

## 직류 전동기 속도 제어에서 PI 제어기보다 우수한 퍼지 논리 제어기의 이득 선정을 위한 연구

(A Study on the Gain Tuning of Fuzzy Logic Controller Superior to PI Controller in DC Motor Speed Control)

김영렬\*

(Young-Real Kim)

### Abstract

Through a lot of papers, it has been concluded that fuzzy logic controller is superior to PI controller in motor speed control. Although fuzzy logic controller is superior to PI controller in motor speed control, the gain tuning of fuzzy logic controller is more complicated than that of PI controller. In this paper, using mathematical analysis of the PI and fuzzy controller, the design method of the fuzzy controller that has the same characteristics with the PI controller is proposed. After that, we can design the fuzzy controller that has superior performance than PI controller by changing the envelope of input of fuzzy controller to nonlinear, because the fuzzy controller has more degree of freedom to select the control gain than PI controller. The advantage of fuzzy logic controller is shown through mathematical analysis, and the simulation result using Matlab simulink has been proposed to show the effectiveness of these analysis.

Key Words : Fuzzy Logic Controller, PI Controller, DC Rotor, Speed Control

### 1. 서 론

전동기의 구동에 있어서 속도 제어는 대개 PI 제어가 일반적이다. 그런데 퍼지 제어기가 발표된 이후로는 많은 논문 및 연구에서 전동기 제어에서 PI 제어

기에 비해 퍼지 제어기가 상대적으로 우수하다는 견해가 호응을 얻고 있다[1-8]. 한편 상대적으로 적은 수의 연구 발표가 이루어지고 있지만, PI 제어기와 퍼지 제어기의 성능 차이가 특별히 없다는 결론을 보여주는 논문 역시 존재한다[9-12]. 또한 퍼지 논리 제어기가 PI 제어기에 비해 우수하다는 결론이 많음에도 불구하고 여전히 PI 제어기가 많이 사용되고 있기도 하다. 그 이유는 퍼지 논리 제어기의 이득 선정이라고 할 수 있는, 멤버십 함수 및 룰 테이블의 선정이 PI 제어기의 이득 선정에 비해 상대적으로 복잡하기 때문이라고 생각된다. 또한 퍼지 논리 제어기의 멤버십 함

\* 주저자 : 안양대학교 전기전자공학과 교수  
\* Main author : Professor Dept. of Electrical and Electronic Engineering at Anyang University  
Tel : 031-467-0869, Fax : 031-467-0800  
E-mail : yrkim@anyang.ac.kr  
접수일자 : 2014년 2월 6일  
1차심사 : 2014년 2월 14일, 2차심사 : 2014년 3월 11일  
심사완료 : 2014년 4월 29일

수 및 룰 테이블의 선정이 설계자의 경험에 많이 의존하기 때문이다. 반면에 PI 제어기의 경우는 이득을 구하기 위해 어느 정도의 기준을 갖고 적용할 수가 있는데, 일반적으로 Ziegler-Nichols 튜닝 방법[10] 또는, 직류 전동기의 경우에는 전동기 상수와 관련하여 적절히 선정하는 방법[11] 등이 있어 퍼지 논리 제어기 보다는 이득 선정이 체계적인 편이다. 따라서 본 논문에서는 퍼지 제어기를 설계하는데 있어서 PI 제어기의 이득을 이용하여 이를 체계적으로 변형하는 방법을 제시한다. 이렇게 함으로써 퍼지 제어기 설계가 단순화 되어 기존의 PI 제어기를 보다 나은 퍼지 제어기로 쉽게 변경할 수 있으며, 새로운 시스템에 퍼지 제어기를 설계할 때에도 기존의 PI 제어기 설계의 지식을 활용할 수 있게 된다. 이를 위하여 본 논문에서는 우선 퍼지 제어기와 PI 제어기의 유사성과 차이점을 이론적으로 수식을 통하여 비교하고, 이를 통해 퍼지 논리 제어기로 PI 제어기와 완전히 동일한 특성의 제어기 설계가 가능하다는 것을 먼저 보인다. 그리고 이러한 퍼지 논리 제어기의 멤버십 함수와 룰 테이블을 체계적으로 변형함으로써 PI 제어기보다 우수한 퍼지 논리 제어기 설계가 가능하다는 것을 보인다. 따라서 이러한 방법을 통하여 PI 제어기 보다 우수한 퍼지 제어 설계를 할 수 있는 방법을 제안하였다. 이를 위해 우선 matlab simulink에서 퍼지 제어기의 출력, 즉 시스템 입력인  $\Delta U$ 의 형태를 오차와 오차의 변화량에 대한 형태로 구해보고, 이를 PI 제어기에서 P 이득과 I 이득에 대한 제어기의 출력, 즉 시스템 입력인  $\Delta U$ 의 형태와 비교함으로써 PI 제어기와 퍼지 제어기의 유사성을 보였다. 그리고 퍼지 제어기로는 PI 제어기와 완전히 동일한 특성의 선형 제어기 설계가 가능함을 수식적으로 보였다. 마지막으로 이렇게 얻어진 퍼지 제어기를 기준으로  $\Delta U$ 의 형태를 오차와 오차에 대해 비선형적인 S 커브 형태가 되도록 룰 테이블을 변형함으로써 PI 제어기보다 단연 우수한 제어기가 얻어짐을 시뮬레이션을 통해 입증하였다. 이와 같이 퍼지 제어기의 설계에 PI 제어기의 이득을 활용함으로써 퍼지 논리 제어기의 룰 테이블의 선정이 기존의 방식에 비해 상대적으로 단순화 되었다. 시뮬레이션의 비교에 있어서 PI 제어기는 이득을 최적이득에 가

깝게 조정하였으며, 퍼지 논리 제어기는 이렇게 설계된 PI 제어기의 이득을 기본으로 멤버십 함수 및 룰 테이블을 선정하였다. 이렇게 함으로써 기존의 PI 제어기의 설계를 이용하여 체계적인 퍼지 논리 제어기 설계가 가능하다.

## 2. PI 제어기와 퍼지 제어기의 비교 분석

PI 제어기는 지금까지 가장 널리 사용되는 제어기이며, 전동기 속도 제어에서 퍼지 제어기의 성능을 비교할 때 그 비교 대상을 대부분 PI 제어기로 하게 된다. PI 제어기의 출력, 즉 시스템의 입력은 오차와 오차의 적분에 대해 구해지며 각각 비례 이득 P, 적분 이득 I가 된다. 그림 1 (a)에 PI 제어기의 구조를 보인다.

이러한 PI 제어기에서의 입력과 출력을 수식으로 표현해 보면 다음과 같다. PI 제어기의 출력은 시스템의 입력이 된다.

$$U = K_I \int E dt + K_P E \quad (1)$$

퍼지 제어기는 오차와 오차의 변화량에 대해 퍼지 제어 법칙을 사용하므로, PI 제어기를 이와 유사한 형태로 하기 위하여, 위식의 양변을 각각 미분하여 보면 다음 수식과 같이 된다.

$$dU = K_I E + K_P \dot{E} \quad (2)$$

이와 같이 미분한 것으로 PI 제어기를 표현하면 그림 1 (c)와 같이 나타낼 수 있다. 이를 연속시간이 아닌 이산시간의 디지털 제어로 구현하면, 다음과 같은 수식이 된다.

$$\Delta U = K_I E + K_P \Delta E \quad (3)$$

퍼지 제어기에서 제어기의 입력과 출력은 오차와 오차의 변화량에 대해 멤버십 함수를 할당하고, 이를 퍼지룰과 Defuzzification에 의해 제어기 출력을 계산한다. 이 과정은 PI 제어기에 비해 상당히 복잡해 보이지

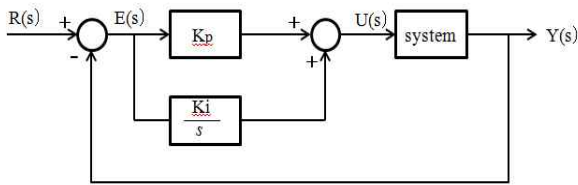
만, 결국 오차와 오차의 변화량에 대한 함수라는 점에서는 PI 제어기와 유사한 점이 있다. 오차와 오차의 변화량에 대한 멤버십 함수를 할당하기 위해 오차와 오차의 변화량을 각각 -1에서 1 사이로 두기로 하면, 제어기에 들어오는 오차와 오차의 변화량은 스케일을 변화시켜 주어야 하고 이때 오차에 대해서는  $K_1$ 을, 오차의 변화량에 대해서는  $K_2$ 를 곱하는 것으로 한다.

따라서 퍼지 제어기에서의 입력과 출력, 즉 시스템 입력은 다음과 같이 표시할 수가 있다.

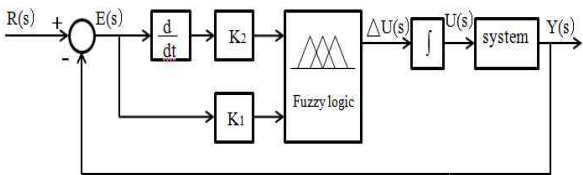
$$\Delta U = f(K_1 E, K_2 \Delta E) \quad (4)$$

이러한 퍼지 제어기의 구조를 그림 1 (b)에 나타낸다.

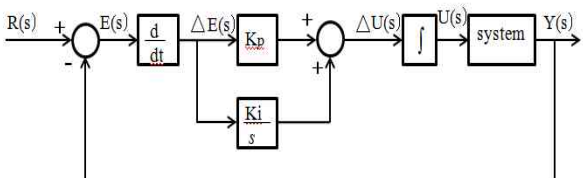
만일, 퍼지 제어기에서 이 함수를 선형 함수로 할 수 있다면 퍼지 제어기의 입출력을 다음과 같이 할 수 있다고 가정할 수 있다.



(a) PI 제어기



(b) 퍼지 제어기



(c) PI를 재구성한 구조

그림 1. PI 제어기 및 퍼지 제어기  
Fig. 1. PI controller and fuzzy controller

$$\Delta U = K_1 E + K_2 \Delta E \quad (5)$$

즉 퍼지 제어기를 선형화 시키면 PI 제어기에서의 적분이득  $K_I$ 와 비례이득  $K_P$ 는 퍼지 제어기에서의 오차에 대한 이득  $K_1$ 과 오차의 변화에 대한 이득  $K_2$ 와 같다고 생각할 수 있다.

따라서  $\Delta U$ 에 대한 그림을 일반적으로 퍼지 제어기로 많이 사용되는 Mamdani 형식에 대해 그려보면 그림 2 (a)와 같이 나타난다. 그림 2의 (a)에서 멤버십 함수는 triangular 형식을 사용한 것이고, 출력은 minimum of maxima의 무게중심법에 의해 defuzzification을 한 경우이다. 퍼지 제어기에서는 여러 가지 멤버십 함수의 형태와 rule table, 그리고 rule을 적용하는 여러 가지 형식을 고려하면 아주 다양한 형태의 비선형인  $\Delta U$ 의 형태가 나올 수 있다는 것을 짐작할 수 있다. 그림 2 (a)에서도 볼 수 있는 것처럼  $\Delta U$ 는  $K_1$ 과  $K_2$ 에 대해 비선형적인 형태가 된다.

앞의 수식 (2)와 (3) 그리고 그림 1의 (c)에서 볼 수 있듯이 PI 제어기에 대해서도 이를 미분한 형태에서  $\Delta U$ 를 구할 수 있고, 이를 그림으로 나타내면 그림 2 (b)와 같이 나타난다.

PI 제어기와 완전히 동일한 특성을 갖는 퍼지 논리 제어기 설계를 위하여 선형 특성을 갖는 퍼지 제어기를 구성해 볼 수 있는데, 이 경우의  $\Delta U$ 를 그림 2 (c)에 나타낸다. 여기에서 퍼지 제어기는 singleton 형식을 사용하고, 멤버십 함수는 triangular 형태를 사용하며, 제어기 출력은 product를 하여 defuzzification을 한다. 물론 singleton 형식에 대해서도  $\Delta U$ 는 다양한 형태로 나올 수 있고, 그림 2 (c)는 그 중의 한 가지 형태가 된다. 그림 2 (a), (b)와 (c)에서 볼 수 있는 것과 같이 퍼지 제어기는 PI 제어기에서의 적분이득  $K_I$ 와 비례이득  $K_P$ 가 오차의 크기 혹은 오차의 변화량의 크기에 따라 가변되는 구조가 된다. 즉 퍼지 제어기가 PI 제어기에 대해서 상대적으로 유리한 점은 운전점에 따라 이득이 비선형적으로 가변되는 구조라는 것이다. 그런 의미에서 퍼지 제어기로 PI와 완전히 같은 동작을 하는 제어기를 만드는 것이 가능하고, 그림 2에서 볼 수 있는 것과 같이 퍼지 제어기의 singleton 형식을 이용한 경우로 이를 구현할 수 있다.

이것은 좀 더 확장하면 PD 혹은 PID 제어를 퍼지 제어기로 표현이 가능하다는 것을 알 수 있다.

즉 singleton 형식의 퍼지 제어기에서

첫째 : 오차와 오차의 변화량에 대한 멤버십 함수를 단순한 삼각형 형태로 하고,

둘째 : 퍼지 제어기 출력에 대한 멤버십 함수를 singleton으로 하며,

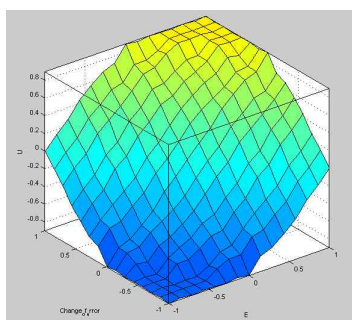
셋째 : 퍼지 제어기를 오차와 오차의 변화량을 입력

1과 입력2로 하여, 대각선 방향으로 좌측 상단에서 우측하단으로, 즉 오차와 오차의 변화량이 음으로 최대인 쪽에서 양으로 최대인 쪽으로 순차적으로 일정하게 증가시켜 나가고,

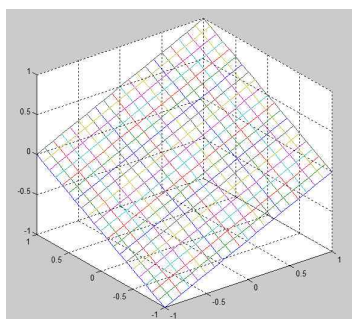
넷째 : 디퍼지화를 할 때 면적의 중심을 취하는 무게 중심 방식으로 하면,

이때의 퍼지 제어기에서 나오는  $\Delta U$ 는 그림 2 (c)에서처럼 PI 제어기와 같은 선형함수로 나타난다.

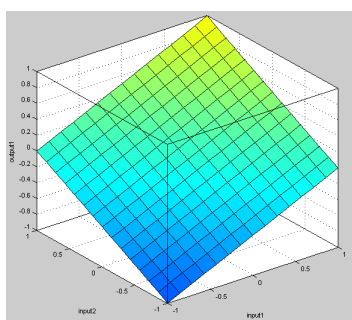
입력의 증가분  $\Delta U$ 가 적분이득  $K_I$ 와 비례이득  $K_P$ 에 대해 선형적인 형태로 나타날 수밖에 없는 PI 비해, 퍼지 제어기는 이를 포함한 다양한 비선형적인 형태가 가능하다. 그러한 다양한 형태중의 하나의 경우로써 퍼지 제어기로는 PI 제어기 구성이 가능하지만, PI 제어기로는 이러한 비선형적인 퍼지 제어기의 출력을 구성할 수 없다. 그만큼 퍼지 제어기는 선택 가능한 자유도가 무수히 많이 존재한다고 할 수 있고, 이러한 점에서 퍼지 제어기가 PI 제어기에 비해 유리하다고 할 수 있다.



(a) 비선형 퍼지 제어기에서의  $\Delta U$



(b) PI 제어기에서의  $\Delta U$



(c) 선형 퍼지 제어기에서의  $\Delta U$

그림 2. PI 제어기 및 퍼지 제어기에서의  $\Delta U$ 의 형태  
Fig. 2. Envelope of  $\Delta U$  in PI and fuzzy controller

### 3. PI 제어기 대응 퍼지 논리 제어기의 구성

Singleton 방식의 퍼지 제어를 이용하여 각각 그림 2 (c)의  $\Delta U$ 를 얻는데 적용한 입력에 대한 triangular 멤버십 함수를 그림 3에 나타낸다. 또한 출력에 대한 멤버십 함수를 그림 4에 나타낸다.

여기서 사용되는 퍼지 제어기는 singleton 즉, sugeno 방식을 사용하며, NB = -1, NS = -0.5, ZE = 0., PS = 0.5, PB = 1로 균일하게 분포하도록 한다.

사용되는 퍼지 룰은 표 1과 같으며 출력은 대각선 방향으로 균등하게 분포시킨다.

이제 퍼지 제어기로 PI 제어기와 같이  $\Delta U$ 가  $K_I E$ 와  $K_P \Delta E$ 에 대해 선형적으로 결정되는 퍼지 제어기를 구성하고, 실제로 이러한 퍼지 제어기가 PI 제어기와 동일한 지를 수식적으로 확인하도록 한다. 그러기 위해서는 입력의 멤버십 함수는 그림 3과 같은 형태를 사용한다. 퍼지 룰 테이블은 표 1과 같은 3×3 형태의 룰 테이블을 사용한다. 그리고 일반적인 경우로 출력의 singleton 형태는 다음 수식과 같이 정하도록 한다.

직류 전동기 속도 제어에서 PI 제어기보다 우수한 퍼지 논리 제어기의 이득 설정을 위한 연구

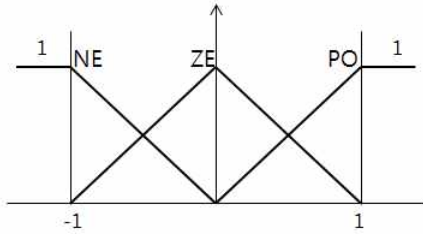


그림 3. 입력 멤버십 함수  
Fig. 3. Input membership function

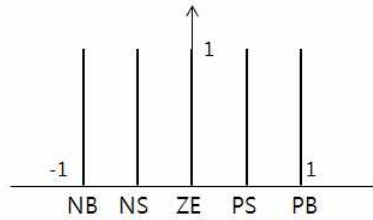


그림 4. 출력의 멤버십 함수  
Fig. 4. Output membership function

표 1. 퍼지룰 테이블  
Table 1. Fuzzy rule table

		error		
		NE	ZE	PO
change of error	NE	NB	NS	ZE
	ZE	NS	ZE	PS
	PO	ZE	PS	PB

$PB$

$$PS = \frac{PB}{2}$$

$$ZE = 0$$

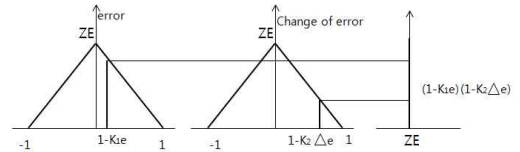
$$NS = \frac{NB}{2} = -\frac{PB}{2}$$

$$NB = -PB \quad (6)$$

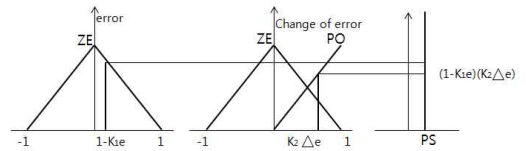
오차와 오차의 변화량의 크기는 입력의 멤버십 함수의 범위인 -1과 1 사이에 있도록 각각 스케일을 조정하기 위해  $K_1$ 과  $K_2$ 를 곱해 주므로,  $K_1e$ 와  $K_2\Delta e$ 에 대해 다음과 같이 네 가지 경우로 나누어 수식적으로 확인해 보도록 한다.

1)  $0 \leq K_1e \leq 1$  이고,  $0 \leq K_2\Delta e \leq 1$  일 때

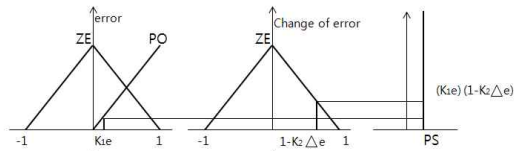
이 때 출력은 product에 의한 singleton으로 구하도록 하며, 룰 테이블에서 출력이 속하는 경우는 다음의 네 가지 경우가 되고, 출력 ZE, PS, PS, PB에 대한 singleton 값은 다음식과 같이 구해진다.



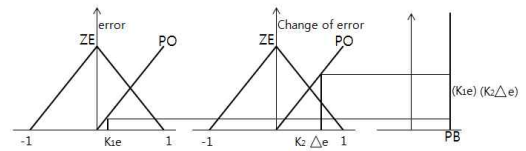
(a) error = ZE, change of error = ZE



(b) error = ZE, change of error = PO



(c) error = PO, change of error = ZE



(d) error = PO, change of error = PO

그림 5. 출력 singleton의 계산 (product)

Fig. 5. calculation of output singleton (product)

$$ZE: (1 - k_1e)(1 - k_2\Delta e) = a$$

$$PS: (1 - k_1e)(k_2\Delta e) = b$$

$$PS: (k_1e)(1 - k_2\Delta e) = c$$

$$PB: (k_1e)(k_2\Delta e) = d \quad (7)$$

$$\Delta U = \frac{a \cdot 0 + b \cdot PS + c \cdot PS + d \cdot PB}{a + b + c + d} \quad (8)$$

여기서 분모와 분자는 각각 다음과 같이 계산된다.

$$a+b+c+d(\text{분모}) \tag{9}$$

$$\begin{aligned} &= (1-k_1e-k_2\Delta e+k_1e \cdot k_2\Delta e) \\ &+ (k_2\Delta e-k_1e \cdot k_2\Delta e) \\ &+ (k_1e-k_1e \cdot k_2\Delta e) \\ &+ k_1e \cdot k_2\Delta e \\ &= 1 \end{aligned}$$

$$a \cdot 0+b \cdot PS+c \cdot PS+d \cdot PB(\text{분자}) \tag{10}$$

$$\begin{aligned} &= (b+c)PS+d \cdot PB \\ &= (b+c)\frac{PB}{2}+d \cdot PB \\ &= \left(\frac{b+c}{2}+d\right)PB \\ &= \left(\frac{k_2\Delta e-k_1e \cdot k_2\Delta e}{2}\right)\frac{PB}{2} \\ &+ \left(\frac{k_1e-k_1e \cdot k_2\Delta e}{2}\right)\frac{PB}{2} \\ &+ (k_1e \cdot k_2\Delta e)PB \\ &= (k_1e+k_2\Delta e)\frac{PB}{2} \end{aligned}$$

즉,

$$\Delta U = (k_1e+k_2\Delta e)\frac{PB}{2} \tag{11}$$

이 된다.

이것은 선형제어기인 PI 제어기와 동일한 형태가 된다는 것을 알 수 있다.

2)  $0 \leq K_1e \leq 1$  이고,  $-1 \leq K_2\Delta e \leq 0$  일 때 같은 방법으로 이 경우에  $\Delta U$ 를 계산하여 보면,

$$\Delta U = (k_1e-k_2\Delta e)\frac{PB}{2} \tag{12}$$

이 된다.

식 (12)에서 식 (11)과 부호가 달라진 것은  $\Delta e$ 가  $-\Delta e$ 가 되었기 때문이다.

3)  $-1 \leq K_1e \leq 0$  이고,  $0 \leq K_2\Delta e \leq 1$  일 때 같은 방법으로 이 경우에  $\Delta U$ 를 계산하여 보면,

$$\Delta U = (-k_1e+k_2\Delta e)\frac{PB}{2} \tag{13}$$

이 경우에는  $e$ 가  $-e$ 가 되었으므로 식 (11)과 부호가 달라진다.

4)  $-1 \leq K_1e \leq 0$  이고,  $-1 \leq K_2\Delta e \leq 0$  일 때 같은 방법으로 이 경우에  $\Delta U$ 를 계산하여 보면,

$$\Delta U = (-k_1e-k_2\Delta e)\frac{PB}{2} \tag{14}$$

이 경우에는  $e$ 가  $-e$ 로, 그리고  $\Delta e$ 는  $-\Delta e$ 로 되었기 때문에 부호가 달라진다.

식 (11), (12), (13), (14)에서 볼 수 있는 것처럼 이들 수식은 PI 제어기에서의 수식 (3)과 동일한 형태가 된다. 즉  $K_1\frac{PB}{2} = K_P$ ,  $K_2\frac{PB}{2} = K_I$ 로 하면 PI 제어기와 퍼지 제어기는 완전히 동일한 형태가 된다. 결국 퍼지 제어기로는 PI 제어기와 동일한 형태를 구성해 낼 수 있다는 것을 수식적으로 보였다. 반대로 퍼지 제어기는 이 외에도 얼마든지 비선형 특성을 갖는 제어기를 구성할 수 있고, PI 제어기로는 이러한 비선형 퍼지 제어기와 동등한 제어기를 구성할 수 없다. 결론적으로 이러한 특성 때문에 퍼지 제어기는 PI 제어기보다 우수한 특성의 제어기를 만들어 낼 수 있는 가능성이 존재한다. 그러므로 위에서 얻어진 퍼지 제어기를 기준으로  $\Delta U$ 의 형태를 오차와 오차에 대해 비선형적인 S 커브 형태로 수정함으로써 PI 제어기보다 단연 우수한 제어기가 얻어짐을 시뮬레이션을 통해 입증하였다. 이렇게 함으로써 퍼지 제어기의 이득 선정, 즉 멤버십 함수 및 룰 테이블의 선정이 상당히 단순화 되었다. 여기에서는 퍼지 제어기의 입력 및 출력의 멤버십 함수로 3x3인 것으로 구성하였지만, 이것은 nxn인 멤버십 함수인 경우에도 성립한다는 것을 시뮬레이션을

직류 전동기 속도 제어에서 PI 제어기보다 우수한 퍼지 논리 제어기의 이득 선정을 위한 연구

통해 간단히 확인할 수 있다. 즉 오차와 오차의 변화에 대한 퍼지 룰 테이블이  $n \times n$ 인 경우에는 출력에 대한 singleton을  $(2n-1)$ 개의 값으로 하면 된다. 그러면 양의 값으로  $(n-1)$ 개, 음의 값으로  $(n-1)$ 개, 그리고 zero가 있게 된다. 따라서 양의 값으로  $(n-1)$ 개를 균등하게 선정하고, 음의 값으로  $(n-1)$ 개를 균등하게 선정하면,  $n \times n$ 의 멤버십 함수에 대한 경우로 일반화 할 수 있다. 이것은 그림 2 (c)에서 확인이 가능한데, 그림 2 (c)에서는  $5 \times 5$ 의 멤버십 함수를 이용한 경우이며, 이 경우에도 선형적인 퍼지 제어기를 구성할 수 있다는 것을 보여준다.

#### 4. PI 제어기와 퍼지 제어기의 시뮬레이션 결과 비교

PI 제어기와 퍼지 제어기의 성능을 비교하기 위한 직류 전동기 속도 제어에서 시뮬레이션을 수행한 결과를 다음에 보인다. 시뮬레이션에 사용된 직류 전동기의 상수는 표 2와 같다.

표 2. 전동기 상수  
Table 2. Motor parameter

Power	3.3kW	Current	25A
Speed	3,000rpm	Ra	0.26Ω
Torque	10.5Nm	La	1.7mH

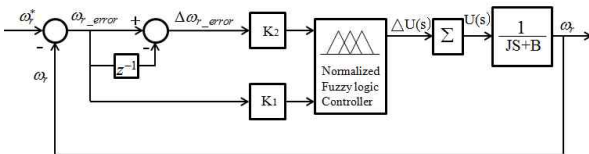
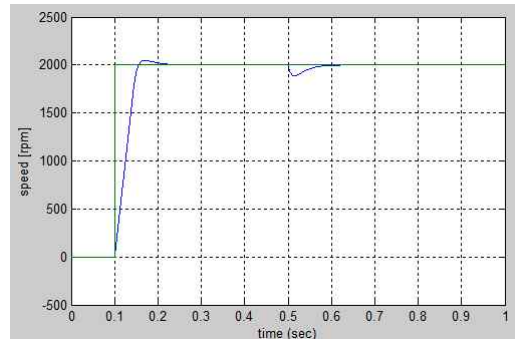


그림 6. 퍼지 제어기의 직류 전동기 속도 제어 시스템 블록도

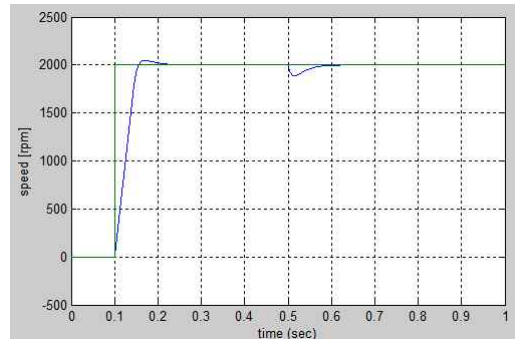
Fig. 6. System Block Diagram of fuzzy controller for speed control of DC motor

세 가지 제어기를 비교하였는데, 하나는 PI 제어기이고, 또 하나는 이와 동일한 특성을 갖도록 설계한 singleton 방식의 퍼지 제어기, 그리고 마지막으로

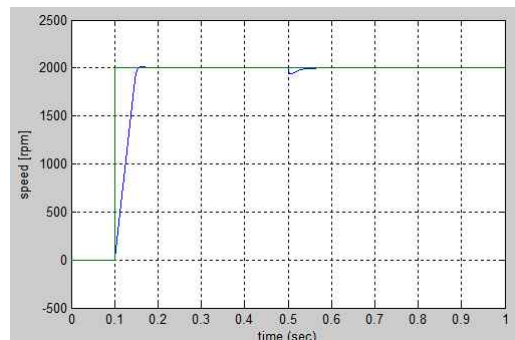
Mamdani 형식의 퍼지 제어기로서 그림 2 (a)와 같이 S 자형 커브로  $\Delta U$ 를 구하도록 하는 퍼지 제어기를 비교하였다. 퍼지 제어기를 이용한 속도 제어 시스템은 그림 6과 같이 구성된다. 여기서  $K_1$ 과  $K_2$ 는 속도



(a) PI 제어기



(b) singleton 퍼지 제어기(PI와 동일한 특성으로 설계)

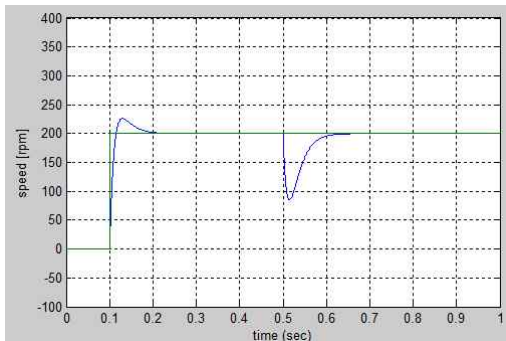


(c) mamdani 방식의 퍼지 제어기

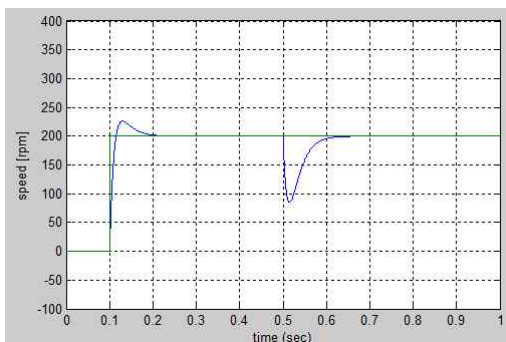
그림 7. PI 제어기 및 퍼지 제어기의 직류 전동기 속도 제어 성능(기준속도 2,000rpm인 경우)

Fig. 7. Performance of PI and fuzzy controller for speed control of DC motor(reference speed: 2,000rpm)

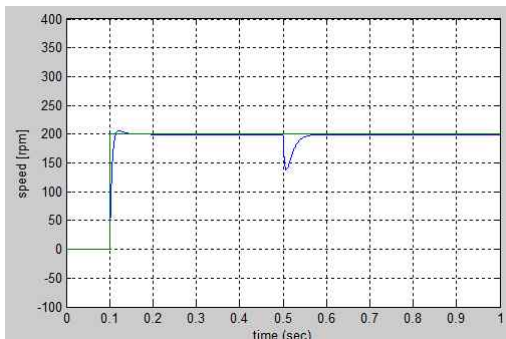




(a) PI 제어기



(b) singleton 퍼지 제어기(PI와 동일한 특성으로 설계)



(c) mamdani 방식의 퍼지 제어기

그림 8. PI 제어기 및 퍼지 제어기의 직류 전동기 속도 제어 성능(기준속도 200rpm인 경우)  
Fig. 8. Performance of PI and fuzzy controller for speed control of DC motor(reference speed: 200rpm)

오차  $\omega_{r-error}$  와 속도 오차의 변화량  $\Delta\omega_{r-error}$  이 -1에서 1사이가 되도록 정규화하기 위한 scale factor이다. 속도 제어기는 디지털 제어를 사용하였고, 1msec마다 속도 제어기가 동작하도록 하였다.

입력의 멤버십 함수에 필요한 속도 오차  $\omega_{r-error}$  와 속도 오차의 변화량  $\Delta\omega_{r-error}$  은 scale factor  $K_1$ 과  $K_2$ 가 각각 곱해져  $K_1\omega_{r-error}$ ,  $K_2\Delta\omega_{r-error}$ 가 되며, 이 값은 각각 -1에서 1 사이의 값이 되도록 한다. 이 값은 그림 3과 같은 멤버십 함수의 입력값이 된다.  $K_1\omega_{r-error}$ ,  $K_2\Delta\omega_{r-error}$ 에 대한 멤버십 함수는 각각 정규화 되어 두 가지 모두 그림 3과 같은 멤버십 함수가 사용된다. 출력의 멤버십 함수는 그림 4와 같은 singleton 방식의 멤버십 함수가 사용되지만, 그림 4는 -1에서 1 사이로 정규화 된 것이고, 실제로는  $PB=-NB$ 는 이득값으로 사용된다. 따라서 실제로 이득은 scale factor  $K_1$  및  $K_2$ 와  $PB$ 의 곱이 되며, 여기에서는  $K_1 \frac{PB}{2} = K_I = 117.11$ ,  $K_2 \frac{PB}{2} = K_P = 1.8639$ 의 값이 사용되었으며, 이 값은 PI 제어기에서 최적화된 값이다. Defuzzification에 의한 제어기의 입력은

표 3. singleton 퍼지 제어기(PI와 동일한 특성으로 설계)에 사용된 퍼지를 테이블

Table 3. Fuzzy rule table used in singleton method fuzzy controller

		error				
		NB	NE	ZE	PO	PB
change of error	NB	NB	NM	NS	NVS	ZE
	NE	NM	NS	NVS	ZE	PVS
	ZE	NS	NVS	ZE	PVS	PS
	PO	NVS	ZE	PVS	PS	PM
	PB	ZE	PVS	PS	PM	PB

표 4. mamdani 방식의 퍼지 제어기에 사용된 퍼지를 테이블

Table 4. Fuzzy rule table used in mamdani method fuzzy controller

		error				
		NB	NE	ZE	PO	PB
change of error	NB	NB	NB	NM	NS	ZE
	NE	NB	NM	NS	ZE	PS
	ZE	NM	NS	ZE	PS	PM
	PO	NS	ZE	PS	PM	PB
	PB	ZE	PS	PM	PB	PB



식 (11)~(14)와 같이 계산되며, 입력 U는 전기자 전류의 기준치가 된다. 직류 전동기에서 토크는 전기자 전류에 비례하기 때문이다.

퍼지 제어기에서 멤버십 함수는 triangle 방식을 사용하였고, PI 제어기와 동일한 특성을 갖도록 설계된 singleton 방식의 퍼지 제어기에는 표 3과 같은 퍼지 룰 테이블을 적용하여 그림 2 (c)와 같이,  $\Delta U$ 가 오차와 오차의 변화량에 대하여 선형적이 되도록 설계하였다. 가장 좋은 응답을 갖도록 mamdani 방식으로 설계한 퍼지 제어기는 표 4와 같은 퍼지 룰 테이블을 적용하였으며, 그림 2 (a)와 같이  $\Delta U$ 가 오차와 오차의 변화량에 대해 비선형적으로 되도록 설계하였다.

당연한 결과이지만 PI 제어기와 이와 동일한 특성을 갖도록 설계한 singleton 방식의 퍼지 제어기는 동일한 특성을 나타내었다.

$\Delta U$ 를 오차 및 오차의 변화량에 대해 S 자형 커브로 구하도록 한 퍼지 제어기는 singleton 방식의 퍼지 제어기나 PI 제어기보다 우수한 성능을 보여주었는데, 이 때  $K_1$ 과  $K_2$ 는 singleton 방식에서와 같게 하였다.

즉 PI 제어기와 같은 특성으로 설계한 singleton 방식의 퍼지 제어기에서 단지, 룰 테이블의 할당을  $\Delta U$ 의 형태가  $K_1\omega_{r-error}$ ,  $K_2\Delta\omega_{r-error}$ 에 대해 S 자형 커브를 갖도록 수정하였을 뿐인데, 응답 특성은 PI 제어기보다 우수하게 된다. 그렇다고 해서 이것이 Mamdani 방식이 singleton 방식보다 우수하다는 것은 의미하는 것은 아니다. 단지 본 논문에서의 singleton 방식에서는 PI 제어기와 동일한 특성을 갖도록 설계를 하였기 때문이다.

singleton 방식에서도 역시 룰 테이블을  $\Delta U$ 가  $K_1\omega_{r-error}$ ,  $K_2\Delta\omega_{r-error}$ 에 대해 S 자형 커브로 되도록 하면 PI 제어기보다 우수한 특성이 얻어진다. 즉 이는 퍼지 제어기가 PI 제어기보다 우수한 특성을 보이는 것이 S 자형 커브로 비선형적이기 때문이라는 것을 의미한다.

S 자형 커브라는 것은 오차가 클 때에는 이득을 적게 줄여주고, 오차가 적을 때에는 이득을 크게 가변한다는 것을 의미한다. 이렇게 이득을 가변할 수 있다는

것이 퍼지 제어기의 우수함이 된다는 것을 확인할 수 있다. 따라서 제안한 방식으로 퍼지 제어기를 설계함으로써 퍼지 제어기의 멤버십 함수 및 룰 테이블을 비교적 단순화하여 설계할 수 있으며, 이 결과는 PI 제어기보다 우수한 특성을 나타내었다.

그림 7에 세 가지 제어기에서의 속도 응답을 나타낸다. 기준 속도 2,000rpm으로의 속도 지령에 대한 응답을 나타내며, 0.5초에 정격의 1/2에 해당하는 부하를 인가하는 경우이다. 그림 7 (a)의 PI 제어기와 그림 7 (b)의 singleton 방식의 퍼지 제어기는 동일한 성능을 보이며, Mamdani 방식의 퍼지 제어기에서는 이들 보다 우수한 성능을 보이고 있다. 기준 속도를 따라 가는 응답과 오버슈트 등에서 우수할 뿐만 아니라 부하 인가시의 응답에서도 그 차이를 볼 수 있다.

그림 8에는 기준 속도 200rpm에서의 속도 응답을 나타내며, 역시 정상 상태 운전하고 있는 중에 0.5초에서 정격 부하의 1/2 부하를 인가한 경우이다. 2,000rpm에서의 시뮬레이션 결과와 비슷하게 그림 8의 속도 응답에서도 역시 Mamdani 방식의 퍼지 제어기가 단연 우수한 특성을 보이고 있다. 결론적으로 PI 제어기와 동일한 성능을 나타내는 퍼지 제어기를 언제든지 설계할 수 있고, 이 보다 우수한 성능을 보이는 퍼지 제어기를 설계할 수 있는 가능성이 항상 존재한다는 점에서 퍼지 제어기가 PI 제어기에 대해 유리하다. 단지 룰 테이블의 할당을  $\Delta U$ 의 형태가  $K_1\omega_{r-error}$ ,  $K_2\Delta\omega_{r-error}$ 에 대해 S 자형 커브를 갖도록 수정함으로써 응답 특성을 PI 제어기보다 단연 우수하게 할 수 있다는 것은 퍼지 제어기의 우수성이 이러한 비선형성에 있음을 입증하며, 또한 퍼지 제어기 설계에서 PI 제어기의 최적 이득을 이용할 수도 있음을 의미한다.

## 5. 결 론

본 논문에서는 PI 제어기와 동일한 특성의 퍼지 제어기 설계가 가능함을 보이고, 그 설계 방법을 제시하였으며, 이를 수식으로 확인하였다. 이를 바탕으로 PI 제어기의 이득을 이용하여 동일한 특성의 퍼지 제어기를 설계하고, 이를 S자형 커브를 갖는 비선형적인 퍼지 제어기로 수정하는 것만으로 PI 제어기보다 단연

우수한 퍼지 제어기 설계가 가능함을 보였다. 따라서 지금까지 퍼지 제어기의 이득, 즉 멤버십 함수와 룰 테이블을 경험적으로만 구하는 것에 비하여 상당히 체계적으로 구할 수 있는 방법이 제안되었다. 그리고 이렇게 비선형으로 변형한 퍼지 제어기는  $\Delta U$ 의 형태가 오차 및 오차의 변화량에 대해 비선형적으로 표현됨으로써 선형제어기인 PI 제어기 보다 우수하게 된다. 이는 시뮬레이션 결과에서도 확인이 가능하였다.

본 논문은 연구년 기간 중 연구되었음.

### References

[1] B. K. Bose, "Expert systems, fuzzy logic, and neural network applications in power electronics and motion control," Proc. IEEE, vol. 82, pp. 1303-1323, Aug. 1994.

[2] Chung-Yuen Wan, Sei-Chan Kim, Bose, B.K, "Robust position control of induction motor using fuzzy logic control," Conference Record of the 1992 IEEE Industry Applications Society Annual Meeting, 1992., pp. 472 - 481 vol.1.

[3] Heber, B., Longya Xu, Tang, Y., "Fuzzy logic enhanced speed control of an indirect field-oriented induction machine drive," IEEE Transactions on Power Electronics, Volume: 12, Issue: 5, pp. 772 - 778, 1997.

[4] Uddin, MN., Radwan, T.S., Azizur Rahman, M., "Performances of fuzzy-logic-based indirect vector control for induction motor drive," IEEE Transactions on Industry Applications, pp.1219 - 1225, 2002.

[5] Jae-Sung Yu, Sang-Hoon Kim, Byoung-kuk Lee, Chung-Yuen Won, Jin Hur, "Fuzzy-Logic-Based Vector Control Scheme for Permanent-Magnet Synchronous Motors in Elevator Drive Applications," IEEE Transactions on Industrial Electronics, Volume: 54, Issue: 4, pp. 2190-2200, 2007.

[6] Jae-Sub Ko, Jung-Sik Choi, Dong-Hwa Chung, "Hybrid Fuzzy Controller for DTC of Induction Motor Drive," Journal of the Korean Institute of Illuminating and Electrical Installation Engineers, Vol. 25, No. 5, May 2011.

[7] Jae-Sung Yu, Young-Hwan Yoo, Byoung-Kuk Lee, Chung-Yuen Won, "Full Fuzzy-Logic-Based Vector Control for Permanent Magnet Synchronous Motors," Journal of the Korean Institute of Illuminating and Electrical Installation Engineers, Vol. 20, No. 10, Dec. 2011.

[8] Jae-Sung Yu, Soo-Cheol Sin, Ju-Man Yoon, Chung-Yuen Won, Sang-Hoon Kim, "Improvement of Speed Control for Spindle Induction Motor in the Field Weakening Region using a Fuzzy Controller," Journal of the Korean Institute of Illuminating and Electrical Installation Engineers, Vol. 19, No. 8, Dec. 2005.

[9] Sousa, G.C.D., Bose, B.K., "A fuzzy set theory based control of a phase-controlled converter DC machine drive," IEEE Transactions on Industry Applications, Volume: 30, Issue: 1 pp. 34 - 44, 1994.

[10] Donescu, V., Neacsu, D.O., Griva, G., Profumo, F., "A systematic design method for fuzzy logic speed controller for brushless DC motor drives," Power Electronics Specialists Conference, 1996. PESC '96 Record., 27th Annual IEEE Volume: 1, 1996, pp. 689 - 694 vol.1.

[11] Mrad, F., Deeb, G., "Experimental comparative analysis of conventional, fuzzy logic, and adaptive fuzzy logic controllers," Conference Record of the 1999 IEEE Industry Applications Conference, 1999. Thirty-Fourth IAS Annual Meeting. Volume: 1, 1999, pp. 664 - 673 vol.1.

[12] Ibrahim, Z., Levi, E., "A comparative analysis of fuzzy logic and PI speed control in high-performance AC drives using experimental approach," IEEE Transactions on Industry Applications, Volume: 38, Issue: 5, pp. 1210 - 1218, 2002 Peter R. Boyce, Human Factors in Lighting, 2nd Edition, Lighting Research Center, 2003.

[13] Digital Control of Dynamic Systems(3rd Ed.), Gene F. Franklin, J. David Powell, and Michael Workman, Ellis-Kagle Press, 2006.

[14] Sang-Hoon Kim, "DC, AC, BLDC Motor Control," Bogdoe Press, 2010.

### ◆ 저자소개 ◆



**김영렬**(金暎烈)

1958년 9월 5일생. 1982년 서울대학교 전기공학과 졸업. 1982~1987년 한국기계연구소 부설 기업기술지원센터 연구원. 1987년 서울대학교 전기공학과 졸업(석사). 1993년 서울대학교 전기공학과 졸업(박사). 1993~1996년 한국항공우주연구소 선임 연구원. 1996년~현재 안양대학교 전기전자공학과 교수.