

퍼지 리셋기능을 갖는 PI형 퍼지제어기

(A PI-Type Fuzzy Controller Having Fuzzy Resetting Capability)

李志弘*, 蔡昌鉉*, 張点煥**

(Jihong Lee, Chang Hyun Chai and Jum Hwan Chang)

要約

2차 이상의 제어대상 시스템에 대한 PI형 퍼지제어기의 단점을 보완하기 위해서 제어입력을 오차와 오차의 변화율에 의한 퍼지규칙에 의해 리셋 또는 증폭시키는 퍼지제어기의 한 형태를 제안한다. 이는 PI형 퍼지제어기가 정상상태오차의 제거에는 우수하지만 내부의 적분기의 특징 때문에 과도한 제어입력이 누적되어 불필요한 진동을 유발하는 경우가 많다. 따라서, 설정위치의 접근 상태에 따라 누적된 제어입력을 소거하거나 증폭시키는 퍼지제어규칙을 제어입력의 증분을 계산하는 부문에 추가적으로 도입하여 응답이 관성을 가지는 2차 이상의 시스템에도 적용될 수 있도록 구성하였다. 제안된 방법은 기존의 PI형 퍼지제어기로는 안정화하기 힘든 시스템과 성능개선에 어려움이 있는 시스템등에 시뮬레이션되어 그 유용성을 보였다.

Abstract

To improve the limitation of fuzzy PI controller when is applied to systems of order higher than one, a fuzzy PI controller that fuzzily resets or amplifies the accumulated control input according to fuzzy rules defined on (error, change of error) space is proposed. The proposed controller structure was motivated by the characteristics of fuzzy PI controller that it generally gives unevitable large overshoot in trial of reducing rise time of response especially when a system of order higher than one is considered. Based on the observation that the undesirable characteristics of the fuzzy PI controller is caused by integrating control input excessively, even though the integrator is introduced to overcome steady state error, we propose a controller that clear out or doubles integrated control input in a fuzzy manner according to the situation to reduce rise time as well as overshoot. To show the usefulness of the proposed controller, it is applied to the systems that are difficult to stabilize or difficult to get satisfactory response by conventional fuzzy PI controllers.

*正會員, 金烏工科大學校 電子制御工學科
(Dept. of Control Eng., Kumoh Nat'l
Univ. of Tech.)

**正會員, 金烏工科大學校 電子工學科

(Dept. of Elec. Eng., Kumoh Nat'l
Univ. of Tech.)

接受日字 : 1992年 10月 15日

1. 서론.

Zadeh 등^[1,2]에 의해 제안된 퍼지이론이 Mamdani 등^[3,4]에 의해 제어에 도입되면서 퍼지제어는 퍼지이론의 응용에 관한 가장 활발한 분야로 부상되고 있으며^[5], 또한 많은 이론적 진전이 있었다.^[6,7] 퍼지제어는 입출력 변수의 수가 아주 많아 기존의 제어 방법으로 다루기 곤란한 경우, 또는 시스템은 간단하지만 기존의 선형적 제어 방법에 의한 성능향상에 한계가 있는 경우에 적합하다. 전자의 경우는 대상 시스템에 대한 여러 정보가 전문가로부터 언어적 정보로 표현 가능하면 쉽게 퍼지제어기로 변환될 수 있다는 퍼지제어기가 갖고 있는 고유한 특성 때문이며, 후자의 경우는 퍼지제어기의 성능에 관계하는 퍼지화 방법, 추론규칙, 비퍼지화 방법, 그리고 언어적으로 묘사되는 각 변수의 모양 등에 의해 쉽게 비선형적인 제어를설계할 수 있기 때문이다. 본 논문에서는 기존의 선형적 제어 방법에 의한 성능향상에 한계가 있는 간단한 제어대상 시스템에 대한 퍼지제어기를 설계하는 문제를 다루고 있다.

현재 해석적인 연구가 많이 이루어지고 있는 퍼지제어기의 형태는 위치형 퍼지제어기(position type fuzzy controller : $(e, \Delta e) \rightarrow u$)와 속도형 퍼지제어기(velocity type fuzzy controller : $(e, \Delta e) \rightarrow \Delta u$)이다. 이들은 기존의 제어기 형태와 비교되어 의미상 퍼지 PD 제어기 및 퍼지 PI제어기라고도 불리는데, 전자는 측정된 시스템 출력으로부터 제어입력 자체를 구하는 형태이고, 후자는 제어입력의 증분을 구하는 형태이다. 퍼지제어기가 사람이 일반적으로 사용하는 제어규칙을 근간으로 한다는 측면에서 보면 사람의 경우 절대량의 감지 보다는 상대적인 양을 감지하고 있으며, 예를들어, 자동차 운전중의 경우를 생각해 보면 사람은 "속도를 80 Km/h로", "속도를 40 Km/h로" 보다는 "지금보다 조금 더 빨리", "지금보다 조금 더 느리게" 등의 방식으로 제어를 한다고 믿어지므로 후자의 형태가 보통의 경우 더욱 실제적일 뿐 아니라 부하, 제어대상의 특성 등을 잘 모르는 상황에서 정상상태의 오차를 없앨 수 있으므로 더 타당한 구조라고 볼 수 있다.

그러나, 속도형 퍼지제어기는 정상 상태의 오차를 줄이는 능력은 탁월하지만 2차 이상의 시스템인 경우에는과도 응답을 개선하기가 어려울 때가 많다.^[13] 2차 이상의 시스템은 물리적으로 신호에 관성을 갖고 있어 PI 형 퍼지제어기에 의해 진동이 유발될 때가 많다. 따라서 퍼지제어기 설계를 제안한 많은 논문^[8-11]들이 1차 시스템에 대해 시뮬레이션한 결과를 보고

하고 있으며, 2차 이상의 경우에도 시스템의 시정수보다 천천히 변화하도록 증분의 최대값의 크기를 제한하고 있음을 알 수 있다.^[13] 이러한 현상의 이유는 보통의 PI 제어기에서 이득 결정이 쉽지 않다는 사실에서 유추가능하며 이러한 문제의 해결을 위해서는 어떤 추가적인 정보의 사용이 불가피하다. 추가적인 정보로는 가속도와 관계하는 \ddot{e} 항을 생각할 수 있는데 일반적인 사람의 경우 \ddot{e} 의 값을 감지하여 제어에 이용하고 있다고 보기는 어렵다. 또 \ddot{e} 의 값 측정장비가 고가이거나 측정 자체가 불가능한 경우가 있어 \ddot{e} 이나 Δe 를 이용하여 간접적으로 구할 수 있으나 이전의 정보를 필요로 한다는 단점이 있다.

본 논문에서는, 기존의 PI형 퍼지제어기의 구조적인 문제점을 해결하고자 새로운 형태의 제어기를 제안한다. 제안된 제어기의 작용은 크게 두 가지로 나눌 수 있다. 하나는 "오차는 큰데 오차의 변화율이 작음" 경우에 상승시간이 빠르게 되도록 누적되는 제어입력을 증폭하는 작용이고, 다른 하나는 "오차는 작는데 오차의 변화율이 큼" 경우에 과도한 제어입력이 누적되면 응답의 진동폭이 커지므로 누적된 제어입력을 제거하는 작용이다. 제안된 제어기를 2차 이상의 플랜트에 시뮬레이션해 본 결과, 기존의 PI형 퍼지제어기로는 안정화하기 힘든 시스템의 경우나 응답의 개선이 어려운 시스템의 경우에도 우수한 성능을 얻었다. 또한 새로 추가된 규칙은, 기존의 PI형 퍼지제어기에서 Δu 를 계산하는 제어규칙과 동일한 퍼지 변수 및 퍼지 대집합등을 사용하여 추가되는 부분을 최소화하였다.

2장에서 PI형 퍼지제어기가 가지고 있는 문제점에 대해 분석하고, 3장에서는 새로운 형태의 제어기를 소개하고 시뮬레이션 결과를 보이며, 4장에서는 결론을 맺는다.

II. 문제설정

PI형 퍼지제어기의 k번째 제어 규칙 R_k 는 식(1)로 기술되고 전체적인 구조는 그림1과 같다.

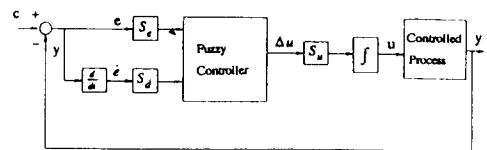


그림 1. PI형 퍼지제어기의 구성도

Fig. 1. The block diagram of a PI type fuzzy controller.

$$R_k : \text{If } e \text{ is } A_k \text{ and } \dot{e} \text{ is } B_k, \text{ then } \Delta u \text{ is } C_k \quad (1)$$

여기서 A_k, B_k, C_k 는 각각 제어오차, 오차의 변화율, 제어입력공간에서 정의된 퍼지집합이다. PI형 퍼지제어기는 제어입력을 누적시켜서 최종입력을 결정하므로 정상상태오차가 없도록 제어가 가능하며 부하의 변동에도 대처할 수 있는 반면, 적절한 상승시간과 오버슈트를 주는 조건을 찾기가 매우 어렵다. 보통의 퍼지제어기를 설계하는 경우에는 제어대상 시스템에 대한 완벽한 정보가 알려져 있지 않은 경우가 보통이므로 적절한 퍼지 집합이나 제어규칙들을 찾는 일이 더욱 어렵다. 이러한 현상을 SISO 1차 시스템을 가정하여 분석한다.

먼저 제어기의 구조를 살펴보자. 그림1의 $S_e, S_{\dot{e}}$ 와 S_u 는 실질적으로 입력력 변수의 퍼지 분할에 관계하는 스케일링 계수(scaling factor)로 제어성과 밀접한 관계가 있으므로 전문가의 지식에 의해 어느 정도 범위의 값으로 정해지며, 추후 시스템의운전을 통해 조정하거나 또는 자체 조정 방법 등에 의해 개선될 수 있다. 스케일링 계수를 사용하면 입력력 변수의 퍼지 분할에 어떤 표준적인 형태를 사용할 수 있다는 장점이 있다. PI형 퍼지제어기의 분석에 이용한 비퍼지화(fuzzification)는 구현이 쉬운 단순화된 형태로 다음 식(2)로 기술된다. 먼저 측정된 오차 및 오차의 변화율에 대한 k번째 규칙과의 적합도를 식(2a)로 구한다

$$w_k = \min(\mu_{A_k}(e), \mu_{B_k}(\dot{e})) \quad (2a)$$

k번째 규칙에 의한 입력증분의 퍼지집합을 나타내는 퍼지집합을 식(2b)로 구하면.

$$w_{C_k}(u) = \min(w_k, \mu_{C_k}(u)) \quad (2b)$$

퍼지집합으로 부터 무게중심법을 구하는 연산자 COG(center of gravity)에 의해 k번째 제어입력의 증분 Δu_k 가 식(2c)에 의해 구해지고

$$\Delta u_k = COG(C_k') \quad (2c)$$

최종적으로 적분기에 가산될 입력증분은 식(2d)로 계산된다.

$$\Delta u' = \frac{\sum_{k=1}^N w_k \Delta u_k}{\sum_{k=1}^N w_k} \quad (2d)$$

이상으로 기술된 방법은 원래의 무게중심법이나 최대치 평균방법에 비해 계산이 간편하고 연속 퍼지집합을 쉽게 다룰 수 있다. 두 개의 규칙 R1, R2인 경우와 계산과정을 그림2에 보인다.

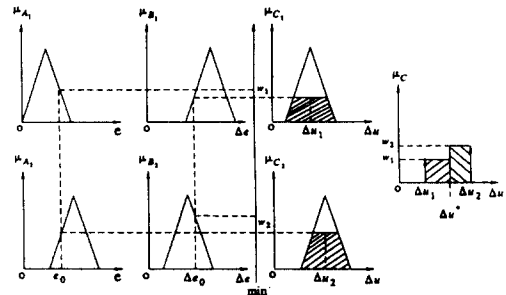


그림 2. 그림으로 나타낸 식(2)의 계산 과정
Fig. 2. Graphical representation of the calculation of eq. (2).

$$R_1 : \text{If } e \text{ is } A_1 \text{ and } \dot{e} \text{ is } B_1, \text{ then } \Delta u \text{ is } C_1 \quad (3a)$$

$$R_2 : \text{If } e \text{ is } A_2 \text{ and } \dot{e} \text{ is } B_2, \text{ then } \Delta u \text{ is } C_2 \quad (3b)$$

그림2를 보면 각 규칙에서 결론부의 연속 퍼지집합은 전제부에서 계산된 규칙의 적합도로 높이를 제한시켜 만들어진 퍼지집합을 퍼지 싱글톤으로 등가화시킨 후, 각 규칙으로 부터 구해진 결과를 무게중심법을 적용시켜 최종 제어입력을 정한다. Δu^* 를 구하는 그림2에서 두 개의 직사각형의 면적이 같다는 점을 고려하면 이러한 계산 방법의 의미를 쉽게 알 수 있다.

$\dot{e} \ e$	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NS	NS	ZE
NM	NB	NM	NM	NM	NS	ZE	PS
NS	NB	NM	NS	NS	ZE	PS	PM
ZE	NB	NM	NS	ZE	PS	PM	PB
PS	NM	NS	ZE	PS	PS	PM	PB
PM	NS	ZE	PS	PM	PM	PM	PB
PB	ZE	PS	PS	PM	PB	PB	PB

그림 3. Δu 를 계산하기 위한 규칙
Fig. 3. Rule table for calculating of Δu .

제어성능에 관계하는 또 하나의 중요한 요소는 제어규칙이다. 전제부와 결론부의 퍼지집합이 정해지면 그 퍼지집합들 사이의 관계를 정해주는 제어 규칙은 전제부의 변수가 2개인 보통의 경우에는 그림3과 같은 테이블 형식의 표시가 가장 편리하다. 이 테이블은 일종의 위상공간으로 응답의 상태에 따라 그림4와 같이 8개의 공간으로 분류될 수 있다.

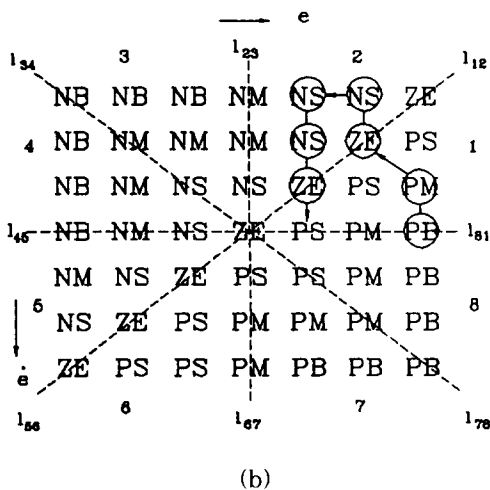
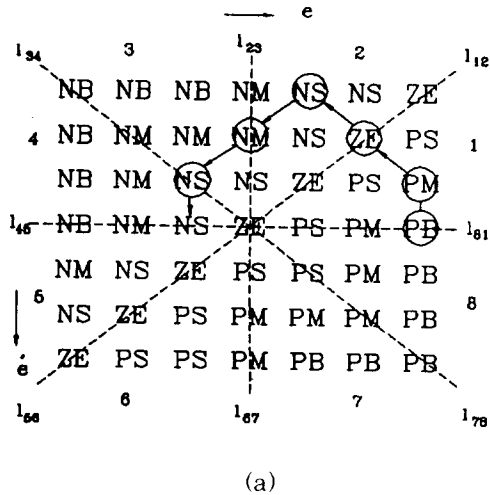


그림 4. 퍼지 위상 공간
Fig. 4. Fuzzy phase plane.

이 제어 규칙을 이용하여 계단 신호를 추적하는 경우를 고려하여 보자. 이 경우 제 1 공간은 가속구간이 되고 제 2 공간은 감속구간, 제 3 공간은 오버슈트 구간으로 응답에 가장 크게 영향을 미치는 영역들이

다. 계단형 신호의 추적에 대한 시스템의 응답을 이 위상공간의 궤적을 따라 분석하여 보자. 초기 위치에서 출발하여 제 1 구간에서 가속력을 얻어 제 2 공간으로 넘어가서 시스템의 특성과 선분 l_{12} 을 대칭축으로 한 제 1, 2 구간의 제어규칙의 대칭 정도에 따라 l_{23} 선분을 넘어 가거나 또는 넘어가지 않거나 원점으로 수렴한다. 이 과정에서 상승시간을 줄이려면 제 1 구간에서 충분한 제어 입력이 누적되도록 해야 하며 오버슈트를 줄이기 위해서는 이 누적된 제어량은 제 2 구간을 통과하는 동안 시스템에 따라 적당량이 제거 되도록 해야 한다.

그러나 그림 4의 선분 l_{12} 을 축으로 제 1, 제 2 구간의 제어 규칙이 대칭이라 하여도 제어 대상 시스템의 특성에 따라 각 퍼지 영역에 머무는 시간이 달라 정확히 상쇄되도록 하기가 어려울 뿐만 아니라, 상쇄되었다 하더라도 2차 이상의 시스템이 일반적으로 가지는 관성에 의해 선분 l_{23} 를 통과하거나 (그림4-(a)의 경우)또는 지나친 상쇄로 목표 값에 못미치는 경우(그림4-(b)의 경우)가 자주 발생한다. 따라서 PI형 퍼지 제어기로는 상승시간과 오버슈트를 동시에 만족시키기 어렵다는 결론이 나오며 이는 일반적인 PI 제어기의 경우와 직관적으로 동일하다. 물론 오버슈트를 줄이기 위해 시스템의 시정수를 고려하여 충분히 천천히 응답이 증가하는 방법이 가능하다. 이렇게 하기 위해선 제어입력의 증가현상을 주로 e 에 의해 충분히 작은 값으로 제한하여야 하는데, 즉 그림 1의 S_r 를 충분히 크게 해야 한다. 계수 S_r 의 변화에 따른 시스템 응답을 고찰하기 위해 PI형 퍼지제어기를 식(4)로 주어지는 2차 시스템에 적용해 보자. 시뮬레이션 결과로 응답은 그림5에, 위상공간상의 궤적은 그림6에 나타나 있다.

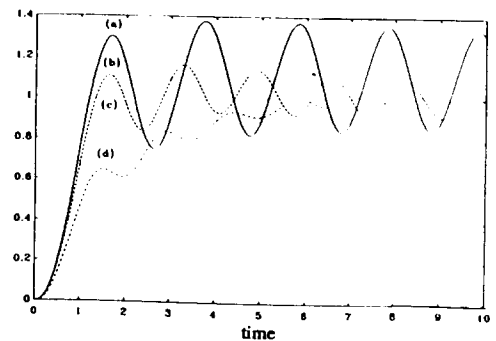


그림 5. 퍼지 PI 제어기에 의한 식(4)의 응답
Fig. 5. Response of eq(4) with fuzzy controller.

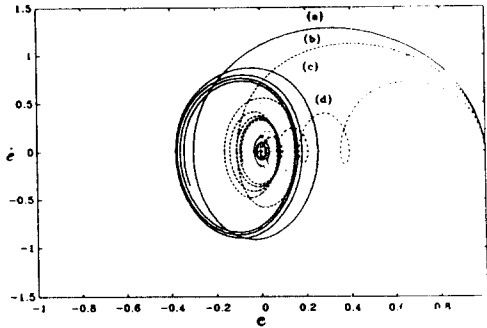


그림 6. (오차, 오차의 변화율) 공간에서 그림5의 궤적
Fig. 6. Trajectories of the responses of Fig. 5 (e, e) space.

$$G(s) = \frac{1}{s(s+1)} \quad (4)$$

- 그림 5.6에서 각 경우의 설계계수는
 (a) $S_e = 1.0 \ S_d = 1.5 \ S_u = 0.5$
 (b) $S_e = 1.0 \ S_d = 2.0 \ S_u = 0.5$
 (c) $S_e = 1.0 \ S_d = 3.0 \ S_u = 0.5$
 (d) $S_e = 0.8 \ S_d = 2.5 \ S_u = 0.5$

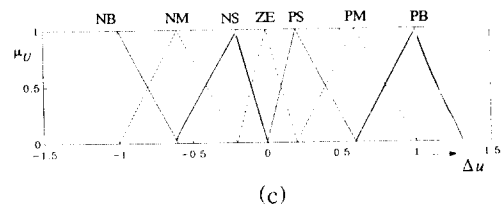
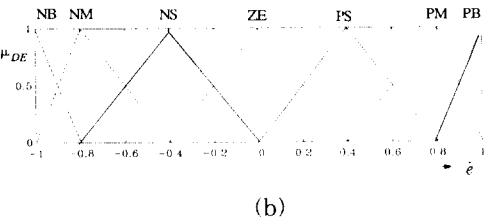
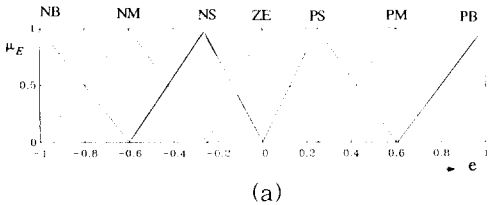


그림 7. 사용된 퍼지 변수들
Fig. 7. Fuzzy sets used in simulation.

이고, 그림3의 규칙과 그림7과 같은 비선형 퍼지 분할이 사용되었다.

즉, 계수를 조정하여 오버슈트를 작게 하면 상승시간이 현저하게 길어지고, 상승시간을 개선하려면 오버슈트가 커짐을 알 수 있다. 이러한 현상은 일단 가속 구간(그림 4의 제 2 구간)에서 누적되는 제어입력의 양을 적절히 조절해 주면 가능하다. 현재의 구조에서는 쉽지 않다. 이러한 결과는 PI형 제어기의 일반적인 현상으로, 입출력 변수의 퍼지 분할이나 제어 규칙의 조정으로는 개선에 한계가 있으므로 보조 정보를 이용하는 새로운 구조의 제어기가 필요하다. III장에서 제안될 제어기의 도입 동기를 스케일링 계수 (b)를 사용하여 시뮬레이션 응답과 제어입력 input 을 동시에 보인 그림8에서 고찰해보자

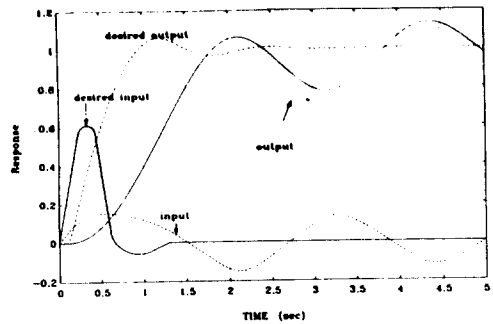


그림 8. 퍼지 PI 제어기에 의한 스케일링 계수 (b)의 응답
Fig. 8. Response of scaling factor (b) with fuzzy PI controller.

시뮬레이션 응답에서 초기에는 제어 입력이 작아서 상승시간이 길어지며, 설정치에 접근했을 때는 누적된 제어입력의 소거 속도가 늦어지므로 Δu 에 의한 증감으로는 원하는 응답(desired response)을 얻기가 어렵다는 것을 알 수 있다. 참고로 S_u 값을 크게해서 제어입력의 증감을 빠르게 하는 방법은 응답을 불안정하게 하는 요인이므로 적절하지 않다. 새로운 구조를 추가해서 그림과 같이 오차가 큰 부분에서는 누적되는 제어입력이 Δu 에 보조적으로 좀 더 커지게 하고 설정치에 어느 정도 접근했을 때는 누적된 제어입력을 Δu 에 보조적으로 빠르게 소거시키면 원하는 응답을 얻을 수 있다는 것이 본 제어기를 구성하게 된 동기이다.

III. 리셋 기능을 갖는 퍼지 PI 제어기

앞서 2장에서 언급한 바와 같이 PI형 퍼지제어기는

구조적으로 2차 이상의 시스템에 대해 적용되기에 문제점이 지적되므로 기존의 PI형 퍼지제어기의 구조를 개선한 퍼지 리셋(fuzzy reset) 기능을 갖는 퍼지제어기의 구조를 제안한다. 제안된 제어기의 작용은 크게 두 가지로 나눌 수 있다. 하나는 “오차는 큰데 오차의 변화율이 작을” 경우에 상승시간이 빠르게 되도록 누적되는 제어입력을 증폭하는 작용이고, 다른 하나는 “오차는 작는데 오차의 변화율이 큰” 경우에 과도한 제어입력이 누적되면 응답의 진동폭이 커지므로 누적된 제어입력을 제거하는 작용이다.

1. 모델

제어기의 전체적인 구조는 그림9와 같다.

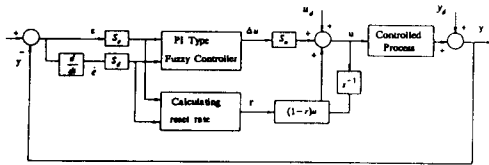


그림 9. 제안된 제어기의 블럭도

Fig. 9. Overall block diagram of proposed controller.

최종 제어입력은 그림9와 같은 과정을 거쳐서 구해 지는데, 먼저 다음의 각 제어 규칙으로 부터 구해지는 제어입력 증분 Δu_k 의 계산은 2장과 같다. 즉,

$$R_k : \text{If } e \text{ is } A_k \text{ and } \dot{e} \text{ is } B_k, \text{ then } \Delta u \text{ is } C_k \quad (5a)$$

$$w_k = \min(\mu_{A_k}(e), \mu_{B_k}(\dot{e})) \quad (5b)$$

$$\mu_{C_k}(u) = \min(w_k, \mu_{C_k}(u)) \quad (5c)$$

$$\Delta u_k = COG(C_k) \quad (5d)$$

에 의해서 w_k 및 Δu_k 가 k번째 규칙으로 부터 정해 지며 최종제어 입력 증가분은.

$$\Delta u^* = \frac{\sum_{k=1}^N w_k \Delta u_k}{\sum_{k=1}^N w_k} \quad (5e)$$

에 의해 구해진다.

다음으로 리셋율 r 을 구하는 과정은 (5)식의 과정 과 같다. 즉,

$$R_k : \text{If } e \text{ is } A_k \text{ and } \dot{e} \text{ is } B_k, \text{ then } r \text{ is } R_k \quad (6a)$$

$$w_k = \min(\mu_{A_k}(e), \mu_{B_k}(\dot{e})) \quad (6b)$$

$$\mu_{R_k}(u) = \min(w_k, \mu_{R_k}(u)) \quad (6c)$$

$$r_k = COG(R_k) \quad (6d)$$

이다. 여기서 r_k 는 k번째 규칙에 의한 리셋율 r 이다. w_k 및 r_k 가 k번째 규칙에서 정해지면 최종 리셋율은

$$r = \frac{\sum_{k=1}^N w_k r_k}{\sum_{k=1}^N w_k} \quad (6e)$$

로 구해진다. 리셋율 r 에 대한 영향을 고찰하면 (i+1) 번째의 제어시각의 제어입력은 식(7)과 같다.

$$u(i+1) = (1-r(i+1))u(i) + \Delta u(i+1) \quad (7)$$

식 (7)에서 r 이 0이면

$$u(i+1) = u(i) + \Delta u(i+1) \quad (8a)$$

로 보통의 적분과정과 완전히 같아지므로 PI형 퍼지 제어기의 형태가 된다. r 이 -1이면

$$u(i+1) = 2u(i) + \Delta u(i+1) \quad (8b)$$

로 제어입력이 기존의 u 값 보다 두배가 누적되고, r 이 1이면

$$u(i+1) = \Delta u(i+1) \quad (8c)$$

로 누적되는 제어입력이 없으므로 그 순간은 Δu 가 제어입력 u 가 되므로 의미상 퍼지 PD 제어기와 같아진다. 즉, 상황에 따라 r 의 값을 적절히 이용하므로써 두 가지 제어기의 장점을 취해 보자는 것이 주된 의도이다.

2. 제어기 동조과정

제어기 동조과정은 다음과 같다.

적절한 상승시간과 오버슈트를 주는 최적의 조건을 찾는 계수 설정방법은 어떤 일정하게 정해진 방법으로 최적의 계수를 결정하기 보다는 여러번의 시행착오에 의하여 결정한다. 이는 최적제어시 계수를 설정하기 위한 정해진 방법이 없기 때문이다.

1) 일반 퍼지제어기의 스케일링 계수를 잘 조정해서 안정된 응답을 얻을 때까지 조정한다. 여기서 안정된 응답이란 설정치에 수렴하는 것을 의미하는 것이 아니고 적당한 오버슈트를 가지며 발산하지 않는 응

답을 뜻한다.

2) 지능형 적분기능 작용을 도입해서 1)의 과정에서 구해진 스케일링 계수를 가지고 설정치에 수렴하는 응답을 얻도록 스케일링 계수를 재 조정한다. 리셋의 작용으로 누적되는 제어입력 u 값이 변동되므로 스케일링 계수의 재조정이 필요하다. 즉, 상승시간을 짧게 하려면 S_u 값을 크게 하여야 하며, 오버슈트를 동반하는 응답이라면 S_u 값을 크게 해서 예진기능을 증가시키고, 긴 상승시간을 가지며 설정치 밑에서 수렴하는 응답이라면 S_u 값을 작게 해서 예진기능을 다소 미약하게 한다.

3. 제어기 동조과정 적용 예

위에서 정리한 동조과정은 여러 형태의 시스템에 적용하여 보자. 첫번째 동조과정에서 PI형 퍼지제어기를 이용하여 발산하지 않는 안정한 응답을 얻도록 동조하고, 이를 토대로 두번째 동조과정에서는 플랜트 출력을 -0.3만큼 변화시키는 교란입력 y_d 가 인가된 경우, 플랜트 입력에 -1만큼 부하외란 u_d 가 인가된 경우로 구분하여 설명하겠다.

1). 1형 계통의 2차 시스템

제어대상 플랜트는 식(9)와 같다.

$$G_1(s) = \frac{1}{s(s+1)} \tag{9}$$

첫번째 동조과정에서 PI형 퍼지제어기의 스케일링 계수를 조정하여 안정된 응답을 얻는다. 여기서 Δu 를 계산하는 제어규칙과 e, \dot{e}, u 에 대한 각 퍼지 집합은 그림3, 그림7과 같다. 각 스케일링계수를 ($S_e, S_{\dot{e}}, S_u$)=(1.1, 8.0, 1)로 했을 때 오버슈트는 20% 이하이고 발산하지 않는 그림10와 같은 출력 응답을 얻을 수 있다.

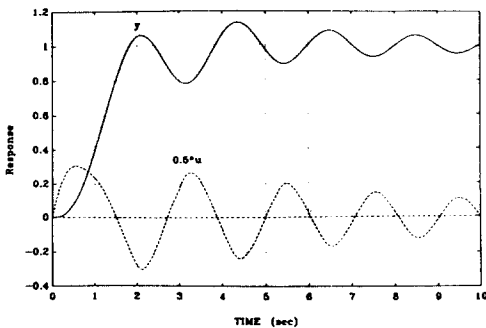


그림 10. 퍼지 PI 제어기에 의한 식(9)의 응답
Fig. 10. Response of eq(9) with fuzzy PI controller.

두번째 동조과정에 따라 지능형 적분기능 작용을 도입하여 설정치에 수렴하는 출력응답을 얻는다. 여기서 e, \dot{e} 에 대한 퍼지집합은 그림7과 같고, 리셋율을 계산하기 위한 퍼지집합과 제어규칙은 그림11, 그림12와 같다.

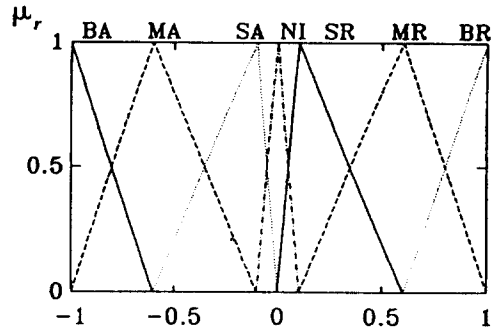


그림 11. 리셋율의 퍼지 변수들
Fig. 11. Fuzzy sets for reset rate r.

$\dot{e} \setminus e$	NB	NM	NS	ZE	PS	PM	PB
NB	SR	MR	BR	BR	BR	MR	SR
NM	NI	NI	SR	MR	SR	NI	NI
NS	NI	NI	NI	SR	NI	NI	MA
ZE	BA	MA	NI	SR	NI	MA	BA
PS	MA	NI	NI	SR	NI	NI	NI
PM	NI	NI	SR	MR	SR	NI	NI
PB	SR	MR	BR	BR	BR	MR	SR

그림 12. 식(9)의 시스템에 대한 reset 율을 계산하기 위한 규칙
Fig. 12. Fuzzy rules table calculating reset rate for eq. (9).

단, 그림 11,12에서 사용된 각 퍼지집합의 이름은 다음과 같다.

- BA : Big Amplification
- MA : Medium Amplification
- SA : Small Amplification

NI : Normal Integration
 SR : Small Reset
 MR : Medium Reset
 BR : Big Reset

그림12를 재구성한 그림13에서 리셋을 r을 추론하는 규칙 테이블의 각 부분의 역할을 조사하자. 그림13에서 위상공간이 갖는 정보를 이용하여 각 부분의 의미를 보면, e 는 크고 \dot{e} 이 작은 (a)영역에서는 누적되는 제어입력을 증가시켜 주어서 상승시간을 빠르게 하며, e 는 작아지고 \dot{e} 이 커지는(b)영역은 (a)영역에서 누적된 제어입력이 과도하므로 응답이 과도한 변화율로 설정치를 통과하지 않도록 미리 쌓인 제어입력을 제거시켜 나가는 영역이며, 그리고 e 가 작고 \dot{e} 이 큰(c)영역은 설정치를 큰 기울기로 통과하는 경우에 대해 제동을 걸기 위한 영역이다.

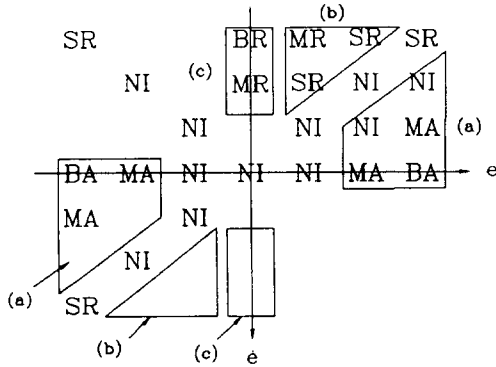


그림 13. (오차, 오차의 변화율) 공간에서의 각 영역별 해석
 Fig. 13. Role of the parts of (e, \dot{e}) space.

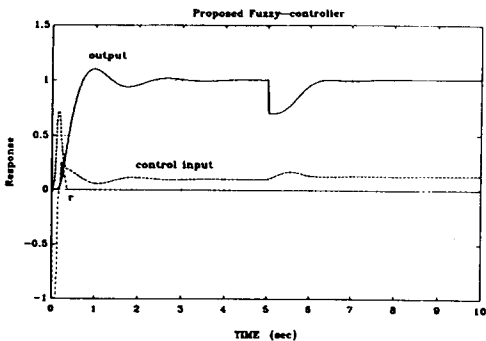


그림 14. 제안된 제어기에 의한 식(9)의 응답
 Fig. 14. Response of eq. (9) using the proposed controller.

재조정된 스케일링 계수는 $(S_e, S_{\dot{e}}, S_u)=(1.0, 0.6, 0.8)$ 이고 5초 이후의 플랜트 출력응답을 -0.3만큼 변화시키는 교란입력 y_d 가 인가되었을 때 출력응답은 그림14와 같다. PI형 퍼지제어기를 사용한 그림10의 경우보다 상승시간과 오버슈트 현저하게 개선되었음을 알 수 있다. 동조과정에서 설명한 것처럼 제안된 제어기는 PI형 퍼지제어기의 제어입력 u 값 보다도 더 큰 u 값이 인가되어도 적절한 때의 리셋의 작용으로 큰 오버슈트없이 설정치에 수렴하고, 출력에 인가된 교란입력에도 비교적 강인하게 적응함을 알 수 있다. 즉, 오차가 클때는 제어입력의 잔류량이 증폭되어 제어입력이 커지고, 오차가 작아지는데도 제어입력이 큰값을 유지하게 되면 제어입력의 잔류량이 제거되어 제어 입력이 작아지므로 응답이 안정하게 설정치에 도달하게 된다.

(1) 부하외란 u_d 가 인가된 경우

두번째 동조과정에서 리셋율을 계산하기 위한 퍼지 집합은 3-1절에서 그림11에 보인 형태이나 제어규칙은 그림15이다.

$\dot{e} \backslash e$	NB	NM	NS	ZE	PS	PM	PB
NB	SR	MR	BR	BR	BR	MR	SR
NM	NI	NI	SR	MR	SR	NI	NI
NS	NI	NI	NI	SR	NI	NI	MA
ZE	BA	MA	NI	NI	NI	MA	BA
PS	MA	NI	NI	SR	NI	NI	NI
PM	NI	NI	SR	MR	SR	NI	NI
PB	SR	MR	BR	BR	BR	MR	SR

그림 15. 식(9)의 시스템에 대한 reset 율을 계산하기 위한 규칙

Fig. 15. Fuzzy rules table calculating reset rate for eq. (9).

3-1절에서 $e=ZE$ 이고 $\dot{e}=ZE$ 이면 리셋율은 SR이라는 항이 있어서 플랜트 출력응답은 감쇠(damping)된다. 교란입력이 플랜트 입력에 인가된 상황에서는 리셋율 SR의 감쇠 효과가 부하외란에 대응하는 증분 제어입력을 일정한 크기로 제한하고 있어서 효율적이지 못하기 때문에, $e=ZE$ 이고 $\dot{e}=ZE$ 일때 리셋율은 NI로 해서 충분한 제어입력이 누적되도록 하는 것이

타당하다. 재조정된 스케일링계수는 $(S_e, S_d, S_u) = (1.0, 0.7, 1.4)$ 이고 5초 이후의 부하외란 $u_F=1$ 로 한 출력응답은 그림16과 같이 얻을 수 있다.

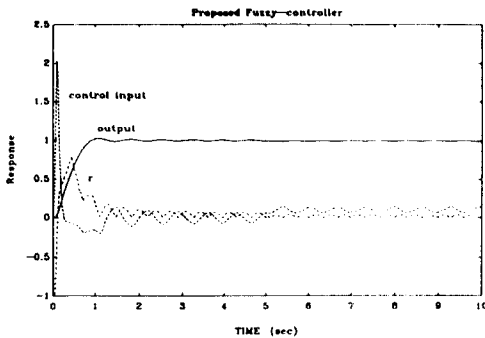


그림 16. 제안된 제어기에 의한 식(9)의 응답
Fig. 16. Response of eq. (9) using the proposed controller.

결과를 보면 설정치 부근에서 1% 미만의 진동은 있지만 부하외란에 대응하는 제어입력이 제한없이 누적되므로 부하외란에 상당히 강인함을 알 수 있다.

2) 0형 계통의 4차 시스템
제어대상 플랜트는 다음과 같다.

$$G_3(s) = \frac{10000}{(s+1)(s+5)(s+20)(s+100)} \quad (10)$$

첫번째 동조과정에서 모든 제어규칙과 퍼지변수는 3-1절에서 2차 시스템에 적용한 것과 같으며 각 스케일링계수를 $(S_e, S_d, S_u)=(1.0, 0.6, 0.1)$ 로 했을 때 그림17과 같은 출력응답을 얻을 수 있다

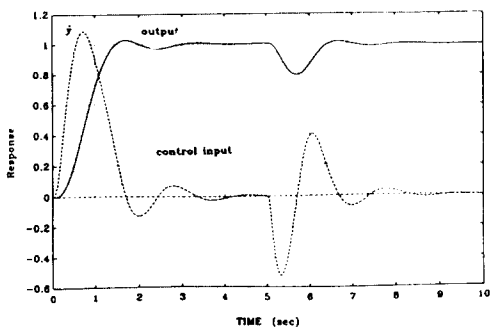


그림 17. 퍼지 PI 제어기에 의한 식(10)의 응답
Fig. 17. Response of eq. (10) with fuzzy PI controller.

결과를 보면 0형 계통의 플랜트는 적분기를 포함하지 않으므로 PI형 퍼지제어기로도 좋은 출력응답을 얻을 수 있다는 것을 알 수 있다. 또한 5초 이후 플랜트 입력에 인가되는 부하외란에도 강인한 특성을 보여주고 있다

두번째 동조과정에서는 상승시간만 빠르게 한다는 목적만으로도 좋은 출력응답을 얻을 수 있으므로 리셋율을 계산하기 위한 제어규칙은 그림 18처럼 수정해서 적용한다.

$e \backslash e$	NB	NM	NS	ZE	PS	PM	PB
NB	NI	NI	NI	NI	NI	NI	SR
NM	NI	NI	NI	NI	NI	NI	NI
NS	NI	NI	NI	NI	NI	NI	NI
ZE	BA	NI	NI	NI	NI	NI	BA
PS	NI	NI	NI	NI	NI	NI	NI
PM	NI	NI	NI	NI	NI	NI	NI
PB	SR	NI	NI	NI	NI	NI	NI

그림 18. 식(10)의 시스템에 대한 reset율을 계산하기 위한 규칙

Fig. 18. Fuzzy rules table calculating reset rate for eq. (10).

여기서 그림18과 같은 제어규칙의 의미는 오차가 큰 부분에서는 제어입력의 잔류량 증폭작용이 되어서 큰 제어입력이 쌓이도록 유도하다가 제어입력이 큰

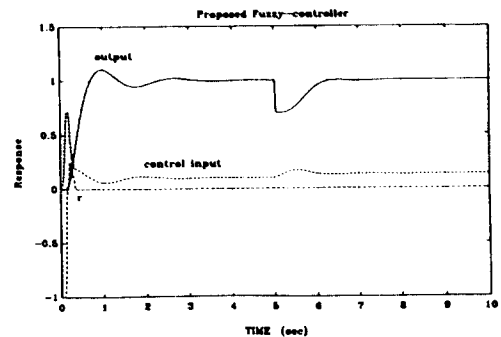


그림 19. 제안된 제어기에 의한 식(1)의 응답
Fig. 19. Response of eq. (1) using the proposed controller.

값을 유지하게 되고 ϵ 의 변화율이 커지면 제어입력의 잔류량 제거작용을 하게 해서 상승시간만을 빠르게 하는 효과를 얻고자 하는 것이다. 첫번째 동조과정에서 쓰인 것과 같은 스케일링계수를 사용해서 얻은 출력응답은 그림19와 같다

결과를 보면 리셋의 작용이 초반에 빠른 상승시간을 만들도록 설계했던 제어규칙처럼 좋은 출력응답이 얻어짐을 알 수 있다.

(1). 부하외란 u_d 가 인가된 경우

두번째 동조과정은 3-1절에서 사용된 퍼지집합과 제어규칙을 그대로 사용하고 재조정된 스케일링계수는 $(S_e, S_r, S_u)=(1.0, 0.65, 0.1)$ 이고 출력응답은 그림20과 같이 얻을 수 있다

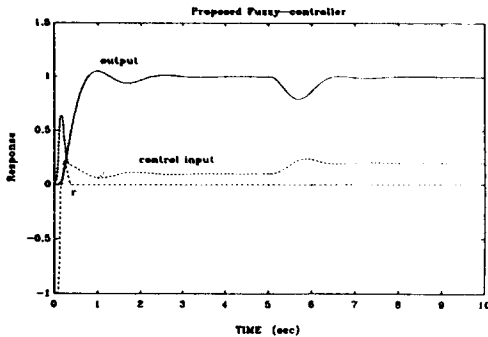


그림 20. 제안된 제어기에 의한 식(10)의 응답
Fig. 20. Response of eq(10) using the proposed controller.

결과를 보면 3-2절에서 0형계통의 플랜트에서는 PI형 퍼지제어기로도 플랜트 입력에 인가된 부하외란에 강인하다는 것을 보였으므로 제안된 제어기를 사용하여 상승시간을 상당히 감소시킬 수 있다.

IV. 결론

본 논문에서는 PI형 퍼지제어기가 2차 이상의 시스템에 적용될 때 구조적 문제점이 지적되므로 기존의 퍼지제어기의 구조를 개선한 퍼지 리셋기능을 갖는 PI형 퍼지제어기의 구조와 동조과정을 제안했다. 기존의 제어기는 제어입력을 누적시켜서 최종입력을 결정하므로 적절한 상승시간과 오버슈트를 주는 최적의 조건을 찾기가 어렵다. 따라서 설정치의 접근상태에 따라 "오차는 큰데 오차의 변화율이 작다면", 누적되는 제어입력이 증가하는데 도움을 주어서 상승시간을 빠르게 한다. 그리고 "오차는 작는데 오차의 변화율

이 크다면", 큰 값으로 누적된 제어량을 감소시켜 응답의 진동폭이 커지게 되는 것을 방지한다. 기존의 제어기로는 안정시키기 힘든 시스템의 경우나, 응답의 개선에 한계가 분명한 시스템의 경우에도 우수한 특성이 나왔음을 시뮬레이션으로 입증하였다.

그러나, 교란입력 u_d 가 플랜트 입력에 인가된 상황에서는 1% 미만의 진동이 생기므로 교란입력과 리셋율의 관계를 명확히 규명해야 하는 연구가 필요하고, 최적의 퍼지제어기를 설계하기 위한 이득이나 계수 등의 조정 알고리즘에 대한 지속적인 연구가 필요하다. 그리고 실제 시스템에 적용해서 제어기의 성능을 입증하는 것도 흥미있는 과제가 될 것이다.

參考文獻

- [1] L. A. Zadeh, "Fuzzy sets," *Informt. Control*, vol. 8, pp. 338-353, 1965.
- [2] S. S. L. Chang and L. A. Zadeh, "On Fuzzy Mapping and Control," *IEEE Trans. Syst. Man Cybern.*, vol. 15, no. 1, pp. 175-189, 1985.
- [3] E. H. Mamdani, "Application of Fuzzy Algorithms for the Control of Dynamic Plants," *Proc. IEE*, vol. 12, no. 12, 1974
- [4] S. Assilian and E. H. Mamdani, "An Experiment in Liguistic Synthesis witha Fuzzy Logic Controller," *Int. J. Man-Machin Stud.*, vol. 7, 1974
- [5] C. C. Lee, "Fuzzy Logic in Control Systems : Fuzzy Logic Controller- Part I, II," *IEEE Trans. Syst. Man Cybern.*, vol. 20, no. 2, pp. 404-435, 1990.
- [6] R. M. Tong, "A Control Engineering Review of Fuzzy Systems," *Automatica*, vol. 13, pp. 559-569, 1977.
- [7] R. M. Tong, "Some Properties of Fuzzy Feedback Systems," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-10, no. 6, pp. 327-330, 1980.
- [8] K. Tanaka and M. Sano, "A New Tuning Method of Fuzzy Controllers," *Proc. IFSA '91*, pp. 207-210, 1991.
- [9] M. Maeda, T. Sato, and S. Murakami, "Design of the Self-Tuning Fuzzy

- Controller." *Proc. Int'l Conf. Fuzzy Logic and Neural Networks*, pp. 393-396, 1990.
- [10] S. Yamada, H. Fujikawa, A. Fuji, and J. Yamakawa, "A Tuning Method for a Fuzzy Controller," *Proc. Int'l Conf. Fuzzy Logic and Neural Networks*, pp. 397-400, 1990.
- [11] M. Yoshida, Y. Tshutsumi, and T. Ishida, "Gain Tuning Method for Design of Fuzzy Control Systems," *Proc. Int'l Conf. Fuzzy Logic and Neural Networks*, pp. 405-408, 1990.
- [12] S. Hayashi, "Auto-Tuning Fuzzy PI Controller," *Proc. IFSA '91*, pp. 41-44, 1991
- [13] B. Kuo, *Automatic Control Systems*, 6th ed., pp. 463-479. Prentice-Hall International Ed., 1991. bp

 著 者 紹 介



蔡 昌 鉉 (正會員)

1956年 3月 23日生. 1980年 2月 연세대학교 전기공학과 졸업. 1982년 2월 연세대학교 대학원 전기공학과 졸업 공학석사 학위 취득. 1989년 2월 연세대학교 대학원 전기공학과 졸업 공학박사 학위 취득. 1988년 9월 ~ 1990년 9월 금오공과대학교 전자제어공학과 전임강사. 1990년 10월 ~ 현재 금오공과대학교 전자제어공학과 조교수. 주관심분야는 적응제어이론, 전문가제어 및 퍼지제어 등임.



張 点 煥 (準會員)

1968년 12월 12日生. 1992년 2월 금오공과대학교 전자공학과 학사 졸업. (제어전공). 1992년 3월 ~ 현재 금오공과대학교 대학원 전자공학과 석사과정 재학중. (제어전공). 주관심분야는 퍼지제어, 신경회로망, 마이크로프로세서응용 등임.

李 志 弘 (正會員) 第 30 卷 B編 第 11 號 參照

현재 금오공과대학교 전자제어공학과 조교수