

적응퍼지 알고리즘을 이용한 DC서보 전동기의 위치제어

지 성현*, 손 재현*, 전 병태*, 임 종광**, 남 문현*

* 건국대학교 전기공학과 ** 국방과학연구소

Position Control For A DC Servo Motor Using Adaptive Fuzzy Algorithm

Sung-Hyon Ji*, Jae-Hyun Son*, Byong-Tae Jeon*, Jong-Kwang Lim**, Moon-Hyon Nam*

* Dept. of Electrical Eng., Kon-Kuk Univ.

** Agency for Defense Development

Abstract

Fuzzy Logic Control imitating human decision making process is a novel control strategy based on expert's experience and knowledge and many process designers are developing its applications. But it is difficult to obtain a set of rules from human operators. And there is a limitation on adjusting to environmental changes.

In this paper, we proposed adaptive fuzzy algorithm to overcome these difficulties using weights added to the rules.

To verify the validity of this control strategy, we have implemented this algorithm for a DC servo motor with PD-type fuzzy controller.

I. 서론

현대 산업 사회에서 서보 전동기의 이용범위는 광범위하며 그 역할 또한 매우 중요하다. 서보 전동기 제어는 보통 구조가 간단하고, 실현하기가 용이한 PID 제어를 많이 이용해 왔다.

그러나 고정이득 제어기는 제어환경의 변화와 시스템의 비선형성에 대해 적절하게 적응하지 못한다. 이러한 문제점들은 파라미터 추정에 의해 미지의 플랫폼 및 복잡한 제어 시스템의 비선형, 시변 특성을 극복할 수 있는 적응제어 기법으로 해결할 수 있다. 그러나 이것도 시스템의 차수가 높아지거나 매개변수의 수가 증가하면 수학적 계산이 복잡해지고 모델링에 어려움이 따르며, 많은 계산시간이 소요되므로 빠른 응답 특성을 요구하는 시스템에 적용하는데 제한요소가 된다.

그래서 최근 이러한 문제를 해결하기 위하여 전문가의 경험이나 지식을 바탕으로 인간의 의사 결정과정을 모방한 퍼지 제어 기법을 서보 시스템에 적용하려는 연구가 진행되고 있다 [1][2][3]. 그러나 전문가로부터 제어규칙을 도출하는 것이 항상 가능한 것이 아니며, 또한 복잡한 동특성을 가진 시스템의 경우 적절한 퍼지규칙을 찾는 것도 쉽지 않으므로 초기에 설계된 퍼지규칙이 시스템 제어에 부적절하게 된다.

그래서 Procyk 등은 자기 조직 제어기(Self Organizing Controller, SOC)[4][8]를 제안하였는데, 이는 플랫폼의 전체

제어성능을 평가하여 규칙을 수정함으로써 제어성능을 개선해 나간다.

본 연구에서는 퍼지제어기 동작중에 측정된 각 규칙의 적용횟수를 가중치(Contribution Weight)로 사상하여 추론 멤버십값에 곱함으로써 제어출력에 대한 각 규칙의 기여도를 확대 또는 축소시킨다. 그럼으로써 변화하는 환경에 적응하여 퍼지 규칙을 수정하는 효과를 얻을 수 있는 가중치 조정법을 제안한다. 이를위해 기존의 PD형 퍼지제어기[3]를 갖는 DC서보 전동기에 적용하여 그 타당성을 시뮬레이션을 통해 검증하였다.

II. 적응퍼지 제어기 설계

2-1. 적응퍼지 제어기 구조

적응퍼지 제어기는 PD 형 퍼지제어기와 성능평가 및 가중치 조정부로 구성된다. PD형 퍼지제어기는 퍼지화부(fuzzifier), 퍼지추론부(fuzzy inference engine), 비퍼지화부(defuzzifier)로 구성되며 그림 1 과 같다.

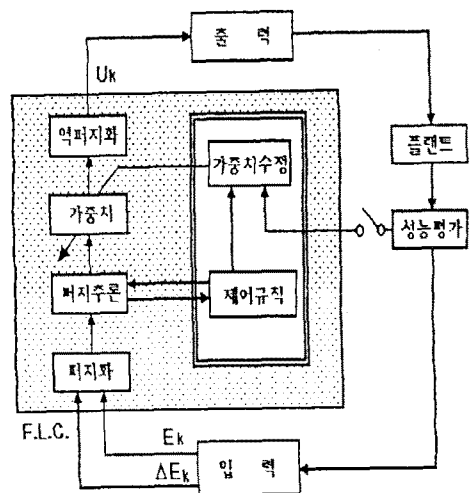


그림 1. 전체 시스템 구성도

2-2. PD 형 퍼지 제어기 설계

본 연구에서는 퍼지제어기의 입력변수로 오차 E_k 와 오차변화량 ΔE_k 를 선정하였고 출력변수로 제어출력 U_k 를 선정하였으며 다음과 같이 정의한다.

$$E_k = SP - Y_k$$

$$\Delta E_k = E_k - E_{k-1}$$

여기서 SP(Set-Point)는 요구위치이고, Y_k 는 실제 출력값을 나타낸다. 정의된 입력변수($E_k, \Delta E_k$)는 매 샘플링마다 측정되어 IF-THEN 형태의 퍼지규칙(fuzzy rule)에 의하여 퍼지제어기의 출력을 결정하게 되는데, 규칙의 정성적 표현을 정량화하기위해 다음과 같은 5단계의 언어값(linguistic value)을 정의한다.

- LP : Large Positive
- SP : Small Positive
- ZE : Zero Error
- SN : Small Negative
- LN : Large Negative

이 언어값들은 전체집합(Universe of Discourse) U에 대해 퍼지 부분집합(Fuzzy subset)이며, 언어값들을 퍼지화하기 위해서는 멤버십함수(membership function)를 이용해야 한다. 멤버십함수는 사용자의 경험과 지식에 의한 주관에 의해 정의되므로, 본 연구에서는 DC 서보 전동기의 선형 및 비선형 특성을 고려하여 Crisp한 영역에 Fuzzy 경계를 조합한 구조인 사다리꼴 형태로 설계하였다. 입-출력변수에 대한 전체집합 U는 이산적인 11개의 정량화레벨(quantization level)들을 원소로 가지며, 표 1은 이산형 멤버십 함수를 나타낸다.

$$U = \{-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5\}$$

표 1. 이산형 멤버십 함수

Linguistic fuzzy Set	Quantized Levels										
	-5	-4	-3	-2	-1	0	1	2	3	4	5
LP	0	0	0	0	0	0	0	0	.5	1	1
SP	0	0	0	0	0	.5	1	.5	0	0	0
ZE	0	0	0	.5	1	.5	0	0	0	0	0
SN	0	0	.5	1	.5	0	0	0	0	0	0
NL	1	1	.5	0	0	0	0	0	0	0	0

퍼지제어규칙은 IF - THEN 형태의 조건문과 동작문으로 구성되며, 언어값을 사용하여 본 논문에서는 제어대상의 페루우프 시스템의 오버슈트(overshoot)가 감소하고, 상승시간(rise time)을 짧게하는 퍼지제어규칙을 표 2와 같이 설계하였다. [3][5][6]

표 2. 퍼지 제어규칙

E_k	ΔE_k	LN	SN	ZE	SP	LP
LN	LN	SN	LN	SN	SP	LP
SN	LN	SN	SN	ZE	SP	LP
ZE	LN	SN	ZE	SP	LP	LP
SP	LN	ZE	SP	SP	LP	LP
LP	SN	SP	LP	LP	LP	LP

퍼지추론은 Mandani[2][4]가 제안한 최대-최소 합성(Max-Min Composition)을 사용하였다. 즉, 정량화된 입력변수 $E_k, \Delta E_k$ 에 대해 각 퍼지규칙들의 멤버십 함수를 구하여, 제어

출력을 결정하게 되는데 제어기 출력계산을 위한 각 규칙의 멤버십값(DOC)은 각기 다르며, 다음과 같이 정의한다.

$$DOC = \min\{\mu_E(E_k), \mu_{\Delta E}(\Delta E_k)\}$$

여기서 μ 는 멤버십 함수이다. 추론된 퍼지집합으로부터 제어기 출력을 구하기 위해서 본 논문에서는 무게중심법(Center of gravity:COG)[7]을 사용하여 비퍼지화 하였으며 다음과 같이 정의한다.

$$u = \frac{\sum_{i=0}^n DOC_i u_i}{\sum_{i=0}^n DOC_i}$$

여기에서 n은 규칙의 수이고, u_i 는 1번째 규칙에 의한 출력의 최대값이다.

본 연구에서는 이와같은 과정을 통해 모든 가능한 입력쌍에 대한 출력값을 계산하여 Look-Up Table을 작성하여 표 3에 나타내었다

표 3. Look - Up Table

E_k	ΔE_k	-5	-4	-3	-2	-1	0	1	2	3	4	5
-5	-5	-5	-4	-3	-3	-2	-2	-1	0	1	1	2
-4	-5	-5	-5	-4	-4	-3	1	0	1	1	2	2
-3	-5	-5	-5	-4	-3	1	0	1	0	1	2	2
-2	-5	-5	-4	-4	-3	-1	0	2	3	3	3	3
-1	-5	-5	-4	-3	-2	0	1	2	3	3	3	3
0	-5	-4	-4	-2	-2	0	1	3	4	4	4	4
1	-4	-4	-3	-2	-1	0	2	3	4	4	4	4
2	-4	-4	-3	-2	-1	1	2	3	4	4	4	4
3	-4	-4	-2	1	0	1	2	3	4	4	4	5
4	-3	-3	-2	-1	0	1	3	3	4	4	5	5
5	-3	-2	-1	0	1	2	3	4	4	4	5	5

2-3. 성능평가 및 가중치 조정기

추론 및 비퍼지화 과정에서 DOC_i 는 퍼지제어기 추론결과로서 얻어지는 멤버십 값으로 각 규칙들이 출력에 미치는 영향의 정도를 나타낸다. 즉, 추론과정에서 얻어지는 각 규칙의 결론은 퍼지제어기의 출력을 결정할 때 사용되는 기여도(Degree of Contribution : DOC)라고 할 수 있다.

따라서 퍼지제어기의 추론과정은 입력변수를 각 규칙에 대한 멤버십 값으로 사상시키며, 이것은 곧 출력에 대한 기여도를 사상하는 함수값이라고 할 수도 있다. 여기에 규칙의 결론을 곱하여 제어기 출력을 결정하므로 규칙을 수정하면, 그에 해당하는 제어출력도 수정된다. 본 연구에서는 퍼지제어규칙을 수정하는 방법으로 가중치(Contribution Weight) 개념을 도입한다.

가중치 조정 알고리즘은 제어출력에 영향을 많이 미치는 규칙은 증폭시키고, 작은 영향을 미치는 규칙은 축소시킴으로서 변화하는 환경에 적응하여 제어를 수행한다.

즉 초기에 설계된 퍼지 제어규칙으로 한주기동안 제어동작을 수행한 후, 적용된 각 규칙의 횟수를 기록하고 제어성능평가(performance metric)를 한다. 한 주기 제어를 수행한 후 측정된 각 규칙 적용의 횟수는 플랜트 제어특성을 반영한 것이므로, 미지 플랜트의 제어나 플랜트 주위환경 변화에 따른 파라미터 변화, 외란이 투입되었을 경우의 제어특성을 규칙 적용의 횟수에 근거한 가중치를 조정함으로써 보상할 수 있다. 성능평가는 시스템의 시간영역 응답특성(overshoot, rising

time, settling time)의 요구 허용오차를 기준으로 하여 제어기 응답특성을 비교하여 평가한다.

제어기 응답특성이 요구 허용오차를 만족할 경우는 초기에 설계된 기존의 규칙을 사용하며, 만족되지 않을 경우에는 가중치 수정 알고리즘을 사용한다. 즉, 첫번째 제어주기동안에 적용된 규칙들의 횟수는 플랜트 특성에 따라 최적제어에 필요한 플랜트 요구량이다. 따라서 본 연구에서는 가중치를 0~2 사이의 값을 갖도록 하여, 초기 가중치는 1로 설정하고, 플랜트 제어특성에 따라 증가 혹은 감소시키도록 하였다. 가중치 조정법의 가중치 조정은 다음 식에 의해 수행된다.

$$W(n+1) = W(n) + m(n+1)$$

여기서 $W(n+1)$ 은 현 상태 가중치, $W(n)$ 은 전 상태 가중치 (초기 1로 설정), $m(n+1)$ 은 현 상태 가중치 조정량이며 다음 식으로 정의된다.

$$m(n+1) = \frac{R_i - r_n / r_i}{S_n}$$

- R_i : 전 상태의 모든 규칙들에 대한 적용횟수
- r_n : 전 상태에 적용된 규칙들의 총횟수
- r_i : 전 상태에 적용된 규칙들의 수
- S_n : 샘플링 횟수

현 상태 가중치 조정량 $m(n+1)$ 의 분자항은 각 규칙의 적용횟수와 평균 적용횟수의 편차이며, 샘플링 횟수로 나눔으로서 0~1 사이 값이 된다.

편차가 0인 경우, 현상태 가중치는 전 상태를 유지하고, 0보다 큰 경우는 편차에 비례하여 전 상태 가중치에 더해져서 현 상태 가중치를 증가시키며, 0보다 작은 경우는 편차에 비례하여 전 상태 가중치를 감소시켜 현 상태 가중치로 조정된다.

이러한 과정으로 매 제어 주기마다 수정되며 수정된 가중치는 추론 멤버십값(기어도;DOC)에 곱해져, DOC를 확대 또는 감소시켜 비퍼지화를 통해서 최종 퍼지제어 출력으로 계산된다. 그림 2는 가중치 조정기를 나타낸다.

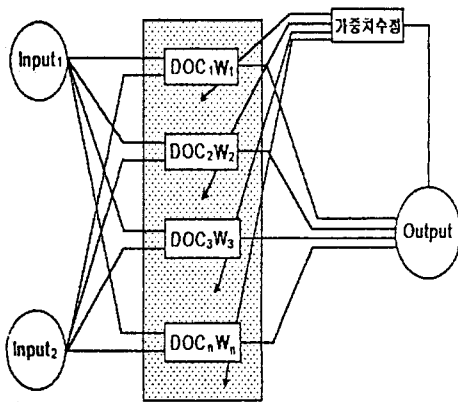


그림 2. 가중치 조정기

이와같이 가중치는 기어도가 낮은 규칙의 추론 멤버십값은 더욱 더 작게 만들며, 기어도가 높은 규칙의 추론 멤버십값은

더욱 크게하여 제어출력에 크게 기여하는 결과가 된다. 그럼으로서, 기존의 SOC (Self - Organizing Controller)[7][8]와 같이 규칙을 수정하는 결과가 되므로 새로운 환경이나, 플랜트의 비선형성, 동특성에 적용하여 성능을 개선하는 효과를 얻을 수 있다.

제어 알고리즘

- 1 단계 : 초기의 퍼지 제어규칙으로 부터 제어
- 2 단계 : 각 규칙의 적용횟수 기록, 저장
- 3 단계 : 제어성능 평가
- 4 단계 : 규칙 적용에 따른 가중치 발생 및 저장
- 5 단계 : 가중치 조정 → DOC 확대 · 축소
- 6 단계 : 단계 2 반복

Ⅲ. 시뮬레이션 결과

본 연구에서는 제안한 적응퍼지 제어기를 DC 서보 전동기 모델에 적용하여 그 타당성을 검증하였다. 시뮬레이션에 사용된 DC 서보 전동기 모델의 전달함수[6]는 다음과 같다.

$$G(s) = \frac{1}{s(s+3.6)}$$

그림 3은 제어 목표가 1로 고정된 기존의 PD형 퍼지 제어기의 출력파형과 적응퍼지 제어기의 가중치를 2회 반복 조정된 출력파형을 나타내며, 가중치 조정 횟수가 증가될수록 상승시간과 정착시간이 개선되는 것으로 나타났다. 이때 샘플링 타임은 20[ms]이고, 제어주기는 5[sec]이다. 그림 4는 PD형 및 가중치 1, 2회 조정된 적응퍼지 제어기의 입력 전압의 변화를 나타낸다. 그림 5와 표 4는 적응퍼지 제어기의 초기 한 주기 제어 동작중에 적용된 규칙 횟수의 히스토그램이고, 이어따라 조정된 가중치 행렬을 나타낸다. 그림 6과 표 5는 가중치 1회 조정후 제어동작중에 적용된 규칙횟수의 히스토그램과 이어따라 조정된 가중치 행렬을 나타낸다. 표 4,5에서 적응퍼지 제어기가 가동되어 규칙의 가중치가 행렬의 중앙으로 편향됨을 볼 수 있는데, 이것은 플랜트의 제어특성이 반영된 것으로 대부분의 입력 변수($E_k, \Delta E_k$)가 이 영역 범위에 존재한다는 것을 의미한다.

Ⅳ. 결론

본 연구에서는 퍼지제어기 동작중에 측정된 각 퍼지 규칙의 적용횟수를 가중치(Contribution Weight)로 사상하여 추론 멤버십값을 매 제어주기마다 수정하여 제어 출력에 대한 기여도를 확대 또는 축소시킴으로서, 플랜트 환경변화에 적용하여 퍼지 제어규칙을 수정하는 가중치 조정법을 제안하였다.

가중치 조정법의 타당성을 검증하기 위해 기존의 PD형 퍼지 제어기를 갖는 DC 서보 전동기에 적용하여 시뮬레이션 한 결과 상승시간(rising time), 정착시간(settling time)이 개선되는 제어 효과를 확인하였다. 즉 가중치로 사상한 적용 규칙의 횟수는 플랜트 제어특성을 반영한 것이므로 가중치 조정에 의한 퍼지제어규칙 수정알고리즘은 타당한 것으로 나타났다.

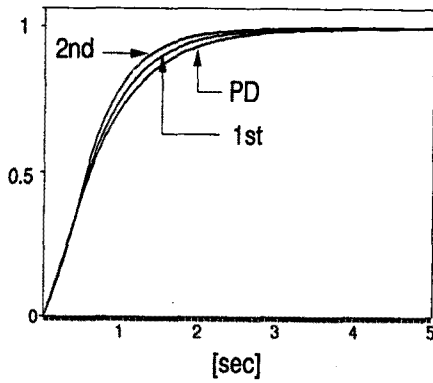


그림 3. 제어목표 1의 출력파형

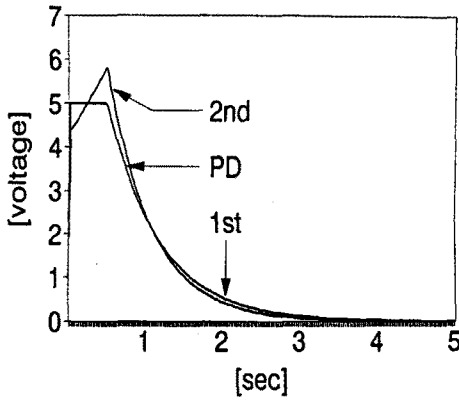


그림 4. 입력전압 변화

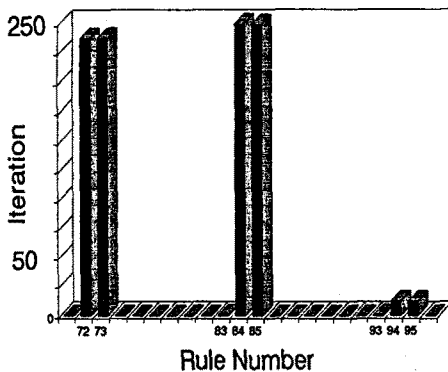


그림 5. 초기 제어동작 중 적용된 규칙

표 4. 1회 조정된 가중치

	-5	-4	-3	-2	-1	0	1	2	3	4	5
-5	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7
-4	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7
-3	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7
-2	0.7	0.7	0.7	0.7	0.7	1.2	1.2	0.7	0.7	0.7	0.7
-1	0.7	0.7	0.7	0.7	0.7	1.2	1.2	0.7	0.7	0.7	0.7
0	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7
1	0.7	0.7	0.7	0.7	0.7	1.16	1.16	0.7	0.7	0.7	0.7
2	0.7	0.7	0.7	0.7	0.7	1.2	1.2	0.7	0.7	0.7	0.7
3	0.7	0.7	0.7	0.7	0.7	0.72	0.72	0.7	0.7	0.7	0.7
4	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7
5	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7

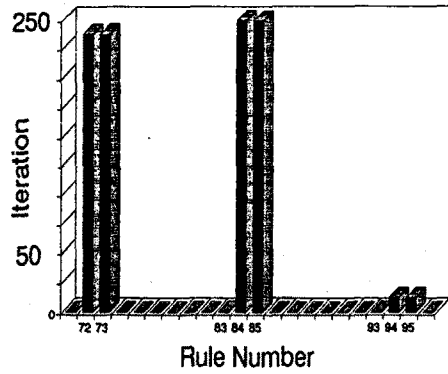


그림 6. 2번째 제어동작 중 적용된 규칙

표 5. 2회 조정된 가중치

	-5	-4	-3	-2	-1	0	1	2	3	4	5
-5	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69
-4	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69
-3	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69
-2	0.69	0.69	0.69	0.69	0.69	1.17	1.17	0.69	0.69	0.69	0.69
-1	0.69	0.69	0.69	0.69	0.69	1.2	1.2	0.69	0.69	0.69	0.69
0	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69
1	0.69	0.69	0.69	0.69	0.69	1.11	1.11	0.69	0.69	0.69	0.69
2	0.69	0.69	0.69	0.69	0.69	0.72	0.72	0.69	0.69	0.69	0.69
3	0.69	0.69	0.69	0.69	0.69	1.17	1.16	0.69	0.69	0.69	0.69
4	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69
5	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.69

참고문헌

- [1] L. A. Zadeh, "Outline of a New Approach to the Analysis of Complex Systems and Decision Process," IEEE Trans. Syst. Man Cybern., vol. SMC-3, pp. 28-44, 1973.
- [2] E. H. Mamdani and S. Appilian, "An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller" Int. J. Man-Machine Stud., vol. 7, pp. 1-13, 1975.
- [3] Y.F.Li and C.C.Lau, "Development of Fuzzy Algorithms for Servo Systems," IEEE Contr. Syst. Mag., vol 9, no. 3, pp. 65-72, 1989.
- [4] T.J. Procyk and E.H. Mamdani, "A Linguistic Self-Organizing Process Controller," Automatica, vol.15, pp. 15-30, 1979.
- [5] C.C.Lee, "Fuzzy Logic in Control System: Fuzzy Logic in Controller Part I", IEEE Transactions on Systems, Man and Cybernetics, vol 20, no.2, pp 404-418. 1990.
- [6] C.C.Lee, "Fuzzy Logic in Control System:Fuzzy Logic Controller, Part II," IEEE Trans. Sys. Man Cybern. vol. SMC-20, no 2, pp. 419-435. 1990.
- [7] M.Braae and D.A.Rutherford, "Theoretical and Linguistic Aspects of the Fuzzy Logic Controller," Automatica, vol.15, no 5, pp. 553-577, 1979.
- [8] B.S.Zhang and J.M.Edmunds, "Self-Organizing Fuzzy Logic Controller", IEEE Proceedings-D, vol. 139, pp. 460-464, 1992.