

유전알고리즘을 이용한 최적퍼지 규칙베이스 시스템의 설계

김종율

부산광역시 사상구 주례2동 산69-1 동서대학교 컴퓨터정보공학부

xmaskjr@gdsu.dongseo.ac.kr

Design of Optimal Fuzzy Rule-base Systems with Genetic Algorithm

Jong Ryul Kim

Division of Computer & Information Engineering, Dongseo University

요 약

본 논문은 퍼지 분류를 위한 퍼지 규칙베이스 시스템에 대한 최적화 해법으로서 유전 알고리즘에 대해 살펴본다. 즉 퍼지 규칙베이스를 이용하는 퍼지 분류 시스템을 최적화를 하는 유전 알고리즘을 제안한다. 본 논문에서 다루는 최적화는 추출되는 퍼지 규칙의 수와 퍼지 분류 시스템의 입력 패턴을 정확하게 분류하는 지에 대한 성능을 포괄적으로 수행하는 것을 의미한다. 마지막으로 본 논문에서 제안하는 유전 알고리즘을 이용하여 수치실험을 수행하고 그 결과를 통해 제안하는 알고리즘의 유효성과 효율성을 생성된 퍼지 규칙의 수와 퍼지 분류 시스템의 성능의 관점에서 논의한다.

1. 서 론

근래 수십 년간 많은 수의 연구자들이 자동차 자동조정 시스템, 데이터 마이닝의 분류 시스템, 트럭 후진 자동제어 시스템 등과 같은 다양한 시스템을 제어하기 위해 퍼지 제어 시스템에 대한 연구를 해 왔다. 특히, 이런 다양한 제어 시스템에 퍼지 시스템을 적용하기 위해 주로 퍼지 if-then 규칙들이 채택되고 있으며 일반적으로 관련 시스템의 전문가들에 의해 if-then 규칙들이 결정되어 왔다 [1]. 최근의 연구들은 측정된 실제계의 데이터로부터 관련 시스템의 전문가의 도움 없이도 자동으로 퍼지 if-then 규칙들을 생성하는 것에 집중되고 있다 [2,3]. 이러한 자기 학습법들 (Self-learning Methods)은 최강강하법 (Gradient Descent Method), 신경망 기반의 방법, 유전 알고리즘 등과 같은 해법들을 이용하여 퍼지 규칙들의 퍼지 집합에 대한 멤버십 함수를 선택하기 위해 주로 제안되었다 [9].

1960년대 이후로 어려운 최적화 문제들을 위한 강력한 알고리즘들을 개발하기 위해 살아 있는 생명체를 모방하는 것에 대한 관심들이 증대되고 있다. 그러한 기술들에 대해 일반적으로 널리 받아 들여 지고 있는 통칭은 진화 연산법 (Evolutionary Computation Method)이 있다. 이러한 부류들 중에 가장 널리 알려진 알고리즘들에는 유전 알고리즘 (Genetic Algorithm), 진화 전략 (Evolution Strategies), 진화 프로그래밍 (Evolutionary Programming), 유전 프로그래밍 (Genetic Programming) 등이 있다. 이것들 중에 유전 알고리즘은 그 탐색 방법을 자연 현상을 모델링한 형태로 취하고 있으며 실제계의 응용에서 빈번히 발생하는 대규모의 최적화 문제들을 위한 확률적 탐색 알고리즘으로 개발되었다 [7,8]. 또한 위에서 나타난 알고리즘들의 다양한 특징들을 결합시키는 매우 많은 하이브리드 방법들도 존재하고 있다. 최근에 다양한 유전 알고리즘들이 퍼지 if-then 규칙들의 생성과 if-then 규칙들에서의 퍼지집합을 위한 멤버십 함수의 결정을 위해 널리 제안되고 있다 [3,9].

유전 알고리즘 기반 기계 학습 (Machine Learning) 방법들은 크게 두개의 형태로 분류할 수 있는데, 그것은 미시간식 접근법 (Michigan Approach)과 피츠버그식 접근법 (Pittsburg Approach)이다. 미시간식 접근법은 각 규

칙을 분류자(classifier)로 불리는 개체(individual)로 나타내고 반대로 피츠버그식 접근법에서는 유전 알고리즘의 유전 과정에서의 개체를 규칙 집합 (rule set)으로 해석하여 사용한다. 퍼지 if-then 규칙들을 생성하고 멤버십 함수들을 결정하기 위한 대부분의 유전알고리즘 기반의 방법들은 퍼지 if-then 규칙 집합 (퍼지 규칙 베이스)을 유전 알고리즘에서의 개체로 다루는 피츠버그식 접근법으로 분류될 수 있다. 본 논문에서의 제안하는 알고리즘도 역시 피츠버그 접근법으로 분류될 수 있다.

본 논문에서는 전문가의 도움 없이 실제계에서 측정된 데이터로부터 퍼지 if-then 규칙들을 자동적으로 생성하기 위해 퍼지 규칙 베이스 시스템을 최적화하는 유전 알고리즘을 제안한다. 즉 퍼지 분류 시스템의 효과를 극대화하면서 퍼지 if-then 규칙의 수를 최소화하는 유전 알고리즘을 제안한다. 마지막으로 제안한 알고리즘은 수치 실험을 통해 퍼지 분류 시스템에서 사용된 퍼지 규칙의 수와 정확히 분류된 패턴의 수 (퍼지 분류 시스템의 성능)의 관점에서 효율적이고 효과적임을 증명한다.

2. 문제 설명

본 논문에서는 l 개의 학습 패턴들 (레이블된 패턴들)이 수치 데이터로서 주어지고 m 차원의 속성 값들과 n 개의 클래스들을 가진 패턴 분류 문제를 다룬다. 여기서 레이블된 입력 패턴들을 $p_i = (p_{i1} p_{i2} \dots p_{im})$, $i = 1, 2, \dots, l$ (속성 값 p_{ij} 는 범위 $[0, 1]$ 에서의 실수치 임)으로 나타낸다. 이것은 모든 속성 값들이 컴퓨터를 이용한 시뮬레이션을 위해 범위 $[0, 1]$ 에서의 실수치로 정규화 되어야만 한다는 것을 의미한다.

2.1 퍼지 규칙 (Linguistic Fuzzy Rules)

m 차원 속성 값과 l 개의 클래스를 가진 패턴 분류 문제를 위해 본 논문에서는 다음과 같은 퍼지 규칙을 이용한다:

R_k : if p_{i1} is M_{k1} and ... and p_{im} is M_{km} then class c_k
with g_{c_k} , $k = 1, 2, \dots, q$,

여기서 R_k 는 퍼지 시스템의 k 번째 퍼지 규칙이고 $p_i = (p_{i1} p_{i2} \dots p_{im})$ 는 m 차원 패턴 벡터의 i 번째 패턴이며 M_{kj} 는 k 번째 퍼지 규칙의 j 번째 속성을 위한 퍼지 집합이고 c_k 는 결과 클래스를, g_{c_k} 는 범위 $[0,1]$ 내의 값을 가지는 확실성 정도(certainty grade)를, 마지막으로 q 는 퍼지 규칙의 총 개수를 의미한다. 각 퍼지 규칙의 결과 클래스 c_k 와 확실성 정도 g_{c_k} 는 2.2절에 나타난 것과 같이 주어진 패턴들로부터 휴리스틱 규칙 생성 프로시저에 의해 손쉽게 구할 수 있다 [4]. 퍼지 분류 시스템에서 이용되는 멤버십 함수들에 대해서는 휴리스틱 규칙 생성 프로시저를 적용하기 전에 언어적 항목에 관련된 전반부의 퍼지 집합들을 위한 멤버십 함수를 먼저 규정해야 한다. 하지만 선택된 퍼지 if-then 규칙들의 언어적 해석은 항상 쉬운 것은 아니다. 따라서 본 논문에서는 그림 1에서 나타난 것처럼 후보 퍼지 if-then 규칙들의 전반부 퍼지 집합에 대한 멤버십 함수들을 6개의 언어적인 값으로 제한하여 사용한다 (즉 S: small, MS: medium small, M: medium, ML: medium large, L: large, DC: don't care). 이러한 언어적인 값들은 모든 속성 값들을 위한 수치 시뮬레이션에서 사용되며 전반부 퍼지 집합 M_{kj} 는 6개의 언어적인 값들 중에 하나가 된다.

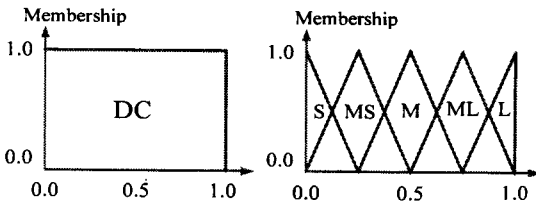


그림 1. 언어적인 규칙을 위한 퍼지 집합들

2.2 규칙 생성법 (Rule Generation Method)

퍼지 규칙의 생성을 위해서 우선 퍼지 규칙 R_k 에 대한 p_i 의 적합도를 다음의 수식을 이용하여 계산한다:

$$\mu_k(p_i) = \mu_{k1}(p_{i1}) \cdot \mu_{k2}(p_{i2}) \cdot \dots \cdot \mu_{km}(p_{im}),$$

여기서 μ_{kj} 는 퍼지 집합 M_{kj} 의 멤버십 함수를 의미하고 그림 1에 나타나 있는 6개의 언어적인 값들 중의 하나를 나타낸다. 적합도의 계산을 수행한 후에 주어진 퍼지 집합 M_{kj} 에 대하여 결과 클래스 c_k 와 확실성 정도 g_{c_k} 를 다음과 같이 결정한다.

Step 1: 각 클래스에 대해 퍼지 규칙 R_k 의 적합도 총합을 다음 수식을 이용하여 계산한다:

$$\beta_c = \sum_{p_i \in Class\ c} \mu_k(p_i), \quad c = 1, 2, \dots, n,$$

여기서 β_c 는 퍼지 규칙 R_k 에 대한 주어진 학습 패턴들이 c 클래스에 속하는 적합도의 총합을 의미하고 $p_i \in Class\ c$ 는 패턴 p_i 가 c 클래스로 분류될 수 있는 것을 의미한다.

Step 2: 퍼지 규칙 R_k 의 결과 클래스 c_k 는 적합도 총합을 최대치를 가진 클래스로 결정된다. 즉 c_k 는 다음 수식을 만족하는 클래스 \hat{c} 로 결정된다:

$$\beta_{\hat{c}} = \max\{\beta_1, \beta_2, \dots, \beta_n\}.$$

만일 클래스 \hat{c} 로 유일하게 결정할 수 없다면 (즉 두 개 또는 그 이상의 클래스들이 동일한 최대치를 가지는 경우) c_k 에 e 를 할당하며 여기서 e 는 공 클래스

(empty class)를 의미한다. 공 클래스로 할당된 퍼지 규칙이 새로운 패턴들에 대한 분류 과정에 어떠한 영향도 끼치지 않기 때문에 본 논문에서는 이러한 규칙을 더미 규칙으로 간주한다.

Step 3: 모든 더미 규칙들에 대한 확실성 정도 g_{c_k} 는 0로 설정하고 ($g_{c_k} = 0$) 그 이외의 규칙들에 대해서는 확실성 정도 g_{c_k} 는 다음 수식과 같이 계산된다:

$$g_{c_k} = \frac{\beta_c - \bar{\beta}}{\sum_{c \neq \hat{c}} \beta_c}, \quad \text{여기서 } \bar{\beta} = \frac{\sum_{c \neq \hat{c}} \beta_c}{n-1} \text{이다.}$$

확실성 정도는 $[0, 1]$ 범위내의 실수치로 나타낼 수 있으며 $c \neq \hat{c}$ 인 경우에 대해 $\beta_c > 0$ 이고 $\beta_{\hat{c}} = 0$ 인 조건일 때 확실성 정도는 최대가 된다. 이것은 퍼지 규칙 R_k 에 의해 적합화되는 모든 패턴들이 동일한 클래스에 속하는 경우 확실성 정도 g_{c_k} 는 1로 계산된다는 것을 의미한다. 반대로 c 개 클래스들 각각의 적합도 총합이 모두 서로 동일할 경우 확실성 정도는 거의 0로 계산된다 (최소의 확실성 정도).

2.3 퍼지 규칙 최적화 문제

생성된 q 개의 퍼지 분류 규칙 집합을 $Z^* = \{R_k | k=1, 2, \dots, q\}$ 로 나타낸다. 집합 Z^* 내의 모든 퍼지 분류 규칙들은 규칙 선택 문제에서의 후보 규칙들로 이용된다. 본 논문에서의 규칙 최적화 문제는 고성능이면서도 간결한 분류 시스템 Z 를 발견해 내기 위해 규칙 집합 Z^* 으로부터 적은 수의 퍼지 규칙들을 선택하는 것을 의미한다. 본 논문에서의 퍼지 규칙 최적화 문제는 두 종류의 목적함수, 즉 적은 수의 퍼지 규칙들 및 고성능의 분류 능력을 가진다. 따라서 본 논문에서의 규칙 선택 문제는 다음과 같이 정식화 할 수 있다:

$$\max w_1 f_1(Z) - w_2 f_2(Z),$$

$$\text{s. t. } Z \subseteq Z^*$$

여기서 w_1 과 w_2 는 두 종류의 목적함수를 하나로 만들기 위해 각 목적함수에 할당되는 상수로서 가중치(weights)를 의미하고 양수 값을 가진다. 또한, $f_1(Z)$ 는 분류 성능 (즉 규칙 집합 Z 에서의 퍼지 규칙들에 의해 정확하게 분류해내는 패턴들의 수)을, $f_2(Z)$ 는 퍼지 규칙 집합 Z 에서의 퍼지 규칙들의 수를 각각 의미한다.

3. 유전 알고리즘

본 논문에서 제안하는 유전 알고리즘은 참고문헌 [7, 8]에 기술되어 있는 단순 유전 알고리즘 (SGA: Simple GA)을 퍼지 규칙 최적화 문제에 관한 정보를 고려하여 문제에 맞도록 수정한 형태이다.

3.1 유전자 표현법 및 초기화

본 논문에서 사용되는 염색체의 유전자 표현법은 $v = (v_1 v_2 \dots v_q)$ 와 같은 벡터로 표현할 수 있다. 여기서 q 는 입력과 출력을 조합하여 생성할 수 있는 퍼지 규칙들의 개수이다. 그리고 v_k 는 퍼지 규칙 집합에서 k 번째 규칙이 퍼지 규칙에 포함되어 분류 과정에서 사용되는지 여부를 나타내고 다음의 3가지 값 중에 하나를 취하게 된다:

$$v_k = \begin{cases} 1 & k\text{-번째 규칙이 규칙 집합 } Z \text{에 포함되면,} \\ 0 & k\text{-번째 규칙이 더미 규칙이면,} \\ -1 & k\text{-번째 규칙이 규칙 집합 } Z \text{에 포함되지 않으면} \end{cases}$$

더미 규칙들은 새로운 패턴들에 대한 분류 처리 과정에서 아무런 영향을 끼치지 않기 때문에 퍼지 규칙 집합 Z 에서 제거되어야 하며 벡터 $v = (v_1, v_2, \dots, v_q)$ 는 $Z = \{R_k | v_k = 1; k = 1, 2, \dots, q\}$ 형태로 복호화 (decoding)될 수 있다.

집단(Population)에서의 각 염색체 벡터 $v_i(t=1, \dots, pop_size)$ 는 더미 규칙들에는 0을 할당하고 그 외의 규칙들에는 0.5의 확률로 랜덤하게 1 또는 -1을 할당하여 생성해 낸다. 여기서 pop_size 는 유전 알고리즘의 탐색 과정상에서 각 세대에 대한 집단의 크기 (염색체 벡터들의 개수)를 의미한다.

3.2 평가 함수

본 논문에서는 다음과 같은 형태로 집단에서의 염색체 v_i 를 평가한다.

$$eval(v_i) = w_1 \cdot f_1(v_i) - w_2 \cdot f_2(v_i),$$

여기서 w_1 과 w_2 는 각 목적함수에 할당되는 상수로서 가중치 (weights)를 의미하고 양수 값을 가진다. 또한 $f_1(v_i)$ 는 3.2.1절에서 기술한 프로시저에 의해 결정되는 분류 성능을, $f_2(v_i)$ 는 3.2.2절에 의해 계산되는 퍼지 규칙 집합에서의 퍼지 규칙들의 수를 각각 의미한다.

3.2.1 새로운 패턴 분류를 위한 퍼지 추론

퍼지 규칙의 규칙 집합을 Z 라고 가정하면 새로운 패턴 p_i 는 다음과 같은 과정을 통해 분류된다.

Step 1: 다음의 수식을 통해 c 클래스 ($=1, 2, \dots, n$)를 위한 α_c 를 계산해 낸다:

$$\alpha_c = \max \{ \mu_k(p_i) \cdot gc_k | c_k = c \text{ and } R_k \in Z \},$$

여기서 $\mu_k(p_i)$ 는 퍼지 규칙 R_k 에 대한 새로운 패턴 p_i 의 적합도를 의미하고 α_c 는 퍼지 규칙의 후반부에 있어서의 c 클래스를 위한 확실성 정도와 적합도를 곱한 것들 중에 최대값을 나타낸다.

Step 2: α_c 값들 중에서 최대치를 다음과 같이 찾아낸다:

$$\alpha_c = \max \{ \alpha_1, \alpha_2, \dots, \alpha_n \}.$$

만일 두개 이상의 클래스들이 동일한 최대치를 가지고 있다면 새로운 패턴 p_i 의 분류를 실패하게 되며 그렇지 않은 경우에는 α_c 에 의해 결정되는 \hat{c} 를 패턴 p_i 에 할당한다.

따라서 새로운 패턴 p_i 는 적합도 $\mu_k(p_i)$ 와 확실성 정도 gc_k 의 곱의 결과를 최대로 가지는 퍼지 규칙에 의해 분류되며 위에서 기술한 프로시저를 이용하여 정확하게 분류된 패턴들의 개수를 알아냄으로써 분류 성능 함수인 $f_1(\cdot)$ 을 계산해 낼 수 있다.

3.2.2 퍼지 규칙 개수

또한 본 논문에서의 퍼지 규칙 개수를 계산하는 것은 염색체 내의 표현형이 1로 표시되는 것들의 개수를 알아봄으로써 퍼지 규칙들의 개수 ($f_2(\cdot)$)를 구할 수 있다.

3.3 선택

GA에서 선택연산은 매우 중요한 역할을 한다. 교차 및 돌연변이 같은 유전자 연산들은 해 공간에서의 탐색을 위한 탐험 (exploration) 연산이라고 한다면 선택연산은 진화과정을 인도하는 GA를 위한 개척 (exploitation) 연산이라고 할 수 있다. 본 논문에서 사용된 선택 연산은 자유로이 해 공간을 탐색하도록 하기 위해 토너먼트 선택법과 엘리트리스트 (elitist) 방법을 혼합한 형태이다.

토너먼트 선택은 n 개의 염색체를 임의로 선택하고 이들 중에서 가장 적응도가 높은 개체를 다음세대의 집단에 포함시킨다. 다음 세대의 염색체들을 모두 얻을 때까지 이 과정을 반복한다. 이때 선택된 염색체를 현재 집단에서 제거하지 않는다. 즉, 한 염색체가 여러 번 선택될 수 있다. 본 논문에서는 n 을 5로 설정하여 이용한다. 선택 압력은 토너먼트 크기인 n 의 값에 따라 좌우되는데 n 이 크면 적합도가 낮은 염색체들이 다음 세대에서 선택되는 기회가 적어진다. 이러한 토너먼트 선택은 최근에 주로 많이 사용되는 선택법 중에 하나이고 구현이 쉬우며 병렬 구조로 수행될 때 용이하고 선택 압력을 쉽게 조절할 수 있다는 장점을 가지고 있다. 본 논문에서 사용하는 선택법을 다음의 프로시저로 설명할 수 있다.

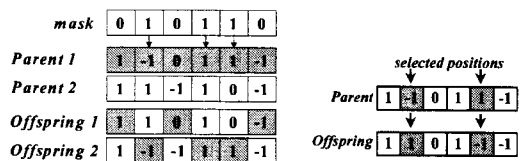
프로시저: 선택법

- Step 1:** 먼저 엘리트리스트 (elitist) 방법으로 이전 세대의 집단의 최선해를 다음세대의 집단으로 선택한다.
- Step 2:** 먼저 이전 세대의 집단 중에서 n 개를 선택한다.
- Step 3:** 선택된 n 개 중에서 가장 높은 적합도를 가진 하나의 염색체를 선택한다.
- Step 4:** 다음 세대의 염색체들을 모두 얻을 때까지 Step 2와 3을 반복한다.

이런 선택연산을 이용하여 최선의 염색체를 현재의 세대에서 다음세대로 유지할 수 있다.

3.4 유전자 연산자들

그림 2에서와 같이 균등교차 (uniform crossover)이라고도 불리는 다점 교차 (multi-point crossover)를 이용한다. 이런 형태의 교차 연산은 임의로 두 부모 염색체를 선택한 후 두 부모 염색체 중에 임의의 위치에 있는 유전자를 자식 염색체의 유전자로 복사하여 자식 염색체를 생성해 나간다. 이러한 균등교차는 우선 교차를 위해 임의로 0 또는 1의 마스크 열을 염색체의 길이만큼 생성한 후 한쪽 부모 염색체로 (Parent1)부터 마스크가 0인 유전인자를, 다른 쪽 부모 염색체 (Parent2)로부터는 마스크가 1인 유전인자를 복사해 하나의 자식 염색체 (Offspring1)를 생성하고 다른 자식 염색체 (Offspring2)는 앞서와 반대로 하여 생성하는 유전 연산자이다.



(a) 다점 교차 (b) 돌연변이

그림 2. 유전 연산자들

본 논문에서의 돌연변이 연산은 그림 2에서 나타낸 것과 같이 두개의 다른 돌연변이율 (P_{m1} 과 P_{m2})을 가지고 임의로 염색체의 표현형을 변환하는 연산이다. P_{m1} 은 1에서 -1로 변이되는 것을 위한 돌연변이율이고 P_{m2} 는 -1에서 1로의 변이를 위한 돌연변이율이다. 유전 알고리즘 동작 과정상에서 언어적인 분류 규칙들의 수를 줄여나가기 위해서 본 논문에서는 P_{m2} 보다도 P_{m1} 을 더 높게 설정하였다.

4. 수치 실험

본 논문에서 제안하는 알고리즘의 성능을 알아보기 위해 2

종류의 문제를 가지고 수치 실험을 수행한다. 최적화를 위한 프로그램은 Visual Studio 2005상의 C++로 작성되었고 실행은 펜티엄 IV 3.2GHz의 PC에서 수행되었다.

4.1 수치예 1

우선 2차원 속성 값을 가진 2개의 클래스 분류 문제를 대상으로 한다 [9]. 수치 실험은 다음과 같은 유전 알고리즘 파라미터로 설정해 수행되었다:

$$w_1 = 10, w_2 = 1, pop_size = 50, P_c = 1.0, P_{m1} = 0.5, P_{m2} = 0.05, max_gen = 200,$$

여기서 max_gen 은 유전 알고리즘의 최대 반복횟수를, P_c 는 교차율을 의미한다.

모든 가능한 퍼지 규칙 집합들로부터 의미 있는 적은 수의 규칙을 선택하는 규칙 베이스 최적화를 위해 제안된 유전 알고리즘을 수치예 1에 적용한 결과는 표 1에 요약되어 있다. 제안 알고리즘을 통해 구해낸 최선의 결과는 다음과 같이 3개의 분류 규칙과 100%의 분류 성능을 가진 것을 얻었다.

- R_1 : if p_1 is DC and p_2 is DC then class 1 with $g_{c_1} = 0.392562$,
- R_6 : if p_1 is DC and p_2 is L then class 2 with $g_{c_6} = 0.878788$,
- R_{19} : if p_1 is M and p_2 is DC then class 2 with $g_{c_{19}} = 0.811189$.

4.2 수치예 2

두 번째 수치예에서는 분류 규칙의 선택을 위해 널리 알려진 iris꽃 데이터에 제안한 유전 알고리즘을 적용해 보았다. iris 꽃에는 setosa, versicolor, virginica의 3종류가 있으며 꽃잎과 꽃받침의 길이와 폭을 이용하여 구분해 낼 수 있다.

이러한 iris꽃 데이터는 꽃잎 길이, 꽃잎 폭, 꽃받침 길이, 꽃받침의 폭의 4차원의 입력 속성 값과 3종류의 클래스를 가진 분류 문제로 나타낼 수 있으며 Anderson [5]에 의해 수집된 iris꽃 데이터로부터 150개의 임의로 생성한 샘플들을 입력데이터 집합으로 해서 수치실험을 수행하였다.

표 1. 수치실험의 결과

Example		$f_1(Z)$	$f_2(Z)$	Time(sec.)
1	Avg.	121	3	41.8
	STD.	0	0	1.4
	Best	121	3	40
	Worst	121	3	46
2	Avg.	137.3	5.8	1029.3
	STD.	7.0	1.6	47.5
	Best	144	5	917
	Worst	122	7	1092

STD.: Standard Deviation, Avg.: Average Time: Computation Time (second)

수치예 1에서의와 같은 유전 알고리즘 파라미터를 가지고 수치 실험을 시행한 결과, 다음과 같은 5개의 퍼지 규칙과 96%의 분류 성능을 가진 결과를 얻을 수 있었으며 표 1에 수치실험의 결과를 요약해 두었다.

- R_6 : if p_1 is DC and p_2 is DC and p_3 is S and p_4 is DC then class 1 with $g_{c_6} = 1.000000$,
- R_{20} : if p_1 is DC and p_2 is DC and p_3 is M and p_4 is MS then class 2 with $g_{c_{20}} = 1.000000$,
- R_{28} : if p_1 is DC and p_2 is DC and p_3 is ML and p_4 is ML then class 3 with $g_{c_{28}} = 0.773386$,
- R_{782} : if p_1 is M and p_2 is M and p_3 is ML and p_4 is MS then class 1 with $g_{c_{782}} = 1.000000$,

R_{1080} : if p_1 is L and p_2 is DC and p_3 is SM and p_4 is DC then class 3 with $g_{c_{1080}} = 1.000000$.

5. 결론

본 논문에서는 퍼지 규칙베이스를 가진 퍼지 분류 시스템의 최적화를 위해 퍼지 규칙베이스 시스템에 대한 유전 알고리즘 기반의 해법을 제안하였다. 본 논문에서의 퍼지 규칙 집합을 이용하는 퍼지 분류 시스템에 대한 최적화 문제는 선택된 퍼지 규칙 개수의 최소화와 추출해 낸 퍼지 규칙들을 이용하여 학습 패턴들을 정확하게 분류할 수 있는 패턴의 개수로 정의되는 분류 성능을 최대화를 포함한다.

제안 알고리즘이 유효성과 효과성을 파악하기 위해 2 종류의 수치예를 이용하여 수치실험을 수행하였다. 수치 실험의 결과를 통해 제안 알고리즘이 퍼지 분류 시스템에서 사용되는 퍼지 규칙의 수와 정확하게 분류되는 패턴들 개수의 관점에서 유효하고 효과적임을 알 수 있다.

앞으로의 연구에서는 분류 성능을 향상시키고 퍼지 분류 시스템에서의 퍼지 규칙들의 수를 최소화하기 위해서 본 논문에서 제안한 유전 알고리즘과 전통적인 학습 방법들을 합성하는 하이브리드 해법을 개발하도록 할 계획이다.

참고문헌

- [1] H. Xu and G. Vukovich, "Fuzzy evolutionary algorithms and automatic robot trajectory generation", *Proc. Of First IEEE Conference on Evolutionary Computation*, Piscataway, pp.595-600, 1998.
- [2] S. Horikawa, T. Furahashi, S. Okuma, and Y. Uchikawa, "Composition Methods of Fuzzy Neural Networks and their Performance Evaluation", *Proc. of IIZUKA' 98*, pp.183-184, 1998.
- [3] C. G. Lim, Y. M. Jung, and E. K. Kim, "A Genetic Algorithm for generating optimal fuzzy rules", *International Journal of KIMICS*, Vol7, No.4, pp.765-778, 2003 (Korean).
- [4] H. Ishibuchi, K. Nozaki, and H. Tanaka, "Pattern classification by distributed representation of fuzzy rules", *IEEE International Conference on Fuzzy Systems*, pp.643-650, 1992.
- [5] E. Anderson, "The irises of the Gaspé peninsula", *Bulletin of the American Iris Society* 59, 2-5, 1935.
- [6] R.A. Fisher, "The use of multiple measurements in taxonomic problems", *Annals of Eugenics* 7, 179-188, 1936.
- [7] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Design*, John Wiley & Sons, New York, 1997.
- [8] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Optimization*, John Wiley & Sons, New York, 2000.
- [9] T. Murada, *Genetic Algorithms for Multiobjective Optimization*, Doctoral Thesis, Osaka Prefecture University, 1997.
- [10] L. B. Booker, D. E. Goldberg, and J. H. Holland, "Classifier systems and genetic algorithms", *Artificial Intelligence*, Vol.40, No.1-3, pp.235-293, 1989.