U_{co}) from individually activated control rules by a fuzzy implication function.

In this study, Mamdani's minimum operation rule was used as a fuzzy implication function (Lee, 1990), i.e., if the control rules have the form of Eq. (2), the truth value of overall resulting inference C (i.e., fuzzy output) can be obtained by

$$\mu_C = \max_{i,j}[\min(w_{ij},\ \mu_{C_{ij}})] \tag{3}$$

where

$$w_{ij} = \min(\mu_{A_i}(\varepsilon),\ \mu_{B_j}(\Delta\varepsilon)) \tag{4}$$

is the truth value of each rule R_{ij}. In other words, the truth value of each activated rule, w_{ij}, is determined by taking the minimum of the truth values of the fuzzy inputs (A_i and B_j), and then the truth value of the resulting inference for each rule is obtained by taking the intersection of w_{ij} and the corresponding output membership function C_{ij}. To obtain the overall resulting inference for all activated rules, the maximum of the individual inference induced by each rule is taken pointwise in the control output space.

The overall fuzzy control output U_{co} from the inference engine has a fuzzy value. However, in order to control a real system, a crisp control output is needed. In this context, the defuzzification action generates a crisp (nonfuzzy) control action u_{co} which best represents the inferred fuzzy control output. In order to remove the computational burden for real-time control, a simplified version of the center of area (COA) method was used for defuzzification:

$$u_{co} = \frac{\sum_{l=1}^{n_R} w_l \cdot C_l}{\sum_{l=1}^{n_R} w_l} \tag{5}$$

where n_R is the number of rules activated at each

time step (at most four in the proposed controller), w_l is the degree of fulfillment of the *if* partition of rule l, and C_l is the centroid of the output membership function (i.e., fuzzy singleton) corresponding to the resulting inference.

3. Simulation Analysis

A computer simulation was performed to investigate the performance of the FLCCC and compare it with that of the conventional cross-coupling control which uses a PID control law (PIDCCC) (Koren and Lo, 1991). Two types of reference contours were used in the simulation: (i) a circular contour and (ii) a rectangular corner. In the simulation, the following parameters were used with a proportional gain $K_p=1.0$ and a sampling time $T=0.01$ sec:

$$K_x=28.3,\ K_y=29.0;\quad \tau_x=0.055,\ \tau_y=0.056 \tag{6}$$

These are the actual parameters of a CNC machine. To make the simulation more realistic, a time-varying system was simulated by adding $\pm$ 5% variations to the four plant parameters: K_x, K_y, τ_x and τ_y. Based upon the results of the friction estimation experiments for the plain slideway (Jee and Koren, 1994), the friction disturbances were simulated as a function of the feedrate V (in mm/sec) with $\pm$8% random variations ΔF_d added to the average friction value. This results in the following friction models that were used in the simulation.

For the X-axis:

$$F_d(V)=\begin{cases} 13.95+\Delta F_d & \text{for } V\geq 12 \\ 0.05V^2-1.16V+20.49+\Delta F_d & \text{for } 0\leq V<12 \\ -0.04V^2-1.00V-18.27+\Delta F_d & \text{for } -12\leq V<0 \\ -11.84+\Delta F_d & \text{for } V<-12 \end{cases} \tag{7}$$

For the Y-axis:

$$F_d(V)=\begin{cases} 12.76+\Delta F_d & \text{for } V\geq 12 \\ 0.04V^2-1.12V+19.99+\Delta F_d & \text{for } 0\leq V<12 \\ -0.02V^2-0.68V-16.84+\Delta F_d & \text{for } -12\leq V<0 \\ -11.80+\Delta F_d & \text{for } V<-12 \end{cases}$$

The center values for the membership functions used in the simulations are listed in Table 2. Since the best accuracy we can achieve through control is about 1.5 BLUs, the center value of the "small" membership functions for the contour error was set to be $\pm$2.0 BLUs. The center values for the "medium" and "large" membership functions were set to be evenly spaced and were $\pm$4.0 and $\pm$6.0 BLUs, respectively. The center values of the control output membership functions were evenly spaced in the range of $\pm$60 which corresponds to about the half of 8 bit control commands. After setting the values for the above membership functions, the centers of the membership functions for the change of the contour error, which are also evenly spaced, were set to be as small as possible to obtain high derivative gains.

Table 2 Center values for the membership functions in the simulations.

Membership Functions	Contour Error	Contour Error Change	Control Action
NL	−6	−0.9	−60
NM	−4	−0.6	−40
NS	−2	−0.3	−20
ZR	0	0	0
PS	2	0.3	20
PM	4	0.6	40
PL	6	0.9	60

The PID control law of the PIDCCC can be represented by the following equation.

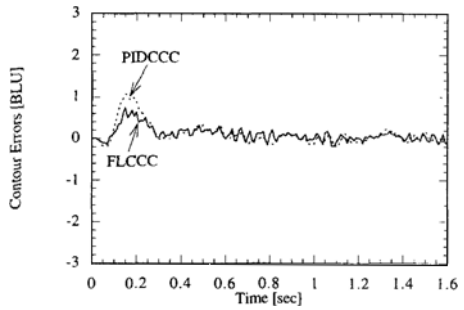
$$W(z)=W_P+W_I\frac{Tz}{z-1}+W_D\frac{z-1}{Tz} \tag{8}$$

where W_P, W_I and W_D are the proportional, integral, and derivative gains, respectively, and T is the sampling time. Because of the high nonlinearity existing in the FLCCC, it is not easy to compare the FLCCC and the PIDCCC in terms of their equivalent gains. However, both controllers were tuned such that their gains are as equivalent as possible, thereby making the comparison under equitable conditions. The proportional and integral gains of the PIDCCC were selected to have the same gains with the FLCCC. The derivative gain of the PIDCCC was tuned to be as large as possible while showing stable performance.

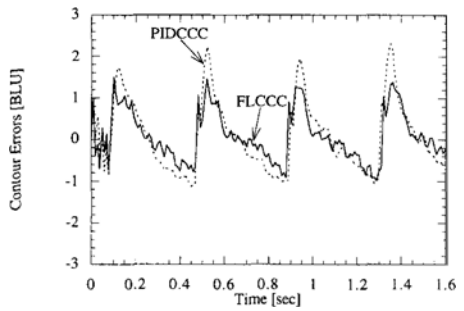
In order to deduce the range of controller gains, the FLC output in the proposed approach may be decomposed into two parts as follows:

Table 3 PIDCCC gains and the range of FLCCC gains in the simulations.

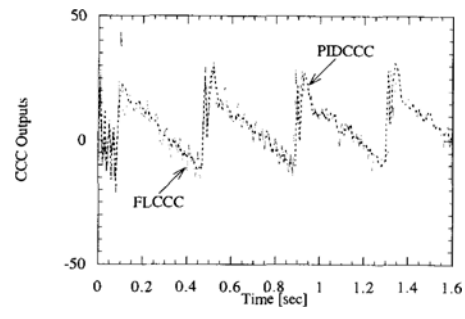
Gains	PIDCCC	FLCCC
Proportional	$W_P=10.0$	$W_P=10.0(K_1=10.0)$
Integral	$W_I=60.0$	$W_I=60.0$
Derivative	$W_D=0.1$	$0\le W_D\le 0.6(K_2=0.6)$



(a) Contour errors without the friction disturbances



(b) Contour errors with the friction disturbances



(c) Cross-coupling control outputs of (b)

Fig. 6 Simulation results for a circular contour.

$$u(k)=u_1(k)+u_2(k) \tag{9}$$

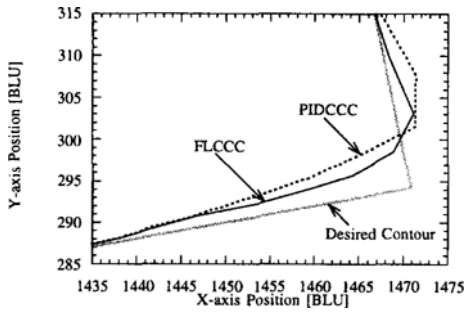
where $u_1(k)$ is the control output component issued only by the contour error, and $u_2(k)$ is the remaining part of the control output due to both the contour error and the change in the contour error. In other words,

$$u_1(k)=\varphi_1[\varepsilon(k)] \text{ and}$$
$$u_2(k)=\varphi_2[\varepsilon(k),\ \Delta\varepsilon(k)] \tag{10}$$

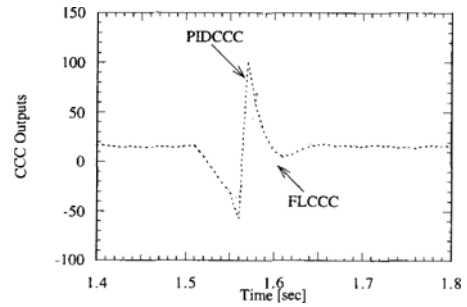
where φ_1 and φ_2 are nonlinear gain elements. If we define the gains which confine the sectors for $u_1(k)$ and $u_2(k)$ as K_1 and K_2, respectively, then

$$0<u_1(k)\,\varepsilon(k)<K_1[\varepsilon(k)]^2 \text{ and}$$
$$0<u_2(k)\,\Delta\varepsilon(k)<K_2[\Delta\varepsilon(k)]^2 \tag{11}$$

In relation to the FLCCC gains, the sectors K_1 and K_2 can be regarded as the maximum proportional and derivative gains, respectively, which are determined from the membership function parameters and the control rule base. The nonlinear gain φ_2 depends on ε as well as $\Delta\varepsilon$, and in general it is a different function according to the value of ε. However, with the rule base in Table 1, all these functions are confined to the region $0\le\varphi_2\le K_2\Delta\varepsilon$. Table 3 shows the PIDCCC gains



(a) Contour errors



(b) Cross-coupling control outputs

Fig. 7 Simulation results for a corner contour.

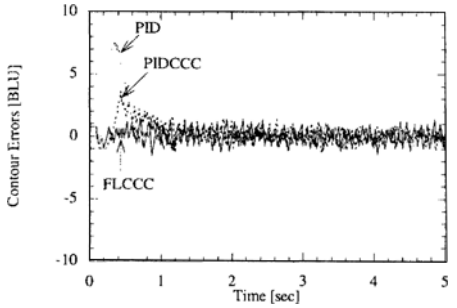
and the range of the FLCCC gains used in the simulations. In the FLCCC, K_1 and K_2 resulted in 10.0 and 0.6, respectively.

Figures 6 and 7 depict the simulation results of the FLCCC and those of the PIDCCC. In order to demonstrate the effect of an axis reversing its direction of motion and to investigate the effect of friction on contour errors, both cross-coupling controls were performed for circular motion with a radius of 5 mm and a feedrate of 1.1 m/min. The controls were performed with and without the friction disturbances in the simulator program, and the results were compared in Fig. 6. Without the friction disturbances (see Fig. 6A), both of the cross-coupling control methods show good contour accuracies except the relatively large contour errors caused by huge initial accelerations. The contour errors of the FLCCC was smaller than those of the PIDCCC during the transient period. On the other hand, with the disturbances (see Fig. 6B), the FLCCC shows better contour tracking performance than the PID cross-coupling control (PIDCCC). Thus, it appears that the FLCCC is less sensitive to friction disturbances than the PIDCCC: the maximum contour error at every 90 degrees around the circle was reduced by a factor of 1.5.

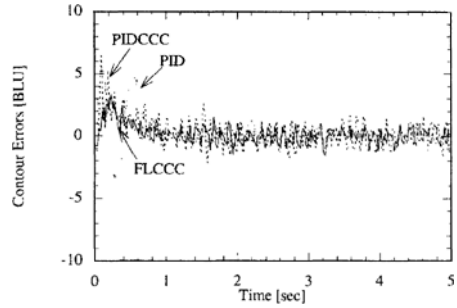
In order to simulate a corner cutting, a rectangular corner contour, denoted by the gray lines in Fig. 7, was used with a feedrate of 0.6 m/min, and the results are shown in Fig. 7. The FLCCC reduced the maximum contour error of the PIDCCC by a factor of 1.8, and the contour error of the FLCCC converged even faster than that of the PIDCCC. Therefore, if a contour has many short segments with sharp corners, the improvement achieved by the FLCCC will be significant.

4. Experimental Tests

In order to verify the theory, the proposed FLCCC as well as a conventional axial PID controller and the PIDCCC were implemented on a 3-hp CNC milling machine. This machine is



(a) Feedrate=0.1 m/min



(b) Feedrate=1.5 m/min

Fig. 9 Comparison of the contour errors of the FLCCC the PID and the PIDCCC for a linear contour.

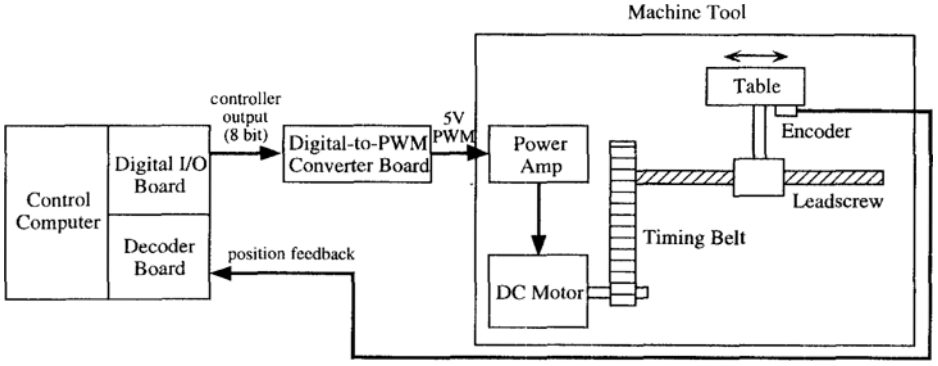


Fig. 8 Schematic of the experimental control system.

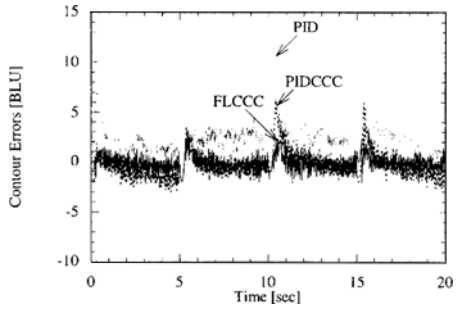
controlled by a general purpose computer (a 33MHz 80486-based PC), thereby enabling us to implement various interpolation and control software. The control computer is interfaced with linear encoders and a digital-to-pulse width modulation (PWM) converter through a quadrature decoder board and a digital I/O board, respectively. The linear encoders are attached on each axis for the table position feedback to the controller, and the digital-to-PWM converter generates a corresponding 5-volt PWM signal from an 8-bit digital control command for each axis. The 5-volt PWM signal for each axis is amplified through a power amplifier on the machine and sent to each DC servo-motor. Figure 8 portrays a schematic diagram of the experimental setup for one axis. The controls were implemented for two axes.

First, several experiments were performed with the axial PID controller and the PIDCCC, and then under the same conditions, experiments were run with the FLCCC for a linear and a circular contour. The typical results are shown in Figs. 9 and 10, respectively. A proportional gain $K_p=0.5$ was used for the axial controllers with both the FLCCC and the PIDCCC. For the axial PID controller, the proportional, integral and derivative gains were 1.5, 8.1 and 0.1, respectively. Table 4 shows the PIDCCC gains and the range of the FLCCC gains used in the experiments. To avoid unstable system behavior, both controller gains were set to be lower than the gains in the simulations. In the FLCCC, the distance between the centers of the output membership functions were set to be smaller near zero for steady-state contouring control.

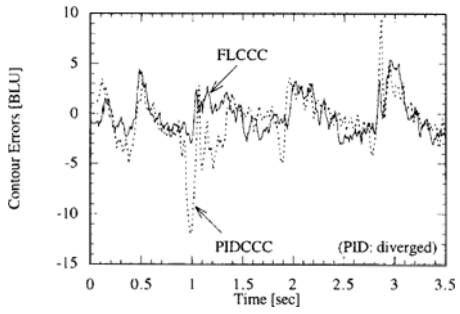
Using a linear contour $x=5y$, the contour errors were compared with different feedrates. For the lower feedrate (0.1 m/min), the FLCCC arrested the contour errors due to static friction, while the PID and the PIDCCC resulted in large initial contour errors because of stiction. For the higher feedrate (1.5 m/min), the FLCCC showed better results than the PIDCCC, not only during the transient period but also along the entire path. The steady-state contour errors of the PID are not significantly different from those of the FLCCC, but the transient contour errors are considerably reduced by the FLCCC.

Table 4 PIDCCC gains and the range of the FLCCC gains in the experiments.

Gains	PIDCCC	FLCCC
Proportional	$W_P=3.0$	$1.4\le W_P\le 2.9\,(K_1=2.9)$
Integral	$W_I=10.0$	$W_I=10.0$
Derivative	$W_D=0.05$	$0\le W_D\le 0.11\,(K_2=0.11)$



(a) Feedrate=0.38 m/min



(b) Feedrate=2.07 m/min

Fig. 10 Comparison of the contour errors of the FLCCC and the PIDCCC for a circular contour:

For a circular contour with a radius of 20 mm, the FLCCC always performed better than the PID and the PIDCCC. For the lower feedrate (0.38 m/min), the FLCCC reduced the contour errors due to stiction (every 90 degrees around the circle). This result also shows that the cross-coupling controllers perform better than axial controllers in trajectory tracking. For the higher feedrate (2.07 m/min), the PIDCCC resulted in large oscillation in the contour errors during the transient periods, while the contour errors of the FLCCC were bounded and remained within

Table 5 Comparison of the contour errors (unit: 10 μm).

		Linear Contour		Circular Contour	
		Transient	Steady-state	Transient	Steady-state
Low Feedrate	PIDCCC	3.53	0.52	6.73	1.36
	FLCCC	1.56	0.56	3.10	0.68
High Feedrate	PIDCCC	6.47	0.83	11.99	2.72
	FLCCC	3.33	0.53	5.41	1.95

the ± 5 BLU range (1 BLU=10 μm). On the other hand, the PID controller caused a saturation in the control commands, and the contour errors diverged. Thus, the result of the PID is not shown in Fig. 10b. When the feedrate was further increased, the PIDCCC also encountered the saturation. The FLCCC, however, continued to operate successfully.

The experimental results are summarized in Table 5. The absolute maximum contour errors during the transient periods and the root mean square (RMS) values of contour errors at the steady-states were compared.

5. Conclusions

A new cross-coupling controller with a fuzzy logic control law has been proposed and its validity has been verified through simulation and actual experimental analyses with different contour shapes and feedrates. For low feedrates, the proposed cross-coupling method reduced the contour errors due to stiction or negative viscous friction. For high feedrates, this new approach provided much better transient responses than the conventional cross-coupling control with a PID control law. Consequently, better contour tracking performance was obtained by using the fuzzy logic cross-coupling control compared with the existing cross-coupling control, regardless of the contour shapes and the feedrates.

Acknowledgement

This work was supported by the Dankook University research fund.

References

Chuang, H. Y. and Liu, C. H., 1991, "Cross-Coupled Adaptive Feedrate Control for Multiaxis Machine Tools," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 113, pp. 451~457.

Jamshidi, M., Vadiee, N., and Ross, T. J. (ed.), 1993, *Fuzzy Logic and Control: Software and Hardware Applications,* Prentice-Hall.

Jee, S. and Koren, Y., 1994, "Friction Compensation in Feed Drive Systems Using an Adaptive Fuzzy Logic Control," *Proceedings of the 1994 ASME Winter Annual Meeting,* DSC-Vol. 55-2, pp. 885~893.

Koren, Y., 1980, "Cross-Coupled Biaxial Computer Control for Manufacturing Systems," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 102, pp. 265~272.

Koren, Y., 1983, *Computer Control of Manufacturing Systems,* McGraw-Hill.

Koren, Y. and Lo, C. C., 1991, "Variable-Gain Cross-Coupling Controller for Contouring," *Annals of the CIRP,* Vol. 40, pp. 371~374.

Koren, Y. and Lo, C. C., 1992, "Advanced Controllers for Feed Drives," *Annals of the CIRP,* Vol. 41, pp. 689~698.

Kulkarni, P. K. and Srinivasan, K., 1989, "Optimal Contouring Control of Multi-Axis Drive Servomechanisms," *ASME Journal of Engineering for Industry,* Vol. 111, pp. 140~148.

Kulkarni, P. K. and Srinivasan, K., 1990, "Cross-Coupled Control of Biaxial Feed Drive Servomechanisms," *ASME Journal of Dynamic Systems, Measurement, and Control,* Vol. 112, No. 2, pp. 225~232.

Lee, C. -C., 1990, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller - Part I and II," *IEEE Transactions on System, Man, and Cybernetics,* Vol. 20, No. 2, pp. 404~435.

Masory, O. and Wang, J., 1991, "Improving Contouring System Accuracy by Two-Stage Actuation," *19th North American Manufacturing Research Conference Proceedings,* pp. 274~280.

Zadeh, L. A., 1965, "Fuzzy Sets," *Information and Control,* Vol. 8, pp. 338~353.

Zadeh, L. A., 1973, "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes," *IEEE Transactions on Systems, Man, and Cybernetics,* Vol. SMC-3, No. 1, pp. 28~44.

Zimmermann, H. -J., 1991, *Fuzzy Set Theory - and Its Applications,* Kluwer Academic Publishers.