

# 행렬 표현 유전자 알고리즘을 이용한 퍼지 제어기의 설계

김 동 일, 차 성 민, 강 전 배, 권 기 호  
성균관대학교 전기전자 및 컴퓨터공학부  
전화 031-290-7197 / 팩스 031-290-7197

## A Design of Fuzzy Controllers Using Matrix Encoding Genetic Algorithm

Dong Il Kim, Sung Min Cha, Jun Bae Kang, Key Ho Kwon  
School of Electrical & Computer Engin  
E-mail : calf77@ece.skku.ac.kr

### Abstract

Fuzzy controllers also show good performance in case of the systems being nonlinear and difficult to solve. But these fuzzy controllers have problems which have to decide suitable rules and membership functions. In general we decide those using the heuristic methods or the experience of experts. Therefore, many researchers have applied genetic algorithms to make fuzzy rule automatically. In this paper, we suggest a new coding method and a new crossover method to maintain the good fuzzy rule base and the shape of membership

### I. 서론

유전자 알고리즘은 다윈의 진화 이론에 기초를 두고 개발된 알고리즘으로써 여러 개의 개체가 동시에 병렬적으로 주어진 환경에 따라서 최적의 상태로 향하여 적자 생존의 방법으로 진화하여 궁극적으로는 최적의 상태에 도달하는 현상을 모방한 탐색 알고리즘이다.<sup>[1-3]</sup>

퍼지 이론의 창시자인 자데(L. A. Zadeh)는 기존의 제어 방법으로는 제어하려는 대상이 복잡할수록 제어 행위를 결정하는데 까지 엄청난 노력이 뒤따라야 함을 인식하였다. 그래서, 수치적인 값보다는 “입력 값이

매우 크면 출력 값은 작게 하라”와 같이 감각적으로 제어할 수 있다면 제어 대상의 복잡성이 증가하더라도 제어 값을 결정하는 것은 어렵지 않을 것으로 생각했다.<sup>[4]</sup> 퍼지 논리는 기존의 논리체계에 모호성을 도입하여 논리체계의 사고의 범위를 확장시켰고, 이러한 퍼지 논리에 의한 제어 방법은 복잡한 특성을 가지는 시스템에서 정량적으로 해석하기 어려울 때 효과적으로 이용되었다.

그러나 이러한 퍼지 제어기의 구성에 있어 어려운 점은 규칙 베이스를 구성하는 일이다. 전문가의 지식이나 실험에 의해 규칙 베이스를 구성 할 수 있지만, 제어 대상이 복잡하거나 특별한 환경일 경우, 그 구성에 많은 어려움을 갖는다. 게다가 규칙 베이스를 구성하였다 하더라도 그것이 최적인지를 알 수 없다. 이러한 어려움을 보완하기 위해 규칙 베이스를 자동으로 생성할 수 있는 방법들에 대한 연구가 진행되고 있다.<sup>[5-6]</sup>

본 논문에서는 유전자 알고리즘의 염색체 모양을 퍼지 제어기의 규칙 베이스의 모양과 동일한 Matrix 형태로 코딩하여 진화 해 나가는 행렬 표현 유전자 알고리즘(MEGA, Matrix Encoding Genetic Algorithm)을 제안하고, 이 표현에 알맞는 새로운 교배 연산자를 도입함으로써, 유전자 알고리즘의 빠른 수렴과 성능이 뛰어난 퍼지 제어기의 설계 방법을 제안한다.

2장에서는 선택, 교배, 돌연변이 등의 일반적인 유전자 알고리즘의 기본개념에 대하여 기술하고, 3장에서는 기존의 퍼지 제어 시스템에 대해서 기술한다. 4장에서는 제안하는 MEGA에 대해서 설명하고, 5장에서

는 제안한 방법의 유용성을 보이는 시뮬레이션의 결과를 보이고, 마지막으로 시뮬레이션 결과를 바탕으로 분석과 향후 연구 방향에 대해서 기술한다.

## II. 유전자 알고리즘을 이용한 퍼지 제어기

유전자 알고리즘을 이용한 퍼지 제어기는 수치데이터 뿐 아니라 언어적 변수로 표현된 제어규칙을 직접 사용할 수 있으며 얻어진 해가 최적해를 나타내는 특징이 있다. 이 제어기를 만드는 방법에는 제어규칙을 미리 결정하고 소속함수를 진화시키는 방법, 소속함수를 미리 결정하고 제어규칙을 진화시키는 방법 및 제어규칙과 소속함수를 동시에 진화시키는 방법이 있다. 일반적으로 제어규칙과 소속함수를 동시에 진화시키는 방법에 의하여 가장 좋은 성능의 퍼지 제어기를 설계할 수 있지만, 제어규칙과 소속함수가 염색체에 동시에 표현되어 좋지 못한 제어규칙이 좋은 제어규칙과 잘진화된 소속함수를 크게 변화시킬 수 있으므로 최적해를 얻을 가능성이 감소되는 단점이 있다.

제어규칙과 소속함수를 표현하는 방법은 여러 가지 있는데 본 논문에서는 행렬형태의 제어규칙 그대로를 유전자 알고리즘의 초기형태로 사용하였다. 제어규칙은 0에서 6까지의 디지털(0: NB (Negative Big), 1: NM (Negative Medium), 2: NS (Negative Small), 3: ZO (Zero), 4: PS (Positive Small), 5: PM (Positive Medium), 6: PB (Positive Big))로 표현했고 무작위 초기화법(random initialization), 룰렛휠 선택(roulette wheel selection)법, 제안하는 교배법, 단순 돌연변이(simple mutation)법, 무게중심법(center of gravity method)등을 사용하였다.

## III. 행렬표현 유전자 알고리즘(MEGA)

기존의 유전자 알고리즘에서 염색체를 1차원의 스트링 형태로 표현한데 반해, 제안하는 방법에서는 단일 입력 단일 출력 퍼지 제어기의 규칙 베이스와 동일한 모양인 행렬 형태로 염색체를 구성한다.

표 1에 음영부분은 MEGA의 염색체의 모양을 나타낸 것이다.

표 1 제어규칙의 예

Table 1 Example of rule sets

$\begin{matrix} de \\ e \end{matrix}$	NB	NM	NS	ZO	PS	PM	PB
NB	0	0	2	1	5	6	6
NM	0	2	1	4	4	5	6
NS	2	1	4	3	4	4	5
ZO	1	4	3	3	3	4	1
PS	5	4	4	3	4	1	2
PM	6	5	4	4	1	2	6
PB	6	6	5	1	2	6	6

이러한 MEGA에서는 그 표현 방식과 목적에 맞는 새로운 교배 연산자가 필요하게 되는데, 본 논문에서는 한 교배점을 임의로 선택하여 그 점이 속해 있는 행과 열을 모두 교배하는 행열교배 연산자(RC-Crossover), 그 점이 속해 있는 행만을 교배하는 연산자인 행교배 연산자(R-Crossover), 그리고 그 점이 속해있는 열만을 교배하는 열교배 연산자(C-Crossover)를 제안한다.

### 3.1 행열교배 연산자(RC-Crossover)

행열교배 연산자는 염색체 상의 한 점을 임의로 선택하여 그 점이 속해있는 행과 열을 모두 교환 하는 방식을 취하는 교배 연산자이다.

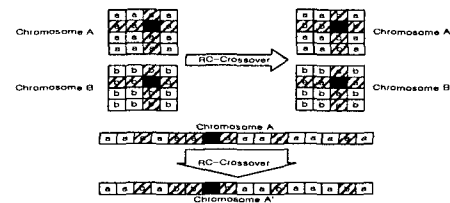


그림 1 행열 교배 연산자

Fig 1 RC-Crossover

그림 1은 행열 교배 연산자의 교배 방법 및 그 효과를 나타낸 것이다. 규칙 베이스에서 좋은 구성부가 행과 열의 형태로 나타난다고 가정했을 때, 좋은 구성부를 파괴하지 않고 두 염색체간의 교배가 일어날 수 있을 뿐만 아니라, 행렬 형태로 표현된 염색체를 스트링 형태의 염색체로 생각할 때, 고전적 유전자 알고리즘(Simple G.A)에서 단일 교배와 같이 한 점만 선택하여 다점 교배와 같은 효과를 얻을 수 있다.

### 3.2 행교배 연산자(R-Crossover)

행교배 연산자는 염색체 상의 한 점을 임의로 선택

## 행렬 표현 유전자 알고리즘을 이용한 퍼지 제어기의 설계

하여 그 점이 속하는 행만을 교환하는 교배 연산자이다.

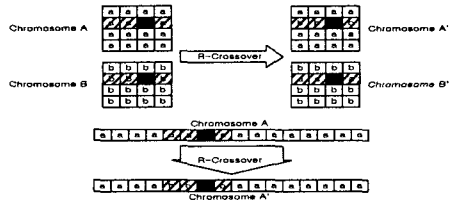


그림 2 행교배 연산자  
Fig 2 R-Crossover

그림 2는 행교배 연산자와 그 효과를 나타낸 것으로서, 그 표현을 스트링으로 생각했을 때, 한 교배점을 선택하여 2점 교배와 같은 효과를 얻을 수 있다. 또한 행으로 구성되어 있는 구성부를 파괴하지 않는다는 장점도 가지고 있다.

### 3.3 열교배 연산자(C-Crossover)

열교배 연산자는 염색체 상의 한 점을 임의로 선택하여 그 점을 포함하는 열만을 교환하는 교배 연산자이다.

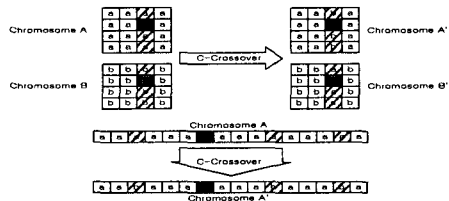


그림 3 열교배 연산자  
Fig 3 C-Crossover

그림 3는 열교배 연산자와 그 효과를 나타낸 것으로서, 열교배 연산자 역시 그 표현을 스트링과 같이 생각한다면 한 점을 선택하여 다점 교배의 효과를 얻을 수 있을뿐만 아니라, 열로 구성되어 있는 구성부를 파괴하지 않는 장점이 있다.

## IV. 시뮬레이션

### 4.1 시뮬레이션 방법

단순 유전자 알고리즘(Simple G.A)의 경우 유전자(chromosome)를 구성하는데 있어 스트링의 형태로 표현하며, 이러한 방법을 퍼지 제어기의 규칙 베이스를 구성하는데 적용할 경우, 구성부 가설을 위배하는 경

우가 발생하게 된다. 본 논문에서는 이러한 유전자 알고리즘의 단점을 극복하기 위하여 행렬 표현 유전자 알고리즘을 사용하고, 그 결과를 알아보기 위하여 원형 이차 시스템의 응답을 이용하여 시뮬레이션하고 그 결과를 비교, 검토한다.

원형 이차 시스템의 페루프 전달함수와 는 다음과 같이 나타난다.

$$G(s) = \frac{w_n^2}{s^2 + 2\xi w_n s + w_n^2}$$

시뮬레이션을 위하여 사용된 파라미터들은 다음과 같다.

표 1 시뮬레이션에 사용된 파라미터  
Table 1 Parameter for Simulation

파라미터	값
Generation	300
Population Size	20
Crossover Rate	0.25
Mutation Rate	0.01
$\xi$	0.25
$\omega_n$	1
Sampling Time	0.001
기대값	0.0098503

알고리즘의 성능을 평가하기 위한 평가 함수는 다음과 같다.

$$Fitness = \frac{1}{1 + \sum_{i=0} Error^2}$$

표 1에서 기대값은 퍼지 제어를 사용하지 않았을 경우 일반적인 원형 이차 시스템의 스텝 응답에 대한 평가값을 구한 경우이다. 퍼지 제어기 설계는 NP-hard 문제로써 최적값을 예상할 수 없으므로, 기대값 이상의 해를 찾는 경우 좋은 해로 생각하고 평가한다.

### 4.2 시뮬레이션 결과

고전적 유전자 알고리즘과 행렬 표현 유전자 알고리즘을 이용하여 각각 다른 10개의 초기 개체군으로 반복 실험하고 그 결과값들의 평균값을 최종 결과값으로 사용하였다. 제어가 없는 상태에서의 원형 이차 시스템의 응답에 대한 평가값인 0.00098503을 기대치로 하여 그 이상의 값을 찾아내는 횟수와 세대수를 비교하였다.

표 2 유전자 알고리즘을 퍼지 제어기 결과  
Table 2 Result of Fuzzy Control Using G.A

	Simple G.A	MEGA with RC-Crossover	MEGA with R-Crossover	MEGA with C-Crossover
발견 횟수	9/10	9/10	9/10	10/10
최초 발견 세대 (평균)	128	53	43	46

표 2는 10개의 다른 초기 개체군으로 실험한 값의 평균을 나타낸 것이다.

각각 다른 10개의 초기 개체군으로 반복 실험한 후 그 결과를 비교해 본 결과 열교배 연산자를 사용한 MEGA는 10번 모두 기대치 이상의 값을 찾았고, 나머지는 각 9번씩 기대치 이상을 찾았다. 이 결과로 유전자 알고리즘을 이용하여 퍼지 제어기의 규칙 베이스를 설계할 경우 좋은 결과를 얻을 수 있다고 말할 수 있다. 또한 기대치 이상의 값을 최초로 찾은 세대수(평균)와 종료 조건 이후의 최대 적합도 값을 비교해 보면 고전적 유전자 알고리즘을 사용하는 것 보다 MEGA를 이용하여 규칙 베이스를 구성하는 경우 더 좋은 결과가 나타남을 알 수 있다.

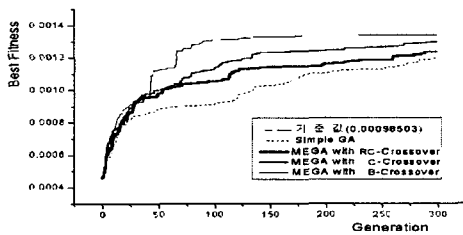


그림 4 최대 적합도 비교  
Fig 4 Comparison of Best Fitness

그림 4와 5는 고전적 유전자 알고리즘과 MEGA를 이용하여 원형 이차 시스템의 제어를 위한 퍼지 제어기를 설계할 때 나타나는 최대 적합도와 평균 적합도를

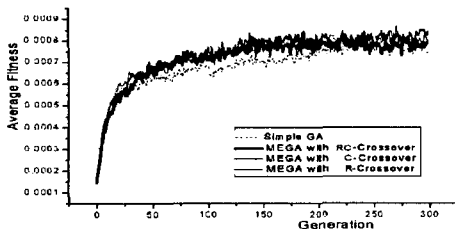


그림 5 평균 적합도 비교  
Fig 5 Comparison of Average Fitness

비교한 것이다.

### V. 결론

제안한 행렬 표현 유전자 알고리즘이 퍼지 규칙 베이스를 생성하는데 있어서 기존의 고전적 방법과 비교하여 빠른 수렴을 할뿐만 아니라 더 좋은 규칙 베이스를 찾는 성능을 발휘함을 알 수 있다.

향후에 연구되어야 할 과제로서 퍼지 제어기의 규칙 베이스를 생성하는데 더욱 적합한 평가 함수에 대한 연구가 필요할 것이다. 또한 퍼지 제어기에 따른 규칙 베이스의 구성부가 어떻게 존재하는지에 대한 연구도 필요할 것이며, 제어 초기와 후기에 사용되는 구성부를 나누어서 최적화 하는 방법에 대한 연구도 필요할 것이다.

또한 단일 입력 단일 출력 시스템뿐만 아니라 2개 이상의 입력을 가지는 시스템 제어에 사용되는 퍼지 제어기의 규칙 베이스를 최적화 하는데 필요한 큐빅 형태의 표현을 사용하는 유전자 알고리즘에 대한 연구도 이루어져야 할 것이다.

### 참 고 문 헌

- [1] Zbigniew Michalewicz, "Genetic Algorithms + Data structures = evolution Programs", Springer-Verlag, New York, Second Extended Edition, 1994.
- [2] Lance Chambers, "Practical Handbook of GENETIC ALGORITHMS, Applications Volume I", CRC Press, 1995.
- [3] Darrell Whitley, "A Genetic Algorithm Tutorial", Colorado State University Technical Report CS-93-103, 1993.
- [4] 엄정국, 원성현, 기초 퍼지 이론과 응용 퍼지 시스템, 정보시대출판부, p.11, 1992.
- [5] C. L. Karr, "Design of an adaptive fuzzy logic controller using genetic algorithm," in Proc. the Fourth Int. Conf. Genetic Algorithms, p. 450-457, 1991.
- [6] P. Thrift, "Fuzzy logic synthesis with genetic algorithms," in Proc. Fourth Int. Conf. Genetic Algorithms, p. 509-513, 1991.