

원환체형 모집단 유전자 알고리즘

강태원^o

강릉대학교 컴퓨터학과
twkang@kangnung.ac.kr

Genetic Algorithm with Torus-Form Population

TaeWon Kang^o

Dept. of Computer Science, Kangnung National University

요 약

전형적인 단순 유전자 알고리즘은 한 개의 모집단으로 구성되며, 진화 과정이 거듭되면 모집단의 개체들은 한 개의 전역해로 수렴하게 된다. 그러나, 많은 문제들은 여러 개의 최적해를 가질 수 있으며, 그것들 모두를 찾는 것이 중요한 경우가 많다. 이 논문에서는 모집단을 원환체(Torus)로 구성하고 개체에 이웃의 개념을 부여하여 모집단이 최적해 집단으로 수렴하는 유전자 알고리즘의 변형을 연구한다. 제안한 방법은 개체사이에 이웃이라는 개념을 부여함으로써 다수의 해를 동시에 찾는다는 생각을 넘어서 다양한 변형 유전자 알고리즘에 대한 새로운 모델이 될 것으로 기대된다.

1. 서론

유전자 알고리즘은 자연선택과 유전자에 기초를 둔 일종의 탐색 알고리즘이다. 이것은 탐색 공간에 대한 어떠한 지식도 사용하지 않으면서도 알고리즘이 간단하고, 강하며(robust), 또한 일반적이기 때문에 많은 응용분야에서 주목을 받고 있다. 자연은 제한된 자원에 대한 개체간의 경쟁 시에 적응성이 강한 개체가 더 많이 살아남을 수 있게 하는 것으로 알려져 있다. 또한, 그들의 상대적인 우수성은 유전자로 특정 지워지며 이들은 다음 세대에 전달되어 결국 우수한 개체의 우수한 특성이 세대를 건너가며 유지되는 것이다.

일반적으로 단순 유전자 알고리즘은 한 개의 모집단을 구성하는 것으로 시작된다. 여기서 모집단 내의 개체는 문제에 대한 잠정적인 해를 나타내는 것이며, 초기 모집단 내의 개체는 임의로 정한다. 초기 모집단이 생성되면 각 개체를 평가하여 그들의 상대적 적합도를 계산하고, 그 값에 근거하여 다음 세대에 자손을 남길(즉, 자신의 유전자를 다음 세대에 넘길) 개체를 선택한다. 선택된 개체들은 자신의 유전자를 모두 전달하거나, 교차를 통하여 다른 개체와 부분적으로 혼합된 유전자를 전달하거나, 돌연변이를 통하여 자신의 것과 다른 유전자를 포함하도록 전달한다. 그렇게 만들어진 개체들로 구성된 모집단이 다음 세대를 형성하는 것이다. 이러한 진화 과정을 되풀이하면 모집단의 개체들을 적합도가 높은 쪽으로 수렴하게 된다. 그리고 그러한 과정 중에 가장 적합도가 높은 개체를 최적해로 사용한다.

이러한 단순 유전자 알고리즘은 일반적으로 하나의 최적해를 찾는 경우에만 효과적이다. 실제로 단순 유전자 알고리즘

을 적용하면 세대가 거듭되면서 모집단 내의 개체들은 거의가 유사한 유전자를 갖게된다[1]. 그러나, 실세계의 많은 문제들은 여러 개의 최적해를 가질 수 있으며, 동시에 그것들 모두를 찾는 것이 중요한 경우가 많다.

단순 유전자 알고리즘에서 모집단이 하나의 최적해로 수렴하는 이유는 각 개체들이 전역적으로 유전자를 교환하기 때문이다. 이 논문에서는 모집단 내 개체들에 이웃의 개념을 부여하여 그것들이 지역적으로 경쟁하고 결과적으로 지역적으로 유전자를 교환하도록 하여 모집단이 여러 개의 최적해를 포함하며 수렴하도록한다. 이를 위하여 이 논문에서는 모집단을 원환체(Torus)로 구성한다.

이 논문의 구성은 다음과 같다. 우선 2장에서는 여러 개의 해를 동시에 찾기 위한 기존의 연구를 살펴보고, 3장에서는 원환체형 모집단 유전자 알고리즘을 제안하고, 4장에서는 실험을 통하여 그것을 분석하고, 5장에서 결론을 맺는다.

2. 관련연구

여기서는 진화 알고리즘에 기반하여 여러 개의 최적해를 동시에 찾는 알고리즘에 대한 연구 사례를 소개한다. 가장 처음으로 이 문제의 해결을 시도한 것은 반복에 기반한 것이다[2]. 이 방법은 동일한 문제에 대하여 유전자 알고리즘을 독립적으로 여러번 적용하여, 여러 최적해를 찾는다는 매우 단순한 것이며, 다수의 최적해가 찾아질 확률이 거의 같다는 가정을 전제로 하지만, 실제로 그렇지 않기 때문에 효과적이지 못하다. 다른 방법으로는 골드버그Goldberg와 리처드슨Richardson이 제안한 것으로 공유에 기반한 방법이 있다[3]. 이 방법은 모집단 내의 각 개체에 대하여 그것들 사이의

(유전형 혹은 표현형 수준에서의) 유사성을 계산한 후, 유사한 집단에 속하는 것끼리 적합도를 감소시키도록 하여, 특정한 형태의 유전자형을 갖는 개체가 모집단의 대부분을 차지하는 것을 제한함으로써 모집단이 여러 개의 최적해로 수렴하도록 유도하는 방법이다. 결국 모집단은 유사한 정도에 의하여 (유전형 혹은 표현형적으로) 독자적인 부 모집단(subpopulation)을 구성하게 되며, 다양한 문제에 적용한 결과 상당히 효과적인 것으로 알려져 있다. 이 방법의 단점은 개체들 사이의 유사성을 평가하는 추가 시간이 필요하다는 것이다. 이것과 유사한 것으로, 일단 최적해가 찾아지면 그것을 배제하도록 적합도 함수를 수정하는 것이 비슬리(Beasley), 불(Bull), 그리고 마틴(Martin)에 의하여 제안되었다[2].

최근에 스피어스(Spears)에 의하여 제안된 것으로 공유와 제한된 짝짓기를 기반으로 하는 방법이 있다[4]. 이 방법 역시 골드버그 등의 방법과 유사하며, 여기서는 각 개체에 (임의로) 꼬리표를 붙여서 같은 꼬리표를 갖는 것들은 같은 부 모집단에 속하는 것으로 취급한다. 그리고, 적합도를 부 모집단 단위로 평가함으로써, 여러 개의 최적해를 동시에 찾을 수 있다. 제한된 짝짓기는 상이한 유전자를 갖는 최적 개체간의 교차를 금지시키는 것을 말한다. 일반적으로 최적 개체간의 교차의 결과가 나쁘기 때문이다.

기존의 연구들은 나름대로의 근거를 가지고 제안된 방법들이며 특히, 그것들은 공통적으로 적합도를 제어하여 특정 개체가 모집단 전체를 차지하는 것을 제한한다는 특징을 갖는다. 이 연구에서는 이러한 것들과는 다르게 모집단 내 개체들 사이에 논리적인 지역성을 부여하여, 모집단이 여러 개의 최적해를 포함하도록 하는 유전자 알고리즘을 제안한다.

3. 원환체형 모집단 유전자 알고리즘

어떤 문제에 유전자 알고리즘(GA)을 적용하기 위해서는 개체의 표현방법, 모집단 구성 방법과 적용할 진화 연산자 및 유전 연산자를 정하여야 한다. 여기서는 원환체형 모집단 유전자 알고리즘(TPGA로 표기)에서의 모집단과 진화 및 유전 연산자를 다룬다.

3.1 원환체형 모집단과 개체의 이웃

유전자 알고리즘에서 개체의 표현법이 정해지면 그러한 개체들의 모임인 모집단을 구성하게 된다. 단순 유전자 알고리즘에서 모집단은 일정한 개수(모집단의 크기)의 임의의 개체들의 모임이며, 모집단 내의 개체들은 단지 같은 세대 구성원이라는 것 이외의 어떠한 관계도 가지고 있지 않다.

TPGA에서는 모집단 내 개체들 사이에 이웃의 개념을 부여하기 위하여 임의로 생성된 개체들을 다음 그림 3.1 (a)와 같은 원환체 구조로 배열되도록 한다. 즉, 격자를 같은 도넛 모양의 도형에서 각 격자가 개체들의 서식지가 된다. 이러한 도형은 한 단면을 자르면 원통형이 되고, 그것을 다시 세로로 잘라 펼치면 그림 3.1 (b)와 같은 2차원 테이블 구조가 된다. 물론 모집단의 이러한 구성은 논리적인 것이다. 즉, 실제 구현은 어떠한 간에 개체들이 논리적으로 그렇게 구성되면 된다.

원환체형으로 구성된 모집단에서 개체의 이웃은 여러 가지 방법으로 정의될 수 있다. 그림 3.2는 가능한 3가지 경우를 나타내며, 그림 3.1의 (b)에는 그림 3.2의 (a)와 같이 인접한 8개의 이웃을 갖는 개체를 내부, 변, 모서리로 구분하여 나타낸 것이다.

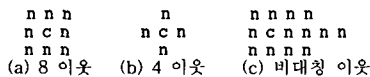


그림 3.2 개체와 이웃개체

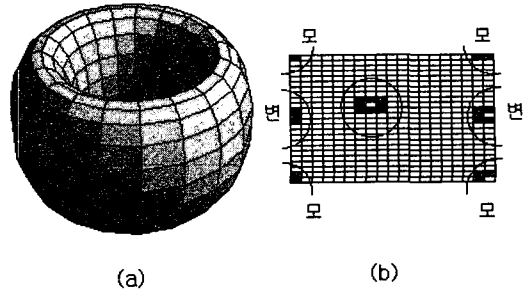


그림 3.1 원환체와 이웃

3.2 진화 및 유전 연산자

유전자 알고리즘을 적용할 때 사용하는 진화 연산자 즉, 선택 연산자 및 교차와 돌연변이로 대표되는 유전 연산자의 종류는 매우 다양하며, 각각의 방법은 나름대로의 특징을 갖는다 [5,6]. 이 논문에서 제안하는 원환체형 모집단 유전자 알고리즘(TPGA) 역시 다양한 진화 및 유전 연산자를 사용할 수 있다. 그러나, TPGA에서 중요한 것은 개체가 이웃의 개념을 가지며, 모집단은 내부적인 지역성을 가지고 있다는 것이다. 따라서, 진화 및 유전 연산자를 고안함에 있어서 지역성이 반드시 적용되도록 해야한다. 극단적으로, 모집단을 원환체로 구성하였다 하더라도, 이웃의 개념이 없이 보통의 연산자들을 사용한다면 보통의 단순 유전자 알고리즘과 다를 것이 없다.

이 논문에서는 가능한 여러 가지 방법 중에 콘웨이(Conway)의 "생명 게임"[7]에 사용된 규칙을 응용한 진화 및 유전 연산자를 제안한다. 제안하는 방법은 다음의 절차를 따른다.

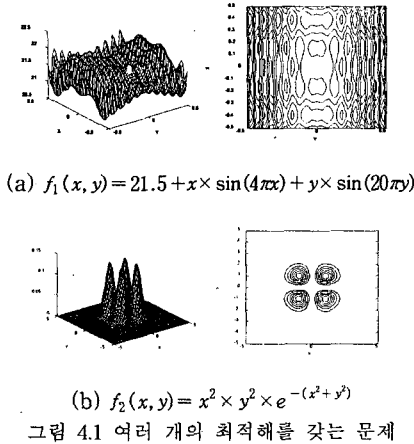
TPGA:

1. 문제 정의
 - 문제를 분석하여, 목적 함수 및 개체의 표현법을 정한다.
2. 모집단 구성 및 이웃의 유형 결정
 - 정해진 개수(모집단 크기)의 모집단을 원환체로 구성한다. 이때 사용할 이웃의 유형도 정한다. 모집단의 크기 IP 는 $n(n \geq 2, \text{개체의 중복 가능})$ 이 될 것이며, 일반성을 잃지 않는다.
3. 적합도 및 지역 적합도 계산
 - 3.1 각 개체의 적합도 fit 를 계산한다.
 - 3.2 각 개체에 대하여, 지역 적합도 lof 를 계산한다. lof 는 자신의 이웃 중에서 자신보다 적합도가 큰 이웃의 개수를 말한다.
4. 진화 및 유전 연산
 - 진화 및 유전 연산은 다음의 규칙을 따른다.
 - 1) lof 가 s 보다 작으면 개체는 변하지 않는다.
 - 2) lof 가 s 보다 작지 않으면 유전 연산을 적용한다.
5. 3번으로 이동 또는 종료

결국, TPGA에서는 각 개체가 (논리적 의미에서)지역적으로 인접한 개체와 경쟁하여 자신의 생,사가 결정되며 따라서, 특정 개체가 모집단 전체로 퍼지는 것을 제한하게 된다. 반면에 전체 모집단 수준에서는 지역의 경계가 확정적인 것은 아니므로(원환체형 모집단은 사방으로 끝없이 연결되어 있다) 자연스럽게 지역간의 유전자 교환이 가능하다.

4. 실험 및 분석

제한한 알고리즘이 여러 개의 해를 동시에 찾는다는 것을 확인하기 위하여, 이 논문에서는 다음 그림 4.1과 같은 두 함수에 대하여 실험을 수행하였다.



함수 (a)는 대단히 많은 최적해를 갖는 경우이고 (b)는 4개의 점 (-1, -1), (-1, 1), (1, -1), (1, 1)에서 최적해를 갖는다. 비교의 대상은 여러 개의 최적해를 동시에 찾는 다른 방법들이 아니며, 이 논문에서는 단순 유전자 알고리즘과 비교하였음을 밝힌다. 여러 개의 최적해를 찾는다는 측면보다는, 그것은 부수적 결과이고, 개체가 이웃을 갖는다는 점이 더욱 중요하다고 본다.

공통적으로 모집단의 크기는 16이며, 세대수는 1000으로 하였고, 유전자 알고리즘의 경우 매개변수는 $P_c=0.75$, $P_m=0.01$ 로 하였다. TPGA의 경우 이웃은 8이웃을 사용하였으며, 3절의 TPGA 적용과정 중 단계 4는 구체적으로 다음을 적용하였다. 돌연변이 확률은 0.5로 하였다.

- 1) lof가 s(=3)보다 작으면 개체는 변하지 않는다.
- 2) lof가 d(=6)보다 크면 현재 최적해로 대체 한다.
- 1), 2) 이 외의 경우는 돌연변이 연산 적용

실험 결과는 그림 4.2와 같다. 해를 찾는다든 측면에서 두 함수 모두 TPGA가 더 좋은 해를 찾았다(GA 최적해는 각각 30.12, 12.05, TPGA 최적해는 각각 30.89, 13.53). 다음 표 4.1은 f_2 함수에 대한 1000번째 세대 모집단 개체의 표현형(x,y)을 나타낸다. 실제로 두 함수 모두 유전자 알고리즘의 경우 개체가 특정한 값으로 수렴되나, TPGA의 경우 다양한 값들을 가지며, 특히 4개의 최적해를 갖는 함수 f_2 의 경우, 총 16개 개체 중 8개가 각각 두 개씩 4개의 최적해로 수렴함을 알 수 있다.

표 4.1 개체의 분포

1.33:-1.05	1.33:-1.05	1.34:-0.97	1.35:-0.97
1.37:-0.97	1.37:-0.97	1.35:-0.97	1.35:-0.97
1.33:-1.01	1.33:-1.01	1.35:-0.97	1.35:-0.97
1.35:-0.97	1.35:-0.97	1.35:-1.13	1.35:-1.13

(a) 1000세대 후 모집단의 개체들(GA)

0.47:4.90	-3.66:-2.30	-2.46:0.85	-0.97:1.11
-0.94:0.98	-0.96:-0.93	0.99:-0.90	-1.66:3.51
0.95:0.98	0.05:1.48	-1.05:-1.07	2.86:-1.06
-4.46:0.48	1.01:-1.01	1.57:1.27	1.01:0.94

(b) 1000세대 후 모집단의 개체들(TPGA)

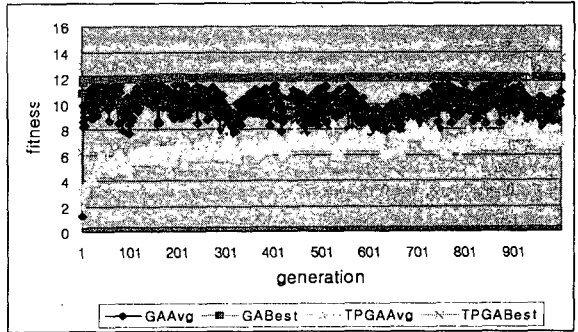
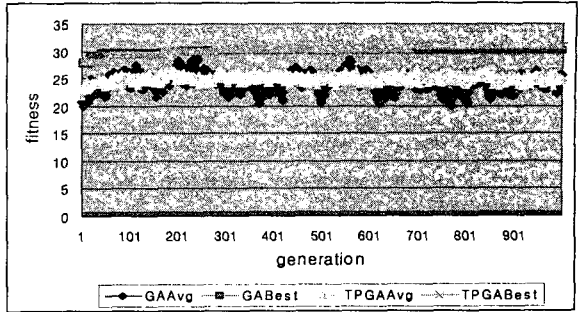


그림 4.2 실험 결과

5. 결론

이 논문에서는 모집단을 원환체(Torus)로 구성하고 개체가 이웃의 개념을 부여하여 모집단이 최적해 집단으로 수렴하는 유전자 알고리즘을 제안하고 평가하였다. 실험에 사용한 함수에 대하여 기대했던 결과를 얻을 수 있었으며, 제안한 방법은 개체사이에 이웃이라는 개념을 부여함으로써 다수의 해를 동시에 찾는다는 생각을 넘어서 다양한 변형 유전자 알고리즘에 대한 새로운 모델이 될 것으로 기대된다.

6. 참고 문헌

- [1]. Z. Michalewicz, "Genetic Algorithms+Data Structures = Evolutionary Programs", pp. 169-172. 1999.
- [2]. Beasley, et. al, A sequential Niche Technique for Multimodal Function Optimization, Evolutionary Computation, Vol.1, No.2, pp. 101-125, 1993.
- [3]. Goldberg, Richardson, Genetic Algorithms with Sharing for Multimodal Function Optimization, Proc. 2nd ICGA, pp. 41-49, 1987.
- [4]. Spears, Simple Subpopulation Schemes, Proc. 3rd ACEP, pp. 296-307. 1994.
- [5]. Hancock, Selection Methods for Evolutionary Algorithms, in Practical Handbook of Genetic Algorithms, Vol. 2, pp. 67-92, 1995.
- [6]. Pawlowsky, Crossover Operator, in Practical Handbook of Genetic Algorithms, Vol. 1, pp. 101-114, 1995.
- [7]. Poundstone, The Recursive Universe, New York: Morrow, 1985.