

論文99-36C-1-8

# 개체군 변환 유전자 알고리즘의 새로운 수명 할당 방식에 관한 연구

## (A Study on a New Lifetime Allocation Method of Genetic Algorithm with Varying Population Size)

————— 權 奇 浩 \* —————

(Key Ho Kwon)

### 요 약

본 논문은 개체군 변환 알고리즘의 새로운 수명할당방식(allocation method)을 제안한다. 이 방법으로 개체군의 크기를 적합도(fitness)에 따라서 조절할 수 있다. 개체군(population)의 크기는 최적해(optimal value) 근처로 갈수록 안정된 상태로 가게 된다. 유전자 코딩에 있어서는 이배체(diploidy) 방식을 사용하였다. 시뮬레이션을 통하여 새로운 수명 할당 방식이 개체군의 크기를 조절할 수 있음을 확인한다.

### Abstract

In this paper, we suggest a new lifetime allocation method of genetic algorithm with varying population size. This method can control the size of the population according to the fitness values. The population size is stabilized near the neighbourhood of the optimal value. We used the diploidy method in the coding of the chromosomes. Several simulations confirm that the new allocation method can control the size of the population.

### I. 서 론

현재까지 적절한 해법이 알려지지 않은 흥미로운 문제들이 많이 있다. 이러한 문제들 중에는 응용에서도 자주 나타나는 최적화문제(optimization problem)들이 많다. 어려운 최적화 문제에 대해서는 확률적인 알고리즘을 사용하고 있고 그 중에서도 유전자 알고리즘은 우수한 성능을 보이고 있다.<sup>[1]</sup> John Holland의 1975년도 저서인 『Adaptation in Natural and Artificial Systems』에 처음으로 소개된 유전자 알고리즘(Genetic Algorithms)은 생물 진화(선택 도태나 돌연변이)의 원리로부터 착안되었다. 이 알고리즘은 여러 개체를 유전자(gene)와 비슷한 구조로 부호화(coding) 한 후, 유전 연산자(genetic operator)를 이

용하여 병렬적으로 진화시켜, 최적의 상태에 이르게 만드는 확률적 탐색 기법이라고 말할 수 있다.

유전자 알고리즘은 잡 샵 스케줄링(Job Shop Scheduling)이나 순회 판매원 문제, 통신 네트워크의 설계, 철도회사의 배송 계획 최적화, 전투기 보수 계획의 최적화 등의 최적화 문제와, 자발적 발생이라고 하는 개념을 사용해 예술이나 엔터테인먼트(entertainment), 인공생명(artificial life) 등에 응용되고 있다.<sup>[2] [3] [4]</sup>

유전자 알고리즘은 전역 탐색 능력은 우수하나 지역 탐색 능력이 떨어지고, 세대가 지남에 따라 유전적 다양성(genetic diversity)을 상실하여 불완전 수렴(premature convergence)에 이른다는 단점이 있다.<sup>[5]</sup> 이러한 단점을 극복하기 위해 초기 개체군의 크기를 증가시켜 주거나 돌연변이 확률을 높이는 방법이 사용되었고 최근에는 군집(crowding), 사전선택(preselection) 등의 방법이 개발되었다.<sup>[6]</sup>

\* 正會員, 成均館 大學校 電氣 電子 및 컴퓨터 工學部  
(Sungkyunkwan University)

接受日字: 1998年5月18日, 수정완료일: 1998年12月23日

또한 각 세대마다 개체군의 크기를 유동적으로 적용하여 유전적 다양성을 향상시키고자 하는 개체군 변환 유전자 알고리즘(GAVaPS)도 제시되었다.<sup>[7]</sup> 그러나 이 방법은 개체군의 크기가 변화할 때 크기가 계속 증가하기 때문에 성능 향상에 별다른 기여를 하지 못하고 있다.

본 논문에서는 개체군 변환 유전자 알고리즘(GAVaPS)에 유전적 다양성을 더욱 향상시킨 이배체<sup>[8]</sup> 부호화를 도입하되, 기존의 GAVaPS의 수명 할당 방식과는 다른 새로운 수명 할당 방식을 도입하여 개체군의 크기를 조절할 수 있도록 하여 유전자 알고리즘의 성능을 향상시키고자 하였다.

## II. 일반적인 유전자 알고리즘

그림 1은 일반적인 유전자 알고리즘에 대한 블록도(block diagram)이다. 유전자 알고리즘은 적용하고자 하는 문제의 매개변수를 부호화 한 염색체(chromosome)라는 것을 이용하는데 이 염색체는 하나의 개체를 표시하며 데이터 영역이나 배열로 생각할 수 있다.

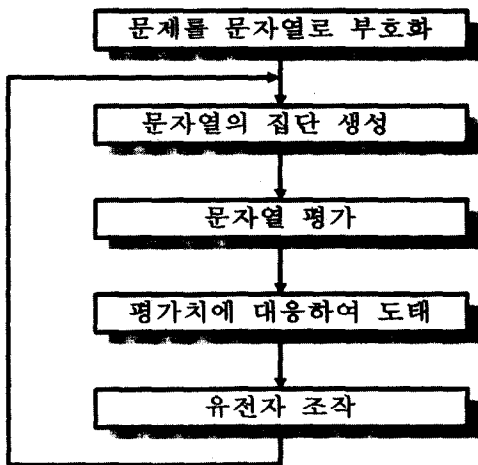


그림 1. 유전자 알고리즘의 블록도  
Fig. 1. The Block Diagram of GA.

염색체는 대부분 일차원의 배열로서 표현된다. 유전자형은 배열상에 표현된 값의 패턴이다. 개체 하나 하나는 해가 될 가능성이 있는 잠재해(potential solution)인데 이러한 개체 여러 개가 모인 개체군(population)에서 부모의 염색체가 자손에게 계승될 때 선택(selection), 교차(crossover), 돌연변이(mu-

tation) 등의 유전 연산자를 각 세대마다 적용하여 최적해로 진화해 나간다.

유전자 알고리즘은 병렬처리(parallel process)가 가능한 구조로 되어 있어 전역 탐색 능력이 우수하며 수렴 속도가 상대적으로 빠른 장점을 갖고 있다. 그러나 지역 탐색 능력은 상대적으로 약하고, 적용하고자 하는 문제에 따라 부호화 방법과 유전 연산자들을 적절히 바꾸어 주어야 한다. 또한 세대가 지남에 따라 모든 개체들의 유전자형이 비슷해지고 이로 인해 유전적 다양성을 상실하여 더 이상의 탐색을 수행하지 못하고 근사 최적해로 수렴할 수도 있다.

## III. 개체군 변환 유전자 알고리즘

### 1. 코딩 방식

유전자 알고리즘에서 일반적으로 사용하는 코딩 방식은 일차원 배열을 사용하는 것이다. 하지만 실제 생물체의 구조를 살펴보면 양쪽 부모의 유전자가 계승되고 유전자의 우성 및 열성에 따라서 형질의 발현이 결정된다. 일반적인 코딩 방식과는 달리 자연계에서는 이차원 코딩임을 알 수가 있다. 이러한 이차원 코딩 방식을 사용해 염색체를 구성한 것이 바로 이배체 구조이다. 이배체 구조는 유전자형(genotype)과 표현형(phenotype)이 다르며, 유전자형에서 표현형을 계산하는데 있어서는 정해진 방법이 필요하다. 여러 가지 방법에 따라 다양한 코딩 방식이 존재한다. 본 논문에서는 부가구조의 코딩 방식을 사용했다.<sup>[9]</sup> 이러한 코딩 방식은 쌍을 이루는 유전자좌의 값을 더하여 정해진 기준치 값을 기준으로 1과 0의 표현형을 결정하는 방식이다. 표현형이 1에 가까운 것을 1의 정도가 높다고 하고, 0에 가까운 것은 1의 정도가 낮다고 한다.

표현형 = 0	표현형 = 1
AA(4) AB(5) BB(6) AC(9) BC(10)	AD(11) BD(12) CC(14) CD(16) DD(18)
The Degree of Oneness ----->	

그림 2. 1의 정도에 따른 표현형의 매핑  
Fig. 2. The mapping by the degree of oneness in phenotype.

예를 들어 설명하면, 각 대립 유전자를 A, B, C, D라고 하고 각각의 유전자에 2, 3, 7, 9의 값을 할당하여 각 위치의 두 유전자의 합이 10보다 크면 1, 10 이하이면 0으로 표현형을 나타낸다. 이러한 방식은 일차원

코딩 방식에 비해 보다 자연에 가깝다고 할 수 있으며 실험을 통해서 유전적 다양성에 기여하는 것으로 알려져 있다.<sup>[10]</sup> 그림 2는 각 유전자의 조합에 따라 나타나는 표현형을 '1의 정도'를 통해서 설명하고 있다.

### 2. 개체군 변환 유전자 알고리즘

개체군의 크기(population size)는 유전자 알고리즘의 중요한 요소 중 하나이다. 개체군의 크기가 너무 작은 경우는 너무 빨리 수렴하여 전역 최적해를 찾지 못하고 근사 최적해로 수렴하게 되고, 너무 큰 경우는 계산 시간이 너무 많이 걸리게 된다.

개체군 크기의 연속 변환을 이용한 유전자 알고리즘(GA with Varying Population Size : GAVaPS)의 수행과정을 보면 그림 3과 같다.

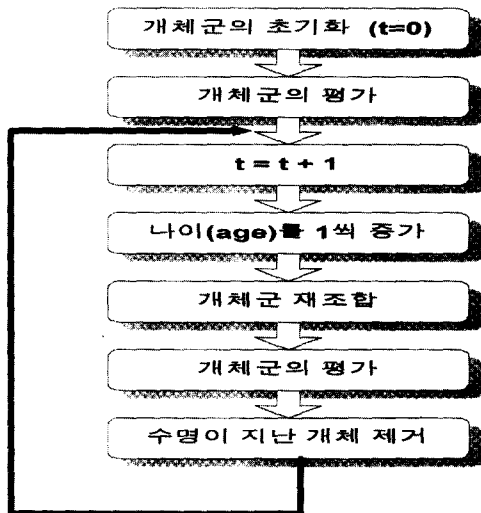


그림 3. 개체군 변환 유전자 알고리즘

Fig. 3. Genetic Algorithm with Varying Population Size.

여기에서 부모를 선택하는 과정은 일반적인 유전자 알고리즘에서 사용하는 방식을 사용하지 않고, 그 대신에 '나이'(age)라는 개념을 도입하였다. 나이는 개체가 생존한 세대수를 말한다. 그런데 염색체의 나이는 선택 과정을 대처할 뿐만 아니라 적합도(fitness)에 의존하고 있고 각각의 세대에서의 개체군의 크기에 영향을 미친다.

t 세대에서 GAVaPS는 개체군 P(t)의 염색체들을 처리한다. 재조합 단계에서는 새로운 추가 개체군, 즉 자손들의 집단이 생성된다. 추가 개체군의 크기(AuxPopSize)는 현 세대의 개체군의 크기(PopSize)

에 대해 일정한 비율만큼 증가한다.

$$AuxPopSize(t) = \lfloor PopSize(t) * \rho \rfloor \quad (\rho: \text{재생산률}) \quad (1)$$

개체군내의 모든 염색체는 염색체의 적합도 값과는 무관하게 똑같은 확률로 재조합을 위해 선택될 수 있다. 자손들은 선택된 염색체에 교차와 돌연변이 등의 유전 연산자를 적용하여 얻는다. 염색체의 선택이 적합도 값에 의존하고 있지 않기 때문에, 즉 선택 단계가 없기 때문에 '나이(age)'와 '수명(lifetime)'이라는 새로운 매개변수를 도입하였다.

### 3. 수명 할당 방식

수명은 평가 단계에서 각각의 염색체에 한 번 할당되는데 개체군을 초기화 한 뒤에는 모든 염색체에 대해서, 그 후로는 재조합 단계를 거쳐 생성된 추가 개체군의 염색체에 할당한다. 염색체의 사망은 염색체의 나이(age)가 수명(lifetime)을 초과하게 되면 발생한다. 따라서, 한 세대가 지난 후의 개체군의 크기는 다음과 같다.

$$PopSize(t+1) = PopSize(t) + AuxPopSize(t) - D(t) \quad (2)$$

여기서  $D(t)$ 는 수명이 다한 개체의 수를 나타낸다.

수명을 각 개체에 할당할 때 주의해야 할 사항으로 평균 적합도보다 높은 적합도를 갖는 개체는 큰 값을 부여하며 낮은 적합도를 갖는 개체는 작은 값을 부여할 수 있어야 하고, 현 세대에 대한 개체군의 크기를 조절할 수 있어서 개체군 크기의 지수적인 증가를 막고 시뮬레이션 코스트를 줄일 수 있어야 한다는 점이다. 개체들의 강화는 추가 개체군에 있는 자손들이 평균 이상으로 할당되는 결과를 가져온다. 각각의 개체가 유전자 재조합을 수행할 확률은 동일한데 수명은 개체가 개체군 내에 존재할 세대수를 결정하기 때문에 각 자손들의 기대치는 수명값에 비례한다. 따라서 평균 이상의 적합도를 갖는 개체는 더 큰 수명값을 주어야 한다. 또한 수명을 계산하는 동안 탐색의 상태도 고려되어야 한다. 이러한 점들을 고려한 수명 할당 방식에는 비례적할당방식(proportional allocation), 선형할당방식(linear allocation), 부선형할당방식(bilinear allocation)이 있다.<sup>[11]</sup>

비례적 할당 방식은 룰렛 휠(Roulette wheel) 선택에서 그 아이디어를 가져왔다. 특정 개체의 수명은 적합도에 비례하여 최소수명 MinLT와 최대 수명

MaxLT 사이에서 나타난다. 그러나 이 방식은 현재까지의 가장 좋은 개체의 적합도에 의해 얻어지는 각 개체의 객관적인 중요성에 관한 정보를 활용하지 않는다. 선형 할당 방식은 현재 존재하는 가장 큰 적합도에 비례하여 각 개체의 수명을 얻는다. 그러나 많은 개체들이 현재의 가장 큰 적합도와 같거나 비슷하다면 그 개체들에 높은 수명을 할당하므로 개체군의 크기가 매우 커지는 문제가 있다. 부선형 할당 방식은 앞의 두 가지 방식을 혼용하였다. 최대 적합도 부근에 존재하는 개체들의 수명을 가급적 차이가 나도록 하는 것인데, 이를 위해 평균 적합도에 관한 정보를 이용하고 또한 최대 최소 적합도 값을 이용한다. 그러나 부선형 할당 방식도 최대 적합도 부근의 개체들의 수명이 차이가 난다고 하더라도 평균 적합도 이상의 개체들의 수와 이하의 개체들의 수가 비슷하므로 개체군의 크기를 적절히 조절할 수 없을 수도 있다. 본 논문에서는 이러한 문제점을 해결하기 위하여 다음과 같은 방법을 제안한다.

lifetime [i] =

$$\begin{cases} \text{MinLT} + \eta \frac{\text{fitness}[i] - \text{MinFit}}{\text{Crit} - \text{MinFit}} \\ \frac{1}{2} (\text{MinLT} + \text{MaxLT}) + \eta \frac{\text{fitness}[i] - \text{Crit}}{\text{MaxFit} - \text{Crit}} \end{cases}$$

if Crit ≥ fitness[i]  
if Crit < fitness[i] (3)

여기서,

- lifetime [i] : i번째 개체의 수명값
- MaxFit : 최대 적합도, Minfit : 최소 적합도
- AbsFitMax : 현재까지 나타난 최대 적합도
- AbsFitMin : 현재까지 나타난 최소 적합도
- MaxLT : 최대수명, MinLT : 최소수명
- $\eta = \frac{1}{2} (\text{MaxLT} - \text{MinLT})$ ,

Crit = MaxFit -  $\sigma$  ( $\sigma$ 는 현재대에서의 적합도의 표준편차)이다.

부선형 할당 방식에서는 분할 기준으로 평균적합도 (average fitness)를 사용하지만<sup>[11]</sup> 제안하는 방식에서는 최대 적합도에서 현재 개체군의 적합도의 표준편차를 뺀 값을 분할 기준으로 사용하였다. 평균적합도를 분할 기준으로 사용했을 경우는 세대가 진화하면 평균 적합도를 넘어가는 개체가 많이 생기므로 이런

개체들에게 높은 수명을 계속 할당해야 하므로 개체군의 크기는 계속 커지게 된다. 하지만 최대 적합도에서 표준편차를 뺀 값을 분할기준으로 사용했을 경우는, 여러 세대가 지날수록 개체들이 최대적합도 부근으로 수렴되므로 수명 할당의 기준선이 평균 적합도보다 높아지게 된다. 따라서 높은 수명을 할당하지 않아도 되므로 자연스럽게 개체군의 크기가 조절된다. 즉 세대가 지남에 따라 대부분의 염색체가 최대값 근방에서 유사한 값을 갖게 되고 표준편차가 작아지므로 초기에는 기준점이 평균적합도에 가까이 있다가 어느 정도 수렴하게 되면 기준점의 위치가 최대적합도 근처로 이동하게 된다. 따라서 대부분의 개체들의 수명이  $1/2(\text{MinLT} + \text{MaxLT})$ 보다 작은 값을 갖게 되어 개체군의 크기를 조절할 수 있도록 하였다.

#### IV. 시뮬레이션

##### 1. 파라미터 최적화 문제

파라미터 최적화 문제(parameter optimization problem)는 유전자 알고리즘을 적용하기가 쉽고, 대부분 2진 스트링(binary string)으로 부호화가 가능한 문제이기 때문에 유전자 알고리즘의 대표적인 응용 분야 중 하나이다. 본 논문에서는 식(4)의 파라미터 x와 y를 조절하여 f(x, y)의 최대값을 구하는 문제를 대상으로 하였다.<sup>[11]</sup>

$$f(x, y) = 0.5 + \frac{\sin^2 \sqrt{x^2 + y^2} - 0.5}{(1 + 0.001(x^2 + y^2))^2}$$

-100 ≤ x, y ≤ 100 (4)

주어진 함수의 탐색공간(search space)은 그림4과 같이 매우 복잡한 형태를 보이고 있다. 초기 개체군의 크기는 20이다. 재생산률  $\rho$ 는 0.4로 주어졌다. 돌연변이 확률은 0.015, 교차확률은 0.65이며, 염색체의 길이는 40으로 주었다. 세대수는 50으로 하였다. 실험과정에서 최대 수명과 최소 수명은 상수로서 각각 MaxLT = 5, MinLT = 1로 하였다.

염색체의 유전자형은 부가구조<sup>[9]</sup>의 이배체를 사용하여 구성하였다. 하나의 유전자 좌에 올 수 있는 대립 유전자는 모두 4가지로 되어 있다. A, B, C, D로 표현하였고 각각의 유전자의 값은 0, 1, 3, 5를 사용하였다. 한 유전자 좌에 표현된 두 개의 대립 유전자의 값의 합이 5 이상이면 표현형으로 1의 값을 갖게 되

고 값의 합이 5보다 작으면 표현형으로 0을 갖는다. 유전자형은 교차와 돌연변이 등의 유전 연산자를 사용할 때 이용하고 표현형은 각각의 염색체를 평가할 때 이용한다.

돌연변이는 4개의 대립 유전자를 사용하기 때문에 대립 유전자들이 순환되어 변화하도록 하였다. 즉 어떤 유전자 좌에서 돌연변이가 발생했을 때 그 유전자 좌의 값이 A면 B로, B면 C로, C면 D로, D면 A로 변환된다.

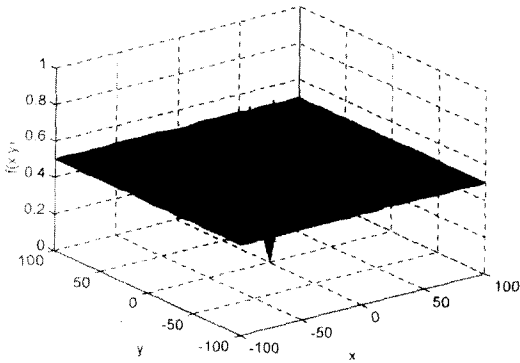


그림 4. 주어진 함수의 탐색 공간  
Fig. 4. Search space of the function.

그림 5는 4가지 수명할당방식(lifetime allocation method)을 썼을 때의 개체군 크기의 변화를 나타내었다. 기존의 할당 방식인 부선행 방식, 선형 할당 방식, 그리고 비례적 할당 방식의 경우는 개체군의 크기가 지속적으로 증가하는 반면 제안하는 방식의 경우는 개체군의 크기가 증가하다가 어느 정도 최적값에 도달했을 때 다시 개체군의 크기가 감소하는 것을 볼 수 있었다. 이는 초기에 적합도의 편차가 상대적으로 클 때는 개체 집단의 크기가 증가하고 최적값 근처에 오면 개체군의 크기가 줄어드는 것을 볼 수 있었다. 선형 할당 방식의 경우는 15세대가 지나면서 개체군의 크기가 1000을 넘어설 정도로 급속히 증가함을 볼 수 있고, 비례 할당 방식이나 부선행 할당 방식이 선형 할당 방식 역시 개체군의 크기가 증가함을 볼 수 있었다.

그림 6은 4가지 수명 할당 방식을 썼을 때의 적합도를 살펴보았다. 4가지 방식이 다소간의 차이는 보였지만, 모두 50세대에 가까워질수록 적합도 값이 0.9에서 1사이로 수렴되는 것을 볼 수 있었다. 제안된 방식에서의 특징은 처음에 적합도의 편차가 상대적으로 클

때는 개체 집단의 크기가 증가하지만 최적값 근처에 오면 개체군의 크기가 줄어들면서 안정된다는 점이다. 그림 7은 이러한 점을 분명히 보여주고 있다. 16세대에서 적합도가 0.978505로 최적해 근처에 이르자 이후 세대에 개체군의 크기가 줄어든다는 것을 볼 수 있었다. 개체군은 17세대에 225개까지 증가하다가 18세대에 187개, 19세대에 183개로 계속 줄어들게 된다.

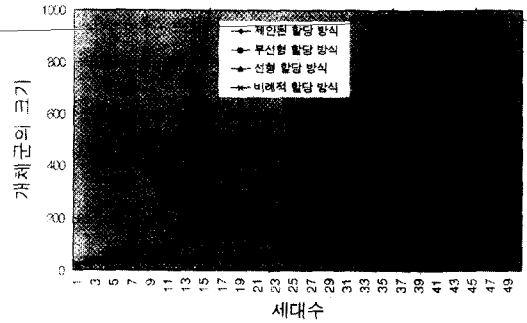


그림 5. 각 수명 할당 방식의 개체군 크기 비교  
Fig. 5. The comparison of population size with each allocation method.

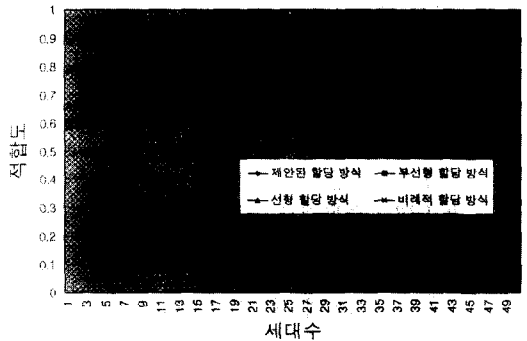


그림 6. 각 수명 할당 방식의 적합도 비교  
Fig. 6. The comparison of fitness values with each allocation method.

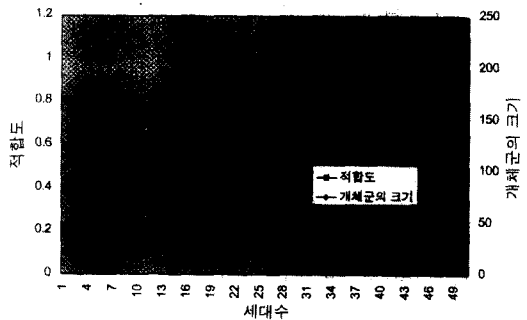


그림 7. 제안된 방식의 세대별 적합도와 개체군 크기의 비교  
Fig. 7. The comparison of fitness values and population size with suggested method.

24세대 이후에 27세대까지 적합도가 다시 0.896146으로 나빠지자 26세대부터 줄어들던 개체군의 크기가 다시금 증가하다가, 28세대부터 적합도가 증가하자 31세대부터 개체군이 줄어들면서 개체군의 크기를 적절히 조절한다는 것을 알 수 있었다.

2. 배낭문제

배낭문제(Knapsack Problem)란 복수의 물체(각각의 물체는 다른 무게와 가치를 가짐)가 주어진 경우에, 무게가 어느 범위 이내에서 물체를 선택하고, 그 때 가치가 최대가 되도록 선택방법을 찾는 조합 최적화 문제(combinatorial optimization problem)이다<sup>[12]</sup>.

적합도 값의 정의는 선택된 물체(Xj)의 가치(Pj)를 모두 합치는 것으로 정의하면 된다. 단지 배낭에 넣는 물체의 무게에는 상한(W)이 있기 때문에, 상한을 넘기는 선택에 대해서는 벌칙으로 상당히 나쁜 적합도를 부여함으로써 의미 없는 해가 생성되는 것을 방지할 수 있다.<sup>[13]</sup>

$$\begin{aligned} \max \quad & \sum_{j=1}^n p_j x_j \\ \text{s.t.} \quad & \sum_{j=1}^n w_j x_j \leq W \\ & x_j = 0 \text{ or } 1 \quad j=1,2,\dots,n \end{aligned} \tag{5}$$

물체의 개수를 20개로 하여 각각에 가치를 부여했고, 초기 개체군은 20개로 하였다. 교차확률은 0.25, 돌연변이 확률은 0.01로 하였고 종료 세대수는 50세대로 제한하여 보았다.

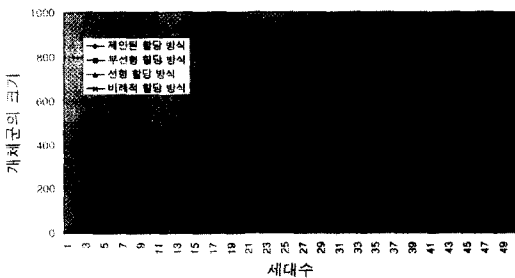


그림 8. 각 할당 방식의 개체군 크기의 비교  
Fig. 8. The comparison of the population size with each allocation method.

그림 8에는 각 할당 방식의 적합도를 나타내었다. 주어진 문제의 경우가 비교적 단순한 조합 최적화 문제였으므로 모두 빠른 세대만에 적합도값(총가치의 합)이 170에서 175사이의 값으로 수렴함을 볼 수 있

었다.

그림 9에는 4가지 수명 할당 방식을 썼을 때의 개체군 크기의 변화를 나타내었다. 함수의 시물레이션과 마찬가지로 제안된 할당 방식을 제외하고는 세대수가 커질수록 모두 개체군의 크기가 증가된다는 것을 볼 수 있었다.

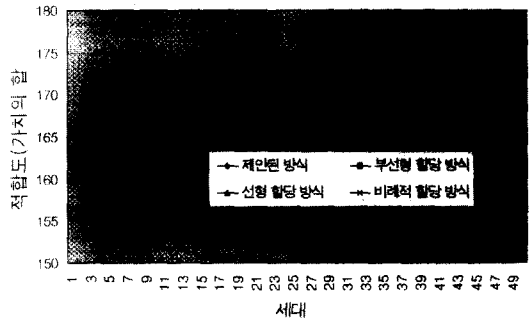


그림 9. 각 할당 방식의 적합도 비교  
Fig. 9. The comparison of fitness values with each allocation method.

본 배낭문제의 예제는 최적해로 빠르게 접근해 가는 문제였으므로, 제안된 방식을 썼을 때에 초반의 세대부터 개체군의 크기가 줄어들면서 98개 정도로 수렴한다는 것을 볼 수 있었다.

V. 결 론

일반적인 유전자 알고리즘은 세대가 지남에 따라 다양성(diversity)을 상실하고 그 결과 불완전 수렴(premature convergence)하는 단점을 갖고 있다. 이러한 단점을 보완하기 위하여 여러 가지 방법이 제시되었으나 기존의 개체군 크기 변환 알고리즘에서의 수명 할당 방식에 의해서는 개체군의 크기가 계속적으로 증가하기 때문에 유전자 알고리즘의 성능에 별다른 향상을 가져왔다고 보기는 어렵다.

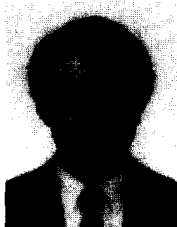
본 논문에서는 기존의 개체군 크기 변환 알고리즘의 수명 할당 방식과는 다른 새로운 수명 할당 방식을 제시함으로써 개체군의 크기가 증가하는 단점을 보완해 보고자 하였다. 제시한 방식의 유용성을 확인하기 위해 파라미터 최적화 문제와 배낭문제에 적용시킨 결과에서, 적합도가 최적값에 이를 때 개체군의 크기가 줄어들면서 안정된다는 것을 볼 수 있었다. 최적값을 찾는 능력에는 기존 수명할당 방식과 큰 차이를 보이지 않지만 개체군의 크기를 안정화시킴으로써 개체군 크

기의 급격한 증대로 인한 여러 문제점을 해결할 수 있다.

### 참 고 문 헌

- [1] Davis, L. and Steenstrup, M. "Genetic Algorithms and Simulated Annealing", Morgan Kaufmann Publisher, Los Altos, CA, 1987.
- [2] Cleveland, G. A. and Smith, S. F. "Using Genetic Algorithms to Schedule Flow Shop Release," Proc. of ICGA-89, 1989.
- [3] Goldberg, D. and Lingle, R. "Alleles, loci, and travelling salesman problem," Proc. of ICGA-85, 1985.
- [4] Gabbert, P., Brown, D., Huntley, C., Markowicz, B. and Sappington, D. "A System for learning routes and Schedules with Genetic Algorithms," Proc. of ICGA-91, 1991.
- [5] Holland, J. "Adaptation in Natural and Artificial Systems", The University of Michigan, 1975.
- [6] Manfoud, S. "Crowding and Preselection Revisited," Department of Computer Science University of Illinois, 1992.
- [7] Arabas, J., Michalewicz, Z., and Mulawka, J. "GAVaPS - a Genetic Algorithm with Varying Population Size," Proceeding of the Evolutionary Computation Conference, part of the IEEE World Congress on Computational Intelligence, 1994.
- [8] Greene, F. "A method for Utilizing Diploid/Dominance in Genetic Search," Proceeding of the Evolutionary Computation Conference, part of the IEEE World Congress on Computational Intelligence, Orlando, 26-29 June 1994.
- [9] Ryan, C. "The Degree of Oneness," <http://www.bioele.nuee.nagoya-uac.jp/ws/c1/papers/p007.htm>.
- [10] David E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley, pp. 148-157, 1989.
- [11] Michalewicz, Z. "Genetic Algorithms + Data Structures = Evolution Programs", Springer-Verlag, pp. 72-80, 1995.
- [12] 기타노 히로아키, "유전자 알고리즘", 대청 컴퓨터 월드, 서울, pp. 197-200, 1993
- [13] Gen, M. Cheng, R. "Genetic Algorithms & Engineering Design", Wiley Interscience, pp. 97-98, 1996.

### 저 자 소 개



權奇浩(正會員)

1975년 성균관대학교 전자공학과 졸업.  
1978년 서울대학교 전자공학과 석사학위 취득. 1978년 ~ 1980년 ETRI. 1998년 서울대학교 전자공학과 박사학위 취득. 1989년 성균관대학교 전자공학과 교수. 1996년 Texas A&M 교환교수. 관심분야는 카오스, 퍼지, 신경 회로망, 유전자 알고리즘 등