

유전자 알고리즘에서 선택 기법을 이용한 해의 수렴 과정에 관한 연구

-A Study on the Convergence of Optimal Value using Selection Method in Genetic Algorithms-

김 용범*

Kim, Yong-Beom

김 병재**

Kim, Byeong-Jea

박 명규**

Park, Myeong-Gyu

Abstract

Genetic Algorithms face an inherent conflict between exploitation and exploration. Exploitation refers to taking advantage of information already obtained in the search. Exploration show that a pattern in bits coupled with another pattern elsewhere in the string is more effective. In this paper shows that the selection method has a major impact on the balance between exploitation and exploration. A more heavy-handed approach seeks to exploit the available information. If decisions must be made quickly ,especially those in real-time trading environments, then quicker convergence through exploitation may be more desirable. Also this paper we present some theoretical and empirical the selection method in genetic algorithms for a GA-hard problem.

1. 서론

유전자 알고리즘은 여러개의 개체(individual)가 동시에 병렬적으로 주어진 환경에 따라서 최적의 상태로 향하며, 적자 생존(fittest survival)의 방법으로 진화하여 궁극적으로는 최적의 상태에 도달하는 생태계의 진화이론에서 따온 알고리즘으로 HOLLAND에 의해 처음으로 도입되었다.

유전자 알고리즘에서 여러개의 개체들로 구성된 세대(generation)가 진화(evolution)할 때 이전의 세대가 터득한 환경에 대한 정보는 염색체(chromosome)의 형태로 표현되어 다음 세대에 전달된다. 이때 조상(parents)의 염색체가 그대로 복제(reproduction)되어 자손(child)에 전달되는 것이 아니라 조상의 염색체에 선택(selection), 교차(crossover), 돌연변이(mutation)연산을 이용한 형태의 염색체로 전달된다. 구 세대중에서 한 개체가 선택되어 자손에 유전 정보를 남길 확률은 일반적으로 그 개체의 상호 작용에 의존하는 값에 따라 변한다. 유전자 알고리즘은 여러 가지 종류의 최적화 문제에 응용되어 좋은 결과를 보여주고 있고, 복잡한 공학적인 최적

* 명지대학교 산업기술연구소 책임연구원

** 명지대학교 산업공학과

화 문제에서도 많이 이용되고 있다. 특히, 이 알고리즘은 근본적으로 각 개체의 병렬적인 진화에 기본은 둔 것으로 최근의 병렬 계산 컴퓨터에 적합한 것으로 이런 점에서 직렬 계산에 기반을 둔 담금질 방법보다 더 빠른 결과를 요한 문제에 적합하다. 그러나 유전자 알고리즘을 이용한다고 해서 전체최적해(global optimum)를 구할 수 있는 것은 아니다. 이것은 평가함수(fitness population)에 따라 다를 수도 있으며 생성된 초기개체군(initial population)이나 돌연변이 비율, 교차비율에 따라 다를 수도 있다. 또한 진화의 세대를 얼마로 하느냐에 따라 그 결과가 달라진다. 그러므로 유전자 알고리즘 연구에서는 이와 같은 매개변수(parameter)들이 주어진 환경에 항상 최적의 된다는 것을 보장할 수 없기 때문이고, 매개변수들이 어떻게 설정되었느냐에 따라 CPU time과 저장공간에 많은 문제점이 나타나게 된다. 이와 같은 문제점을 해결하여 보다 개선된 해와 저장공간을 줄이기 위한 방법으로 매개변수 및 선택연산자의 적절한 조합 및 알고리즘의 종결조건등과 같은 방법이 있다. 그중에서도 적합한(adaptive) 선택연산자의 선정은 해의 질(전체 최적해에 대한 비율)을 개선시키고 CPU time을 줄이는데 결정적 역할을 한다. 유전자 알고리즘은 exploitation과 exploration간의 대립에 의해 수행된다. 여기서 exploitation은 해의 탐색과정에서 이미 얻어진 정보를 적절하게 활용한다는 장점이 있다. 예를 들면, 개체군에서 유전인자 1010을 갖는 string들이 보다 개선된 적합도 값을 갖는 string들이라고 가정하자. 만약 세대가 진행되는 과정에서 1010의 형태를 갖는 이러한 유전인자들의 위치들이 빠르게 고정(fixing)된다면, exploitation은 개체군의 평균 적합도를 개선하는데 이러한 정보를 이용하게 된다. 그러나 개체군이 너무나 빠르게 이러한 유전인자들로 구성된다면 사전수렴(premature)이라는 위험이 따르게 된다. 여기서 이와같은 사전수렴을 방지해주는 것이 즉, string에서 유전인자들의 또다른 형태가 더 효과적일 수 있다는 것을 보여주는 것이 exploration이다. 유전자 알고리즘에서는 exploitation과 exploration간의 이와 같은 경향이 항상 존재하여 세대가 진행되면서 더욱 개선된 해를 탐색하는 기법이다. 이와같은 exploitation과 exploration간의 적절한 균형을 맞추기 위한 결정적 역할을 하는 것이 선택기법이다.

따라서 본 연구에서는 해의 탐색과정에 있어서 결정적 역할을 하는 선택기법을 이용하여 평가함수를 개선시키고, 계산시간을 단축 할 수 있게 각 선택 기법을 비교 검토하여 알고리즘 수행시 발생하는 문제점을 해결하고자 한다.

2. 복제계획

복제계획이라 불리는 적응적인 시스템은 Holland[4]에 의해 처음으로 제시되었다. 기본적으로 적응적인 문제의 여러 해들은 개체군을 이루는 집합 A 에 의해 표현된다고 하자. 복제계획에서, N 번째 세대에서의 시스템의 수행도(performance)는 A 로부터 i 개의 개체들인 s_{iN} 로 구성된 개체군 $A(N)$ 과 관련된 수행비율 $F(s_{iN})$ 로 구성된다. 적응적인 문제에서 해의 표현방식인 s_{iN} 는 복제계획에 의해 진행되는 유전자 형질을 고려한 것이다. 그러므로 새로운 해가 되는 개체들은 유전자 개체군이 적용된 모의실험에 의해 생성된다. 즉, 개체군 $A(N)$ 으로부터 생성된 개체들은 조상들의 개체로부터 선택(select)되고, 유전 연산자에 의해 새로운 자손이 생성된다.

이와 같은 복제계획은 다음과 같은 적용단계를 거친다.

적용단계

- 단계 1. [초기 개체군 $A(0)$ 를 랜덤하게 생성한다.]
- 단계 2. [$A(N)$ 에서 존재하는 각 s_{iN} 를 계산한다.]
- 단계 3. [선택확률 $P(S_{iN})$ 을 다음과 같이 계산한다.]

$$P(S_{iN}) = \frac{F(S_{iN})}{\sum_{i=1}^n F(S_{iN})}$$

단계 4. [선택확률 분포에 의해 $A(N)$ 로부터 선택된 개체들에 의해 $A(N + 1)$ 을 생성하고 유전 연산자를 적용한다. 단계 2로 간다.]

단계 3에서 적용된 선택기법은 여러 기법들 중에서 일반적으로 유전자 알고리즘에 적용된 기법인 roulette wheel에 의한 기법이다. 이 선택기법의 효과는 적합도에 의해 선택된 조상을 재 배치하는 것이다. 비록 이와 같은 선택과정이 랜덤하다 할지라도, 각 조상이 선택될 기회는 적합도에 대한 직접적인 비율에 의거한다.

이 알고리즘에서 세대가 진행되는 동안 적합도가 작은 개체들은 소멸되고, 적합도가 큰 개체들은 개체군 전체에 넓게 확산되어 균형을 이루고 있다. 한편, 최악(worst)의 개체가 때때로 이 알고리즘의 적용시 선택될 수도 있다. 이와 같은 경우, 이러한 선택기법이 적용된 유전자 알고리즘의 수행도는 좋지 못하므로, 평균 이하의 적합도를 갖는 개체는 무시해도 좋다. 이 기법의 적용시 개체들의 평가함수는 유전자 알고리즘이 대상문제의 최대값을 찾는 탐색기법이므로 반드시 양수가 되어야 한다.

복제계획에서 조상의 개체에 의해 생성되는 자손의 기대수는 적합도에 대한 비율로 주어진다. 그러므로, 다음 세대를 구성하는 개체 s_{iN} 의 기대수는 다음과 같다.

$$\begin{aligned} E[S_{iN}] &= n \cdot P(S_{iN}) \\ &= n \cdot \frac{F(S_{iN})}{\sum_i F(S_{iN})} \\ &= \frac{F(S_{iN})}{\frac{1}{n} \sum_i F(S_{iN})} \\ &= \frac{F(S_{iN})}{F(A(N))} \end{aligned}$$

평균 적합도를 갖는 개체들은 평균적으로 하나의 자손을 생성하게 되고, 반면에 평균 이상의 개체들은 하나 이상의 자손들이 생성되며, 평균 이하의 개체들은 하나 이하의 자손을 생성한다. 그러므로 주어진 환경에 적응되기 위해 다른 선택과정을 갖지 않는 상태에서 적합도에 의한 복제비율은 개체군의 보다 큰 비율을 취하게 되는 $A(0)$ 에서의 최상의 개체에 의해 세대 $A(N)$ 의 결과를 생성한다. 그렇지만 자연과 이와 같은 인공의 시스템에서 자손은 그 조상에 대해 정확하게 두 배로는 복제되지 않는다. 높은 적합도의 기대치를 갖는 생성된 개체들에 의해 이와 같은 선택과정을 이용하는 것이 유전 연산자의 역할이다.

3. 선택과정

대상문제의 개체군을 갖고 실행되는 GAs은 세대가 진행되는 동안 점차적으로 개선된 해를 갖는다. 가장 적합하게 생존한다는 것은 개체군에서 최상으로 수행되는 개체만이 결국 생존한다는 것이다. GAs의 주된 구성은 한 세대로 부터 다음 세대까지 진행하는데 이용되는 선택과정이다. 여기에는 잠재적인(potential) 조상(parents)들을 어떻게 선정하고 그들을 이용하여 어떻게 새로운 자손(offspring)을 생성시키는지에 관련된 많은 가능한 변화가 나타난다. 본 절에서는 GAs에서 새로운 자손을 생성시키는데 결정적 역할을 하는 선택기법들에 대해 언급하고자 한다.

3.1. Roulette Wheel selection

이 과정은 적합도에 근거한 가중치(weighted)를 이용한 선택과정을 이용한다. 예를들어, 10개의 개체(individual)를 갖는 하나의 개체군을 가정하자. 여기에는 A라는 형태에 6개의 유사한 개체들이 포함되고, B에는 4개의 유사한 개체들을 포함된다고 하자. 형태가 A인 개체들은 12인 fitness값을 갖고 있고, B인 개체들은 7인 fitness값을 갖는다고 가정하자. 이러한 개체군의 총 fitness값은 $6 \times 12 + 4 \times 7 = 100$ 이다. 따라서 A는 총 fitness값의 72퍼센트를 차지하고, B는 28퍼센트를 차지한다. 이와같이 비율에 의해 복제(reproduction)가 이루어지는 방법이 weighted roulette wheel기법이다. 이러한 접근방식을 가지고, 수준 A에는 72퍼센트의 wheel로 구성되고 B는 28퍼센트의 wheel로 구성된다. 다음 세대를 선정하기 위해, 단지 wheel을 10번 회전시키고 A와 B형태의 새로운 개체들을 혼합하여 창출한다. 새로운 세대를 창출 할 때, 평균이상의 개체들이 형태 A에는 약 7개의 개체가, B에는 2개의 개체가 포함된다. wheel의 회전은 기회(using a random number generator in the computer implementation)에 의해 통제되어지기 때문에, 실제 A/B의 분열은 0|10부터 10|0까지 어느것도 될 수 있다. 그렇지만, 평균치 이상인 보다 큰 fitness값을 갖는 개체는 다음 세대를 구성하는 개체가 될 확률이 그만큼 높아진다. 이와 같은 roulette wheel 기법은 다음과 같은 적용단계를 거친다.

적용단계

- 단계 1. [모든 개체들의 적합도를 더한다.]
- 단계 2. [각 개체들의 선택확률과 누적 선택확률을 계산한다.]
- 단계 3. [0과 1 사이에서 개체군의 크기(n)만큼의 난수를 발생한다.]
- 단계 4. [발생된 난수보다 크거나 같은 누적 선택확률 값을 갖는 개체가 다음 세대를 구성하는 개체군의 개체가 된다.]

예를 들어, 다음과 같은 10개의 개체들로 구성된 개체군을 가정하자.

개체(individual)	1	2	3	4	5	6	7	8	9	10
적합도(fitness)	8	2	17	7	2	12	11	7	3	7
선택확률	0.1053		0.2237		0.0263		0.1447		0.0395	
		0.0263		0.0921		0.1579		0.0921		0.0921
누적선택확률	0.1053		0.3553		0.4737		0.7763		0.9079	
		0.1316		0.4474		0.6316		0.8684		1

여기서 0과 1사이에서 10개의 난수를 발생하면, roulette wheel기법에 의해 선택되어 다음 세대를 구성하는 개체군은 다음과 같다.

난수	0.267640	0.596379	0.487218	0.513040	0.192649					
		0.739615	0.906407	0.344526	0.168955	0.565321				
선정된 개체	3	7	6	9	7	3	6	3	3	6

Roulette Wheel기법은 확률적인 과정을 따르는 기법으로, 상대적으로 큰 적합도를 갖는 개체는 다음 세대를 구성하는 개체로 복제될 가능성이 높고, 반면에 적은 적합도를 갖는 개체는 그 가능성이 그만큼 작아진다. 따라서 이 표에서 적합도가 큰 개체인 3, 6, 7의 개체는 다음 세대에 2배 이상의 비율로 복제된다는 것을 알 수 있다.

3.2. Stochastic remainder selection.

위에서 언급된 10개의 개체들인 A-B로 구성된 개체군을 가정하자. 다음 세대에 A라는 형태의 개체가 복제될 기대 수는 7.2개이고, 반면에 B라는 형태의 개체가 복제될 기대 수는 2.8개이다. 본 기법은 첫번째, 기대치의 정수부분인 7과 2를 다음 세대를 구성하는 개체들로 할당시킨다. 즉 A성분을 7개 B성분을 2개 복제하여 다음 세대를 구성한다. 다음으로 10번째 개체를 결정하기 위하여 소수점이하부분을 weight a roulette wheel방법을 이용한다. 따라서 본 기법은 A와 B개체들간에 7:3 또는 8:2비율로 다음 세대를 발생시키게 된다. 본 기법은 fitness 비율에 밀접하게 관련된 것으로, 다음 세대를 결정하는데 더욱 강압적이라는 것이 주목할 만하다.

3.3. Stochastic universal selection

이 기법은 앞의 두기법과는 다른 variation이 있는 것으로, 단지 그것을 마음속으로 어느정도 형상화시키는 것이다. 이와같이 형상화 시킨 그림은 wheel에 A가 표시되어 있는 부분이 72%, B가 표시된 부분이 28%를 차지하는 roulette wheel이다. 10개의 개체들이 동등하게 표시된 영역의 바깥쪽 테두리를 상상한다. roulette wheel은 1번의 회전이 주어진다. 다음 세대를 구성하는 개체들을 결정하기 위하여, 표시된 10개의 개체들을 조사한다. 즉 A구역의 반대편에는 A형태의 개체를 할당하고, B구역의 반대편에는 B형태의 개체를 할당한다. 본 기법은 stochastic remainder selection과 유사한 결과가 나타난다. 즉 본 기법의 경우는 7:3또는 8:2의 개체들로 구성된다.

3.4. Genitor selection

본 기법은 1989년에 Whitley에 의해 개발되어진 variation으로, 단순한 ranking에 근거한 선택(selection)단계를 이용한다. 첫번째 단계는 전단계에서 계산된 적합도(fitness)에 의한 string의 ranking을 생성한다. 다음으로 개체군에서 최상의 string을 복제(copy)하여 최악(worst)의 string과 대체한다. 랜덤한 초기 개체군에서 최적해에 관련된 string이 생성된다고 가정하자. 또한 이러한 최적 string의 이름을 OPTIE라 하자. GAs의 일련의 단계를 살펴보면, OPTIE는 교차와 돌연변이연산을 통해 변형될 수 있다. 그러나 만약 OPTIE가 변형되지 않고 계속 생존하게 된다면, 각 세대에서 최악의 string과 계속 대체된다. 세대마다 하나의 string만이 대체되므로, 개체군이 수렴하기 위해서는 최소한 개체군의 크기만큼의 세대가 취해져야 한다. 그렇지만, 더욱 바람직한 수렴을 위해서는 개체군의 크기보다 훨씬 큰 세대 수를 탐색하여야 한다. 만약 선택이 강압적인(heavy-handed)형태로 행해졌다면, 개체군은 빠르게 수렴하게 된다. 예를들어, 최상의 string을 5개 복제하여 5개의 최악의 string과 대체하면, 빠른 수렴은 가능하지만 양질의 해는 보장하지 못한다. 이와같은 경우를 premature convergence라 한다. 즉, 조기수렴은 탐색영역을 조사하는데 충분한 시간을 가지면서 탐색하는 것이 아니라, 단지 강한 개체쪽만을 탐색해 갈 뿐이다.

3.5. Ranking selection with roulette wheel

본 과정은 fitness에 의해 개체군을 rank-ordering하는 것에 의해 시작한다. 다음으로, 하나의 assignment function에 의해 다음 세대를 위한 각 개체의 확률이 주어진다. 보다 큰 순위의 개체는 다음 세대를 구성할 때 보다 큰 확률을 갖는다. roulette wheel은 assignment function에 의해 결정되어진 우선순위를 가지고 이용한다. n개의 크기를 갖는 개체군의 다음 세대는 주어진 wheel을 n번 회전시켜 만든다. 이 과정은 개체군에서 더욱 우수하게 수행되는 개체쪽으로 selection을 하는 것이나 다음 세대를 구성하는데 있어 어떤 특정한 개체들을 강요

하는 것은 아니다.

3.6. Tournament selection

tournament는 다음 세대에 참여될 수 있는 개체들간의 작은 jousting시합과 같다. 이 과정은 동시에 두개 또는 그 이상의 개체들을 포함한다. 예를들어, 두 개체들이 개체군으로부터 랜덤하게 선택되면, 그들의 fitness값을 비교한다. 두 개체중에서 수행이 더 좋은 개체, 즉 fitness값이 큰 개체는 다음 세대를 구성하는 개체로 자동적으로 선택되어 지거나, 가중치가 부여된 동전을 던져 두 개체들간에 한 개체가 선택될 수 있게 모의실험한다. 예를들어, 수행이 더 좋은 개체가 선택될 확률이 75%로 결정되었다면, 앞면이 나올 확률이 0.75인 가중치가 부여된 동전을 던진다. 만약 동전이 앞면이 나왔다면, 수행이 더 좋은 개체를 선택하고, 그렇지 않으면 그 나머지를 선택한다. tournaments는 개체군의 다음 세대가 완전하게 구성될 때까지 계속적으로 수행한다.

4. 각 선택기법의 적용에 대한 모의실험

본 연구의 모의실험에는 IBM PC 586, Pentium 133MHz, 32MB Memory의 환경에서 실행하였으며, 사용된 프로그램은 Mathematica Ver 2.2.3으로 작성하여 20번의 실험수행 결과 19번이 전체 최적해에 도달 되었을 때 그 평균치를 설정된 매개변수(개체군의 크기 $N=100$, 교차확률 $P_c=0.60$, 돌연변이확률 $P_m=0.001$)를 아래와 같이 주어진 문제[참고:그림 1]와 제안된 선택기법에 적용하여 최적해를 구하기 위한 모의실험의 수행 및 계산시간의 감소에 따른 문제점을 파악하고자 한다.

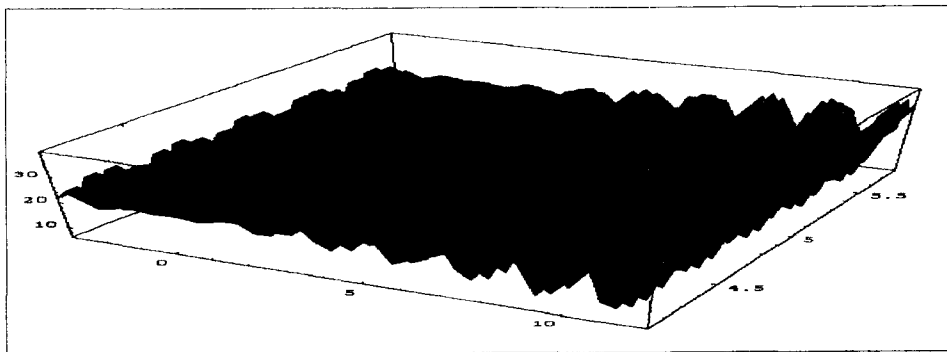
$$f(x, y) = 21.5 + x \sin(4\pi x) + y \sin(20\pi y) \\ - 3.0 \leq x \leq 12.1 \\ 4.1 \leq y \leq 5.8$$

또한 해의 종결조건으로는 bias[7]를 대상문제에 적용시켜 세대수를 결정하였고, 보다 정확한 전체 최적해를 구하기 위해 각 선택기법들의 수행결과를 [그림 2]에서 [그림 7]에 도시하였고, [표 1]은 기존의 유전자알고리즘 수행시 선택기법으로 사용되는 Roulette wheel기법과 비교한 해의 질과 계산시간의 결과를 제시한 것이다. 여기서 해의 질은 전체최적해의 값($f^* = 39.2$)과 비교한 결과이다.

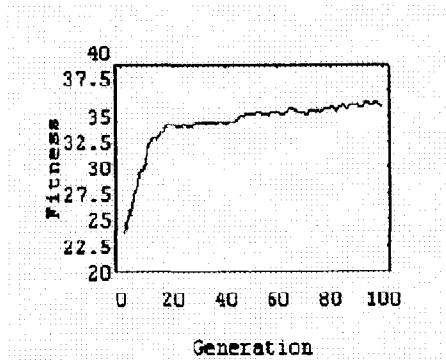
[그림 2]에서 [그림 4]에 도시된 선택기법인 Roulette Wheel, Stochastic remainder, Stochastic universal 등은 확률적인 과정을 따르는 기법으로, 상대적으로 큰 적합도를 갖는 개체는 다음 세대를 구성하는 개체로 복제될 가능성이 높고, 반면에 적은 적합도를 갖는 개체는 그 가능성이 그 만큼 적어지는 선택기법들이다. 이와같이 확률적 선택방식이 적용된 모델에서는 가장 우성인 염색체와 가장 열성인 염색체가 소멸되거나 생존될 수도 있는 한계로 인해 해의 개선에 도움을 주지 못하는 경우가 발생된다. 그러므로 가장 우성인 염색체들을 세대가 진행되는 동안에 계속 생존하게 하여 해의 개선에 도움을 줄 수 있는 선택기법의 필요성이 제기된다.

[그림 5]에서 [그림 7]에 제시된 Genitor, Ranking, Tournament 등과 같은 기법은 선택이 강압적인 형태로 행하여지는 것으로 가장 우성인 개체는 세대가 진행되는 동안 가장 열성인 개체들과 대체되는 기법으로 해의 개선율이 빠르게 진행된다는 것을 알 수 있다. 그러나 이와같은 기법들은 해의 개선율은 좋으나 개체군이 작을 경우 세대가 진행되면서 염색체들간에 동질(homogeneous)이 되려는 경향으로 인해 사전수렴의 원인이 된다. 또한 강압적인 선택기법은 각 세대마다 가장 우성인 염색체들을 메모리상에 저장하고 있어야 하므로 확률적 선택기법에 비해 계산시간은 3%정도 증가하나, 해의 질은 5%까지 크게 개선시키는 효과가 있다.

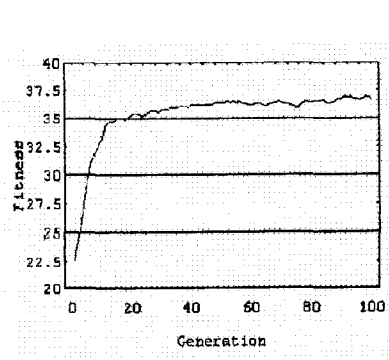
따라서 개체군의 크기가 큰 경우 우성인 개체들을 계속 발생시키는 선택연산자는 우성염색체를 메모리상에 저장해야 하므로 계산의 복잡도 면에서 매우 불리하다. 그러므로 개체군의 크기가 큰 경우에는 랜덤한 선택기법인 Roulette wheel을, 반면에 개체군의 크기가 작은 경우에는 해의 개선폭이 높은 Ranking에 관련된 선택기법을 이용하는 것이 GAs 수행시 계산시간을 줄이고 해의 질을 개선시키는데 결정적 역할을 하게된다.



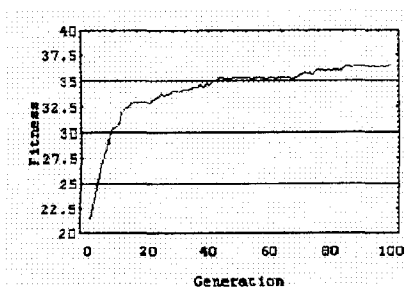
[그림 1] $f(x, y) = 21.5 + x \sin(4\pi x) + y \sin(20\pi y)$ 의 그래프



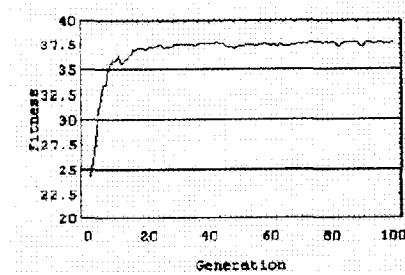
[그림 2] Roulette Wheel selection



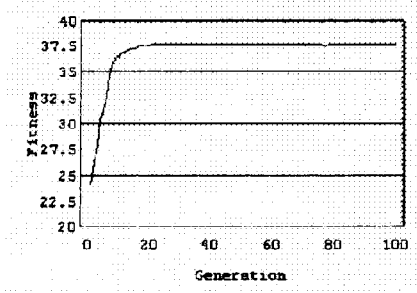
[그림 3] Stochastic remainder selection



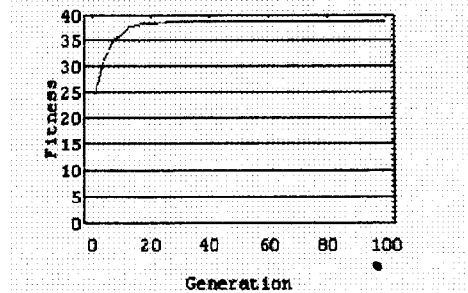
[그림 4] Stochastic universal selection



[그림 5] Genitor selection



[그림 6] Ranking selection



[그림 7] Tournament selection

[표 1] 각 선택기법간의 결과 비교

종 류	계산시간(초)	해의 질(%)
Roulette Wheel selection	78.3	92.5
Stochastic remainder selection	78.7	94.3
Stochastic universal selection	77.6	93.1
Genitor selection	82.3	96.4
Ranking selection	81.5	95.8
Tournament selection	82.4	98.9

5. 결론

본 연구에서는 유전자알고리즘에 의해 대상문제를 해결하는데 있어 계산시간을 단축시키고 해의 질을 개선시키는데 결정적 역할을 하는 선택기법들을 모의실험을 통해 비교 연구하였다.

확률적 선택방식이 적용된 모델에서는 가장 우성인 개체와 열성인 개체의 소멸과 생존이 해의 개선에 도움을 주지 못하는 주요 원인이 되고 있는데, 두 개체를 선택하여 두 개체중 우성인 개체를 계속 생존시키는 Tournament 모델은 해의 질을 개선시키는 매우 효과적인 방법임을 모의실험을 통해 확인하였다. 또한 실험결과 강압적인 선택기법은 확률적인 선택기법에 비해 계산시간은 3% 정도 증가하나, 해의 질은 5% 까지 크게 개선시키는 효과가 있다.

본 연구에서는 선택기법에 대한 연구를 한 기법씩 개별적으로 실험을 수행하였다. 그 결과 해의 개선에 직접적인 영향을 주는 더욱 다양한 복제과정 및 그것을 조합한 복제과정의 연구가 필요하다고 본다.

참고문헌

1. Bäck, T., Hoffmeister, F. (1991), "Extended Selection Mechanisms in Genetic Algorithms", In Belew, R., Booker, L., pp 92 - 99.

2. Davidor, Y., Ben-Kiki, O. (1992), "The Interplay Among the Genetic Algorithm Operators: Information Theory Tools Used in a Holistic Way", In Reinhard, M. and Bernard, M., pp 75 - 84.
3. De Jong K. A. (1975), "An Analysis of the Behaviour of a Class of Genetic Adaptive Systems", Ph.D. thesis, University of Michigan.
4. Holland, J. H. (1975), *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor.
5. Michalewicz, Z. (1992), *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag.
6. Reinhard, M., Bernard, M. (1992), *Parallel Problem Solving from Nature*, Proceedings of the Second Conference on Parallel Problem Solving from Nature, North-Holland.
7. 김 용범 (1995), 유전자알고리즘의 수렴현상에 관한 연구, 박사학위논문