

---

# 퍼지 보상기와 퍼지 동조기를 이용한 PID 제어기의 Deadzone을 포함한 시스템 제어

박재형\*, 김승철\*, 조용성\*, 최부귀\*\*

## Control of Systems Containing Deadzone of PID Controller using Fuzzy Compensator and Fuzzy Tuner

Jae-Hyung Park\*, Seung-Chul Kim\*, Yong-Seong Cho\*, Boo-Kwi Choi\*\*

### 요 약

PID 제어기는 미지의 Deadzone을 가진 시스템에 적용될 때, 제어 성능이 저하되는 단점을 나타낼 수 있다. 이러한 문제를 해결하기 위하여, 본 논문에서는 2층 구조의 퍼지 논리를 이용한 PID 제어기를 제안하였다. 이 제어기의 구조는 전형적인 PID 제어기에 퍼지 보상기와 퍼지 동조기를 추가하여 구성하였다. 제안된 제어 방법은 전형적인 PID 제어기에 비하여 뛰어난 과도응답과 정상상태 성능을 나타낸다. 이 방법이 플랜트의 정상상태 이득뿐만 아니라 Deadzone 비선형성에서도 강인하다. 제안된 제어기의 성능은 시뮬레이션을 통하여 검증하였다

### Abstract

A conventional PID controller has poor performance when it applied to systems with unknown deadzones. To solve this problem, this paper proposes PID controller using two layered-fuzzy logic. The structure of controller is reconstructed with fuzzy compensator and fuzzy tuner on the conventional PID controller. Our proposed control scheme shows superior transient and steady-state performance compared to conventional PID controller. The scheme is robust to variations in deadzone nonlinearities as well as the steady-state gain of the plant. The performance of the developed controller is verified through simulation.

---

\* 동아대학교 대학원 전자공학과

\*\* 동아대학교 전자공학과 교수

접수일자 : 1999년 1월 20일

I. 서론

PID 제어기는 간단한 구조로서 폭넓은 동작 조건에도 강인한 성능을 나타내기 때문에, 여러 산업 분야에 널리 이용되고 있다. 그러나 PID 제어기가 Deadzone을 포함한 시스템<sup>[1]</sup>을 제어할 때 정상상태오차가 나타날 수 있고, 또한 PID 계수 값을 조정하기가 쉽지 않다는 어려움이 따른다. 따라서 본 논문에서는 이를 해결하기 위해 2단 퍼지 논리를 포함한 PID 제어를 사용하여 이러한 문제점을 개선하고자 한다.

Deadzone은 포화, 히스테리시스 및 기어의 back rush와 같은 전자와 기계부<sup>[2]</sup>에 흔히 나타나는 현상으로 정확한 수치를 알 수 없으며, 시변이다. 따라서 시스템이 Deadzone을 포함하고 있을 경우, 정상상태 오차를 일으키는 원인이 되며, Deadzone의 폭이 크면 클수록 오차는 커진다.

더욱이, PID 제어기의 3가지 파라미터인 비례, 적분 및 미분 계수는 과도 응답을 최적화하고 정상상태오차를 제거할 수 있도록 조정되어야 하는데, 시스템의 변화와 Deadzone의 변화에 따라 파라미터를 재조정하는 것이 쉽지 않다. 따라서 제어기의 파라미터를 최적의 값으로 선택하기 위해 많은 노력과 시간이 필요하며, 이를 보다 쉽게 해결하기 위한 연구가 진행되어왔다.<sup>[3],[4]</sup>

본 논문에서는 Deadzone에 따른 정상상태오차를 줄이면서 PID 계수 값을 적절히 찾을 수 있도록 2단 퍼지 논리를 포함한 PID 제어를 제안하였다. 이 제어기가 개선된 성능을 나타내고 있다는 것을 시뮬레이션으로 검증하였다.

II. 퍼지 보상기

이 절에서는 Deadzone을 가진 시스템의 정상상태오차를 제거하여 출력응답을 개선하는 퍼지 보상의 원리를 기술하였다<sup>[5]</sup>.

Deadzone을 가진 액추에이터의 특성은 식 (1)과 그림 1로 표시된다.

$$D[v] = \begin{cases} s(v-d), & \text{if } v > d \\ 0, & \text{if } -d \leq v \leq d \\ s(v+d), & \text{if } v < -d \end{cases} \dots\dots (1)$$

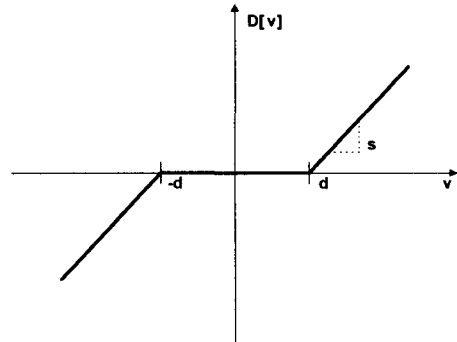


그림 1. Deadzone을 가진 액추에이터의 특성  
Fig. 1 Characteristics of actuator with deadzone

여기서,  $s$ 는 Deadzone 양측의 기울기이고,  $-d \sim d$ 는 Deadzone의 폭이다. ( $s, d > 0$ )  
시스템의 동 특성은 식 (2)와 같다.

$$\begin{aligned} e(k) &= r(k) - y(k) \\ \Delta e(k) &= e(k) - e(k-1) \\ C[e(k), \Delta e(k)] &= K_p \Delta e(k) + K_I e(k) \\ &\quad + K_D (\Delta e(k) - \Delta e(k-1)) \\ u(k) &= K_1 r(k) + C[e(k), \Delta e(k)] \\ u(k) &= D[u(k)] \\ y(k) &= F(z)[u(k)] \end{aligned} \dots\dots\dots (2)$$

- 여기서,  $r(k)$  : 기준입력
- $y(k)$  : 플랜트의 출력
- $u(k)$  : 액추에이터의 출력
- $v(k)$  : PID 제어기의 출력
- $e(k)$  : 추적오차
- $\Delta e(k)$  : 추적오차의 변화
- $C[e(k), \Delta e(k)]$  : 추적오차와 추적오차 변화의 선형합수
- $K_1$  : 피이드 포워드 이득
- $K_1 r(k)$  : 액추에이터의 입력
- $F(z)$  : 플랜트의 전달함수
- $K_p, K_I, K_D$  : PID제어기 파라미터

만약,  $C[0, 0] = 0$ 이고,  $r(k) = r$ 로 정해져 있다면, 액추에이터의 정상상태 입력은  $k_1 r$ 이다.

Deadzone이 존재하지 않을 경우( $d=0, s=1$ ), 플랜트의 출력은 식 (2)를 정리하면 식 (3)과 같다.

$$y(k) = P(z)[K_1 r(k) + C[e(k), \Delta e(k)]] = r - e(k) \quad (3)$$

$e(k) = r(k) - y(k)$ ,  $r(k) = r$ 일때, 정상상태에서의 시스템을 살펴보자. 이 경우에  $\Delta e(k) = 0$ 이라면, 식 (4)와 같다.

$$y(k) = K_s[K_1 r + K_f e(k)] = r - e(k) \quad (4)$$

$K_s$ 는  $K_s = \lim_{z \rightarrow 1} P(z)$ 인  $P(z)$ 의 상태이득이다. 또한  $K_1 = 1/K_s$ 이면 이 제어기는 잘 구성되었다고 볼 수 있으므로 식 (4)는  $K_s K_f e_{ss} = -e_{ss}$ 가 되어  $e_{ss} = 0$ 이 이 식의 해라고 볼 수 있다. 여기서,  $e_{ss}$ 는 정상상태오차이다.

Deadzone이 존재할 경우( $d > 0, s > 0$ ), 플랜트의 정상상태 출력은 식 (5)와 같다.

$$y_{ss} = K_s D[K_1 r + K_f e_{ss}] = r - e_{ss} \quad (5)$$

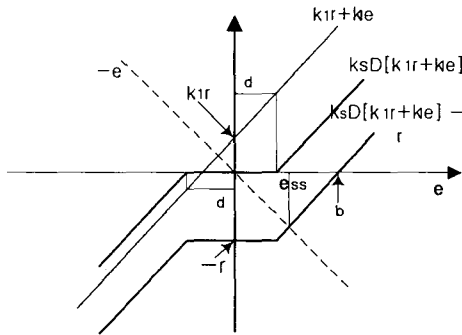


그림 2.  $-e$ 와  $K_s D[K_1 r + K_f e(k)] - r$   
Fig. 2  $-e$  and  $K_s D[K_1 r + K_f e(k)] - r$

퍼지 보상기의 원리는 다음과 같다. 그림 2에서  $b$ 만큼 좌측으로  $K_s D[K_1 r + K_f e(k)] - r$ 의 그래프를 이동하여 원점과 교차하도록 한다. 이는 정상상태 오차가 0이 된다는 것을 의미한다. 곡선을 이동하기 위해서는  $e$ 에  $b$ 를 더하고, 기준입력  $r$ 에 대하여 상수  $u'$ 를 추가함으로써 이러한 효과를 얻을 수 있다. PID 제어기의 기준 입력에 추가되

어진  $u'$ 의 적당한 값을 계산하기 위해 퍼지 논리 규칙을 사용했다. 이 제어기는  $d$ 와  $s$ 의 값이 변화하더라도 강인한 제어 성능을 나타낸다.

그리고 보상기의 퍼지 논리 법칙은 일반적인 퍼지 논리 법칙을 기초로 하여 작성하였고, 무게 중심법(COG)을 사용하였다. 제어 원리를 설명하면 다음과 같다.

첫째, 변수  $r'(k)$ 와  $e'(k)$ 를 다음과 같이 식 (6)으로 정의한다.

$$\begin{aligned} r'(k) &= r(k) + u'(k) \\ e'(k) &= e(k) + u'(k) \end{aligned} \quad (6)$$

여기서, 오차  $e(k) = r(k) - y(k)$ 와 오차 변화율  $\Delta e(k) = e(k) - e(k-1)$ 는 퍼지 보상기의 입력이며,  $u'(k)$ 는 퍼지 보상기의 출력이다.

보상기를 포함한 시스템의 동 특성은 식 (7)과 같다.

$$\begin{aligned} e(k) &= r(k) - y(k) \\ \Delta e(k) &= e(k) - e(k-1) \\ u'(k) &= F[e(k), \Delta e(k)] \\ r' &= r(k) + u'(k) \\ e'(k) &= r'(k) - y(k) \\ \Delta e'(k) &= e'(k) - e'(k-1) \\ C[e'(k), \Delta e'(k)] &= K_p \Delta e(k) + K_f e'(k) + K_D (e'(k) - 2e'(k-1) + e'(k-2)) \\ v(k) &= K_1 r'(k) + C[e'(k), \Delta e'(k)] \\ u(k) &= D[v(k)] \\ y(k) &= P(z)[u(k)] \end{aligned} \quad (7)$$

여기서,  $F[e(k), \Delta e(k)]$ 는 퍼지 논리에 대응되는 비선형 형태이며, 다음과 같은 언어 규칙  $L$ 의 집합과 소속 함수  $M$ 과 관련된 집합이고, 퍼지화기, 퍼지화 처리부 및 비퍼지화기의 세 단계로 구성된다.  $L$ 과  $M$ 은 다음과 같다.

$$\begin{aligned} L &= \{NB, NM, NS, ZO, PS, PM, PB\} \\ M &= \{M_{NB}, M_{NM}, M_{NS}, M_{ZO}, M_{PS}, M_{PM}, M_{PB}\} \end{aligned}$$

그림 3은 소속함수를 나타낸 것이다. 퍼지화의 처리는 언어 값이 지정한대로 입력  $e(k)$ 와  $\Delta e(k)$ 를 변환한다. 특히, 각 언어 값  $l \in L$ 에 대하여 다음의 식에 관련된 소속 함수  $M_l$ 에 있어서 입력  $e(k)$ 와  $\Delta e(k)$ 에 대한  $n_e(l)$ 과  $n_{\Delta e}(l)$ 의

수를 지정한다.

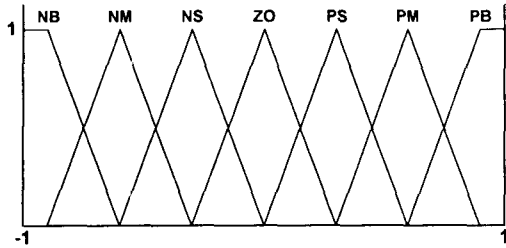


그림 3.  $e, \Delta e, u'$  소속 함수  
Fig. 3  $e, \Delta e, u'$  membership function

$$n_e(l) = M_l(C_e e(k))$$

$$n_{\Delta e}(l) = M_l(C_{\Delta e} \Delta e(k))$$

여기서  $C_e$  와  $C_{\Delta e}$ 는 스케일 계수이다.  $n_e(l)$  과  $n_{\Delta e}(l)$ 의 수는 퍼지 논리 처리부에 이용된다. 규칙은 다음과 같은 형식으로 표현된다.

Rule : if  $e(k)$  is  $l_e$ ,  $\Delta e(k)$  is  $l_{\Delta e}$ , then  $u'$  is  $l_{u'}$ ,  
보상기의 규칙은 총 23개로 표 1과 같다.

표 1. 보상기의 퍼지논리 규칙  
Table 1. Fuzzy logic rules for compensator

		$\Delta e(k)$						
		NB	NM	NS	ZO	PS	PM	PB
$e(k)$	NB		NB	NB	NB	NM		
	NM				NM			
	NS				NS	ZO		PM
	ZO	NM	NM	NS	ZO	PS	PM	PM
	PS	NS	NS	ZO	PS	PM	PB	PB
	PM				PM			
	PB							

그리고 실제적인 퍼지보상기의 출력은 다음과 같다.

$$u'(k) = u'(k-1) + F[e(k), \Delta e(k)] \dots (8)$$

### III. 퍼지 계수 동조기

이 절에서는 퍼지 논리를 이용하여 PID 제어기의 각 파라미터 값을 자동적으로 조정하는 동조기에 대하여 기술하였다. 여기서 사용된 접근법은 제어기 파라미터를 조정할 수 있는 퍼지 규칙과 추론을 정의하는 것이다.

$K_P, K_D$ 는  $[K_{P, \min}, K_{P, \max}]$ ,  $[K_{D, \min}, K_{D, \max}]$ 에 의해 미리 규정된 영역 내에 존재한다고 가정한다. 여기서  $K_{P, \min}, K_{P, \max}, K_{D, \min}$  및  $K_{D, \max}$ 는 Zeigler-Nichols 방법에 의해 계산된 값들이다. 이와 같은 방법으로 적당한 영역이 실험적으로 결정되면 식(16)에서 주어진다. 편리를 위해  $K_P'$ 와  $K_D'$ 는 아래의 선형 변형 영역에서 정규화 된다.

$$K_P' = (K_P - K_{P, \min}) / (K_{P, \max} - K_{P, \min}) \dots (9)$$

$$K_D' = (K_D - K_{D, \min}) / (K_{D, \max} - K_{D, \min})$$

제안된 방법에서 PID 파라미터는 오차  $e(k)$ 와 오차 변화율  $\Delta e(k)$ 에 의해 결정된다. 적분 시간상수는 미분 시간상수와와의 관계로 결정된다.

$$T_i = \alpha T_D \dots (10)$$

그리고 적분 이득은 아래에 의해 얻어진다.

$$K_I' = K_P / (\alpha T_D) = K_P^2 / (\alpha K_D) \dots (11)$$

파라미터  $K_P', K_D'$ 와  $\alpha$ 는 아래 형식의 퍼지 규칙의 집합에 의해 결정된다.

If  $e(k)$  is  $A_i$  and  $\Delta e(k)$  is  $B_i$  then  $K_P'$  is

$$C_i, K_D' \text{ is } D_i, \alpha = \alpha_i, i = 1, 2, 3, \dots, m$$

여기서  $A_i, B_i, C_i, D_i$ 는 퍼지 집합이고,  $\alpha_i$ 는 상수이다. 비퍼지화는 다음과 같다.

$$K_P' = \sum_{i=1}^m \mu_i K_{P, i}' \dots (12)$$

$$K_D' = \sum_{i=1}^m \mu_i K_{D, i}' \dots (13)$$

$$\alpha = \sum_{i=1}^m \mu_i \alpha_i \dots (14)$$

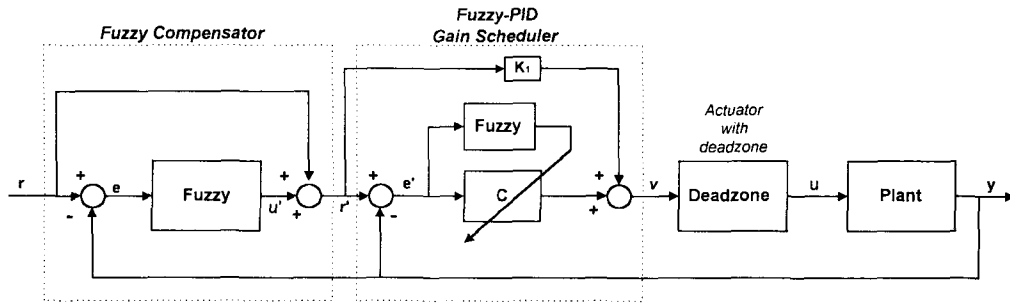


그림 4. 전체 구성도  
Fig. 4 Block diagram of the overall system

여기서,  $\mu_i = \mu_{A_i}[e(k)]$ ,  $\cdot \mu_{B_i}[\Delta e(k)]$ 인 소속 함수이다.

$K_p'$ ,  $K_D'$ ,  $\alpha$ 가 한번 얻어지면, PID 제어기 파라미터는 식 (9)와 식 (11)로부터 아래의 방정식으로 정리된다.

$$K_p = (K_{p, \max} - K_{p, \min})K_p' + K_{p, \min} \dots (15 a)$$

$$K_D = (K_{D, \max} - K_{D, \min})K_D' + K_{D, \min} \dots (15 b)$$

$$K_I = K_p^2 / (\alpha K_D) \dots (15 c)$$

여러 프로세서에 대하여 시뮬레이션한 것을 기초로 하여  $K_p$ 의 영역과  $K_D$ 의 영역을 정의하면 다음과 같다.

$$K_{p, \min} = 0.32K_u, K_{p, \max} = 0.6K_u$$

$$K_{D, \min} = 0.08K_u T_u, K_{D, \max} = 0.15K_u T_u$$

..... (16)

여기서  $K_u$ ,  $T_u$ 는 Ziegler-Nichols 방법에 의해 측정된 비례제어에 대한 안정 임계에서의 진동의 주기와 이득이다. Ziegler-Nichols 방법에 의해 측정된  $K_u = 0.8$ ,  $T_u = 0.5$ 로 결정되었다.

본 논문에서 사용된 퍼지 동조기의 역할은 세 변수  $K_p'$ ,  $K_D'$ 와  $\alpha$ 를 찾는 것이다. 예를 들어  $K_p'$ 에 대한 소속 함수와 퍼지 규칙을 그림 5와 표 2에 각각 나타내었다<sup>[6]</sup>.

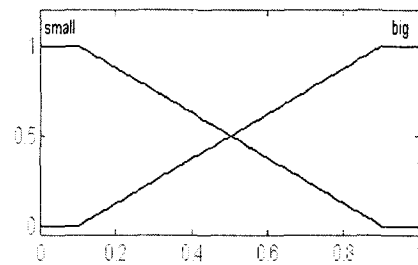


그림 5. 소속함수  $K_p'$ 의 출력  
Fig. 5 Output membership function  $K_p'$

표 2. Fuzzy tuning rules for  $K_p'$

Table 2. Fuzzy tuning rules for  $K_p'$

$e \backslash \Delta e$	NB	NM	NS	ZO	PS	PM	PB
NB	B	S	S	S	S	S	B
NM	B	B	S	S	S	B	B
NS	B	B	B	S	B	B	B
ZO	B	B	B	B	B	B	B
PS	B	B	B	S	B	B	B
PM	B	B	S	S	S	B	B
PB	B	S	S	S	S	S	B

#### IV. 전체 제어기

본 논문에서 사용된 전체 제어기의 구성도는 그

림 4와 같다. 여기서, 첫째 블록은 퍼지논리를 이용한 보상기이며, 두 번째 블록은 계수 동조기를 추가한 PID 제어기이다. PID 제어기 입력으로  $e(k)$ 와  $\Delta e(k)$ 를 대신하여  $e'(k) = e(k) + \mu(k)$ 와  $\Delta e'(k) = e'(k) - e'(k-1)$ 를 사용하며,  $u'(k)$ 는 퍼지 보상기의 출력이다. 특히 앞에서 언급한 특성 방정식과 같이 PID 제어기의 출력은 다음과 같다.

$$u(k) = K_I r'(k) + C[e'(k), \Delta e'(k)] \dots\dots\dots (17)$$

여기서,  $r'(k) = r(k) + u'(k)$ 이다.

V. 시뮬레이션

식 (18)과 같은 플랜트를 도입하고, 시뮬레이션을 통하여 이를 PID 제어기, 퍼지 보상기를 이용한 PID 제어기와 본 논문에서 제안한 퍼지 보상기와 퍼지 동조기를 이용한 PID 제어기의 출력 응답을 서로 비교 평가하였다.

$$P(k) = 0.005u(k-1) + 0.005u(k-2) + 1.9y(k-1) - 0.9y(k-2) \dots\dots\dots (18)$$

샘플링 시간은 0.01초를 사용하였고, 퍼지 논리 보상기의 스케일 계수는 다음의 값이 사용되었다.

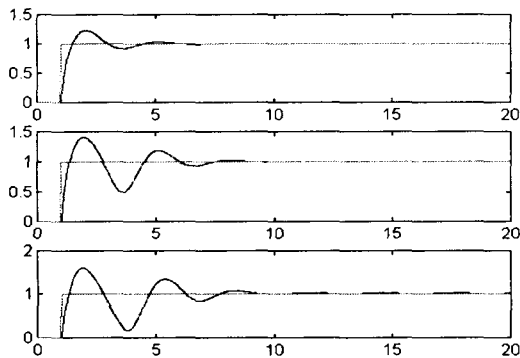


그림 6. 전형적인 PID 제어기의 출력응답  
( $d=0, 0.5, 1$ 이고,  $s=1$ )

Fig. 6 Output response of conventional PID controller

$$C_e = 4.5/r, C_{\Delta e} = 49.5/r, C_F = 0.2r$$

또한,  $r = 1, K_I = 0.1$ 로 각각 설정하였다.

그림 6, 7, 8, 9는 Deadzone을 가지고 있는 시스템을 각기 다른 제어기를 사용하여 얻어 낸 출력 응답이다.

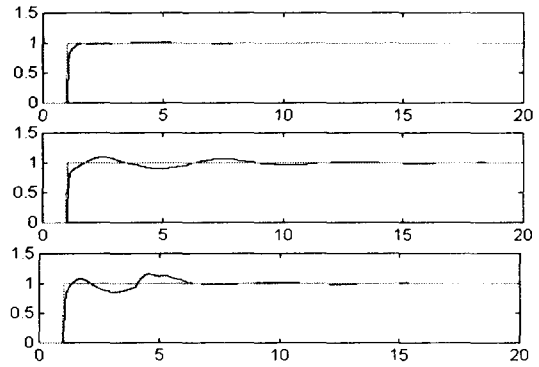


그림 7. 퍼지보상기를 이용한 PID 제어기 출력응답  
( $d=0, 0.5, 1$ 이고,  $s=1$ )

Fig. 7 Output response of PID controller using fuzzy compensator

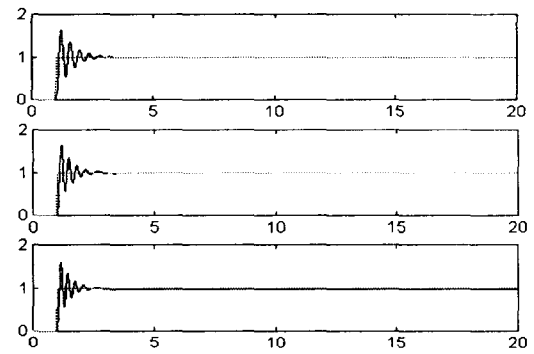


그림 8. 퍼지 보상기 및 퍼지 동조기를 이용한 PID 제어기의 출력응답  
( $d=0, 0.5, 1$ 이고,  $s=1$ )

Fig. 8 Output response of PID controller using fuzzy compensator and fuzzy tuner

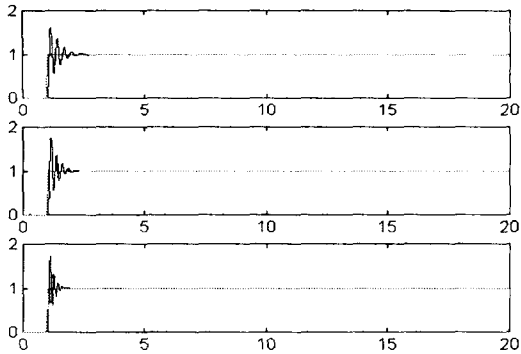


그림 9. 퍼지 보상기 및 퍼지 동조기를 이용한 PID 제어기의 출력응답  
( $d=0.5$ 이고,  $s=2, 3, 6$ )

Fig. 9 Output response of PID controller using fuzzy compensator and fuzzy tuner

### VI. 결 과

본 논문에서는 Deadzone을 포함하는 시스템에서 나타나는 문제점을 해결하고자, 퍼지 보상기와 퍼지 동조기를 이용한 PID 제어기를 제안하였고, 이 제어기의 제어 성능을 평가하였다.

제안된 제어기는 전형적인 PID 제어기와 비교하여 우수한 과도응답과 정상상태 성능을 가지고 있다는 것을 알 수 있었으며, 계수 값들도 비교적 안정되게 추정할 수 있었다. 이러한 결과는 먼저 오차 값이 퍼지 논리 보상기에서 제거되었고, 다시 PID 제어기의 계수 값을 자동적으로 조정함으로써 가능하였다. 더욱이, 제안된 제어기는 플랜트의 정상 상태 이득뿐만 아니라, Deadzone의 폭과 기울기의 변화에도 강인한 성능을 나타내었다.

### 참고문헌

[1] D. A. Recker, P. V. Kokotovic, D. Rhode and J. Winkelman, "Adaptive nonlinear control of systems containing a deadzone", in Proc. of the IEEE Conf. on Dec. and Contr., pp. 211 1~2115, Brighton, U. K., Dec. 1991  
[2] C. A. Desoer and S. M. Shahruz, "Stability of dithered nonlinear systems with backlash or

hysteresis", Int. J. Control, vol, 43, no. 4, pp. 1045~1060, 1986  
[3] Zhen-Yu Zhao, Masayoshi Tomizuka, "Fuzzy Gain Scheduling of PID Controller", IEEE Transactions on Systems Vol 23, No. 5, pp. 1392 -1398  
[4] T. Iwasaki and A. Morita, "Fuzzy auto-tuning for PID controller with model classification." in Proc. NAFIPS '90 Toronto, Canada, June 6-8, 1990, pp. 90-93.  
[5] J. H. Kim, J. H. Park, and S. W. Lee, "Control of Systems with Deadzones usings PD Controllers with Fuzzy Precompensation", proc. of the IEEE Conf. Intelligent Control, pp.451-456 1993 August.  
[6] 임영도, 김승철, 박재형, "Fuzzy-PID 제어기를 이용한 차량의 정속주행 시뮬레이션", 한국해양정보통신학회논문지, 제2권 제4호, pp603-610, 1988  
[7] Y. F. Li and C. C. Lan, "Development of fuzzy algorithms for servo systems", IEEE Contr. Syst. Magazine, vol. 9, pp. 65~72, 1980.



박 재 형(Jae-Hyung Park)

1987년 동아대학교 전자공학과 졸업(학사)

1990년 동아대학교 대학원 전자공학과 졸업(석사)

1999년 동아대학교 대학원 전자공학과(박사수료)

\* 관심분야 : 신경망 및 퍼지제어, 유전알고리즘, 로보틱스



김 승 철(Seung-Chul Kim)

1995년 동아대학교 전자공학과 졸업(학사)

1997년 동아대학교 대학원 전자공학과 졸업(석사)

1997년~현재 동아대학교 대학원 전자공학과(박사과정)

\* 관심분야 : 퍼지제어, 유전알고리즘, 진화프로그래밍



조 옹 성(Yong-Seong Cho)  
1988년 부경대학교 전자공학과  
졸업(학사)  
1990년 동아대학교 대학원 전  
자공학과(석사)  
1997년~현재 동아대학교 대학  
원 전자공학과(박사과정)

\* 관심분야 : 퍼지, 신경망, 카오스, 유도 가열기



최 부 귀(Boo-Kwi Choi)  
1960년 연세대학교 전기공학과  
졸업(학사)  
1980년 경희대학교 대학원 전  
자공학과 졸업(석사)  
1970년~현재 동아대학교 전자  
공학과 교수

\* 관심분야 : 지능형제어, 최적제어