개미 집단을 외삽한 적응형 유전 알고리즘(ANT-GA)

김중항° 이세영 장형수*
서강대학교 컴퓨터학과
{pero°, philolight, hschang}@smolab.sogang.ac.kr

Sogang University

Joong Hang Kim° Se-young Lee Hyeong Soo Chang*
Department of Computer Science, Sogang University

요 약

본 논문에서는 개미 집단을 외삽한 새로운 적응형 유전 알고리즘을 제안하고, 최적해로의 수렴이 어려운 여러 가지 대표적인 함수들에 대한 실험을 통하여 제안된 알고리즘이 우성 형질의 유전 알고리즘보다 더 빠른 속도로 최적해에 수렴하며 파라미터 값에 따른 유연성을 가지고 있는 알고리즘임을 확인하였다.

1. 서 론

본 논문에서는, 개미 집단을 적용한 새로운 적응 유전 알고리즘(Genetic Algorithm, GA)을 제안한다. 이미 잘 알려져 있는 기본적인 유전 알고리즘(Canonical Genetic Algorithms, CGA)[1]은 수행 과정에서 주어진 최적화 문제에 대한 2진의 문자열(string)로 인코딩된 하나의 해에서 각 비트를 고정된 확률로 변이(mutation)시킨다. 이에 반하여, 본 알고리즘에서의 각 비트의 변이는 해당 비트와 관련된 변이 확률 분포와 "군지능(swarm intellignce)"을 적용하여 이루어진다. 본 알고리즘은 선택(selection)과 교배(crossover)의 종류에 무관하다.

CGA와 CGA의 수많은 변형들[2]은 보통 개체(chromosome)들이 가진 비트들의 상관관계나 상호 작용을 이용하지 않는다. 본 알고리즘은 한 군집(population)내의 개체들의 사회적 행동에 기초하여 각 비트의 위치에 대한 변이 확률 분포를 갱신(즉, 각 개체들이 얼마나 최적인가에 대한 "집합적"인 지식을 만듦)함으로써 개체군들의 조망(landscape)을 "학습"하고, 다음 세대의 군집을 만들어내는 데 사용하게 된다.

ANT-GA로 불리는 본 알고리즘은 변이 과정의 연산에 있어서 Gutjahr의 Graph-based Ant Colony Optimization(GACO)[3]의 확장을 기본으로 한다. GACO는 개념적인 개미들의 이동을 시뮬레이션하고 목적 함수(objective function)에 대한 사후 평가(post-hoc evaluation)에 기초한 "좋은" 경로의 강화를 통하여 임의의 조합 문제(combinatorial problem)의 목적 함수를 최적화하는 데 응용된다. Graph-based ant system[3]에서는 개미들의 이동을 나타내기 위한 구조로 유향 그래프(directed graph)를 사용한다. 조합 최적화 문제의 가능한 해들은 그래프의 경로들로 인코딩되며, 각 반복(iteration)의 처음에 개미들은 그래프상의 같은 노드에 위치한다. 개미들은 반복이 진행됨에 따라, 현재 노드의 하위 노드를 임의적으로 선택하게 된다. 각 하위 노드로 이동할 확률은 그래프상의 간선(arc)들에 부여된 "페로몬 값"으로부터 계산된다.(또한, "가시도 값(visibility value)"이라 불리는, 계속적인 이동이 가능한지를 선평가(pre-evaluation)한 값이 사용됨) 모든 개미가 이동을 끝내고 나면, 해당 경로들은 문제에 대한 해로 디코딩되어 평가된다. "유망한" 경로에 해당하는 간선들의 페로몬을 증가시켜서 강화시키게 되고, 이 과정이 반복된다. 이는 주어진 문제에 대한 최적 경로를 찾아내는 과정(GA에서 최적해를 찾아내는 과정)으로 ANT-GA의 변이 단계에 외삽(extrapolation)된다.

기본적으로, GA에서 임의의 확률 분포로 0이나 1의 값을 가지는 2진 문자열의 각 비트를 확률 변수로 보고, 이를 그래프의 노드와 개미의 한 번의 성공적인 움직임(one-step walk)으로 표시되는(각각 0과 1에 대응) 2개의 간선(비트에서 나오는 간선과 다른 비트로 들어가는 간선)에 대응시켜 그래프를 구성한다. 노드의 부분적인 간선에 남겨진 페로몬 값은 해당 비트의 확률 분포를 변화시키고 이는 ANT-GA의 변이 단계에서 사용된다. 페로몬 값의 생성 과정은 개미들의 이동에 근거하며, 이를 GA의 선택과 교배 작용의 기본 개념과 결합시켜 확장한다. 짧게 요약하면, 주어진 군집(GA의 2진 문자열의 집합)에서 특정 선택 및 교배를 하여 임시 개체군을 얻고 각 비트에 대한 확률 분포를 갱신한 후(페로몬 갱신) 새롭게 만들어진 확률 분포를 통해 현재 군집의 각 문자열(개미들의 경로)을 변이시켜 새로운 다음 세대의 군집을 생성한다.(자세한 것은 2장 참조)

* 이 논문은 2004년도 한국학술진흥재단의 지원에 의하여 연구되었음(KRF-2004-003-D00294).

2. Graph-based ACO의 extrapolation : ANT-GA

$X=\{0,1\}^n$가 길이 $n<\infty$이고 $f:X\rightarrow R^+$인 양의 적합도 함수(fitness function)를 가진 2진 문자열의 유한 해 공간일 때,

$$\min_{x\in X} f(x)$$

인 최적화 문제를 생각해 보자. 문제의 목표는 $\min_{x\in X} f(x)$를 만족하는 $x^*\in X$를 찾는 것이다. 본 논문에서는 해 $x\in X$의 i번째 비트의 위치를 x_i로 나타낼 것이다.

ANT-GA의 각 단계에 대한 개념적인 설명은 다음과 같다.

ANT-GA

- 초기화:

  군집의 크기 $m>0$을 결정하고, $\mu\in(0,1]$을 결정한다. $k=0$으로 초기화한다. 초기 군집 $P(0)$를 임의로 초기화하고, $x_0^e=\arg\min_{(x\in P(0))} f(x)$으로 정한다.

- 반복:
  - 적합도 평가: $x\in X$에서, $f(x)$를 평가하여 $x_k^*=\arg\min_{(x\in P(k))} f(x)$를 구한다.
  - 선택/교배: 선택/교배로 임시 군집 $P_I(k)$를 생성한다.
  - 변이 확률 θ의 적응:
    * 우성(Elite) 문자열(최적 해) x_k^e를 구한다.

      $x_k^e=\arg\min_{\{x\in\cup_{j=0}^{k-1}P(j)\cup P_I(j)\}} f(x)$
    * x_k^e와 식(1)과 (2)를 사용하여 ψ함수 값을 갱신한다.
    * 식(3)으로 θ값을 갱신한다.
  - 변이:

    θ값을 통하여 $P_I(k)$를 변이시켜, 다음 세대의 군집 $P(k+1)$를 생성한다.
  - $k\leftarrow k+1$

현재 군집 $P(k)=\{x^1,\dots,x^m\}(x_i\in X,\ i=1,\dots,m,\ m$은 고정된 값))로부터, $k\ge 0$인 각각의 반복 단계에서 $P(k)$의 해 혹은 문자열을 평가하여

$$x_k^* = \underset{\{x\in P(k)\}}{\arg\min} f(x)$$

를 얻는다. 즉, 현재 군집에서의 최적 해를 얻는다. 그 다음 해들을 선택하고, 임의의 방법으로 $P_c\in[0,1]$의 확률을 가지고 교배를 시행한다. 교배 방법은 1점(one-point) 교배, 2점(two-point) 교배, 균등(uniform) 교배[4], 혹은 P_c를 개량(tuning)한 적응(adaptive) 교배[5] 등을 사용한다. 교배를 통하여, $x_c^i\in X$이고 $i=1,\dots,m-1$인 $P_c(k)=\{x_k^*, x_c^1,\dots,x_c^{m-1}\}$인 임시 군집을 생성한다. 본 알고리즘에서는 선택, 교배의 방법을 자유롭게 선택하여 $P(k)$로부터 $m-1$개의 문자열을 생성하는 것을 가정한다. 만일 $P_c=0$이면, $P(k)=P_c(k)$이다. 이제, $\theta_i(k)$가 주어진 2진 문자열의 i번째에 위치한 비트가 '1' 값을 가질 확률('0'값을 가질 확률은 $1-\theta_i(k)$)이라 하고, $i=1,\dots,n$에 대하여 변이 확률 $\theta_i(k)$를 갱신한다.(CGA에서, $\theta_i(k)=P_m\in(0,1)$이다. CGA가 수행되는 동안 문자열의 각 비트에 대한 변이 확률은 같다.) 이때 최적인 해를 함께 갱신해야 한다. $k\ge 1$의 반복 동안, $P(0)\cup P_c(0)\cup\cdots\cup P(k-1)\cup P_c(k-1)$에서 최적 해 x_k^e

$$x_k^e = \underset{\{x\in\cup_{j=0}^{k-1}P(j)\cup P_c(j)\}}{\arg\min} f(x)$$

를 구한다.

변이 확률을 가지고 $P_c(k)$의 각 해를 변이시켜 $P(k+1)$을 얻는다. 예로, $n=3$인 k번째 반복에서, 2진 문자열 001은 확률 $\theta_1(k)(1-\theta_2(k))(1-\theta_3(k))$로 100으로 변이된다.

변이 확률은 $P_c(k)$의 해에 대한 적합도 값과 최적 해 x_k^e(강조된 "유망한" 해)의 적합도 값을 기초로 하여 갱신된다. $P_c(k)$를 정렬된 집합 $\{x^1,\dots,x^m\}$라고 하고, 각각의 해 번호(solution index) $l=1,\dots,m$과 $s=0,1$, 그리고 각 비트의 위치 $i=1,\dots,n$에 대하여, 함수 ϕ를 다음과 같이 정의한다.

$$\psi_k^l(i,s)=\begin{cases}\varphi(f(x^l)) \text{ if } f(x^l)\le f(x_k^e) \text{ and } x_i^l=s\\ 0 \text{ } otherwise\end{cases} \quad (1)$$

φ는 감소함수(non-increasing function)이다. 예를 들어 $f(x^l)\le f(x_k^e)$일 때, $\varphi(f(x^l))=1/f(x^l)$로 선택할 수 있다.

만일 $\sum_{i,s}\sum_{l=1}^{m}\psi_k^l(i',s')=0$일 경우 각각의 i와 s에 대하여, $\psi_k(i,s)=\psi_{k-1}(i,s)$라고 하고, 그렇지 않을 경우는 $s=0,1$과 $i=1,\dots,n$에 대하여

$$\psi_k(i,s)=(1-\mu)\psi_{k-1}(i,s)+\mu\frac{\sum_{l=1}^{m}\psi_k^l(i,s)}{\sum_{i,s}\sum_{l=1}^{m}\psi_k^l(i',s')} \quad (2)$$

을 사용하여 $\psi_k(i,s)$를 구한다. 계수 $\mu\in(0,1]$은 이전 값의 강도를 조절하기 위한 알고리즘의 매개 변수로 사용된다.

ψ의 식 (1)과 (2)에 의하여, "유망한" 비트의 위치와 0과 1의 값에 대한 ψ 함수 값은 증가되고, 비트의 값과 위치는 미래의 변이 과정에 의하여 다음 세대에서 더 많이 나타나게 된다.

$\theta_i(k)$의 갱신은 $i=1,\dots,n$에 대하여, 다음과 같다.

$$\theta_i(k+1)=\frac{\psi_k(i,1)}{\psi_k(i,0)+\psi_k(i,1)} \quad (3)$$

최초 군집 $P(0)$는 임의의 2진 문자열로 초기화하고, x_0^e는 $x_0^e=\arg\min_{x\in P(0)}f(x)$으로, ψ_{-1}^l은 $s=0,1$과 $l=1,\dots,m$ 그리고 $i=1,\dots,n$에 대하여 $\psi_{-1}^l(i,s)=0.5$로 초기화 한다.

3. 성능 평가

ANT-GA의 성능을 평가하기 위하여, 우성 지배의 유전 알고리즘(Genetic Algorithm with Elitism, EGA)이라 불리우는 알고리즘[6]을 평가 대상으로 삼았다. 두 가지의 알고리즘은 동일한 기계와 환경에서 실험하였다.

ANT-GA와 EGA에서 문제에 대한 해들의 인코딩 및 디코딩은 실수의 디지털 샘플링 기법(quantization of real value)[7]에 수정을 가하여 실수를 2진 문자열로 바꾸었다. 미리 주어진 한계값(maximum depth) $x_{\min}=(0,1)$에 대하여, $A\le x_{\min}$이고 $x_{\max}=A\sum_{i=1}^{N}2^{i-1}$를 만족하는 N과 가장 큰 A를 구한 다음, $u_n^k=\{0,1\}$인 $u_n=(u_n^1,u_n^2,\dots,u_n^N)^T$의 문자열 u_n이 있다고 할 때

$$x_n=A\sum_{i=1}^{N}u_n^i2^{i-1}$$

으로 나타내었다.

초기 군집인 $P(0)$에서의 임의로 설정된 각 해들은 동일하게 적용하였으며, ANT-GA, EGA 모두 개체군의 수는 50으로 고정하고, 토너먼트 선택법(tournament selection)과 0.5의 교배 확률을 가지는 2점 교배법을 사용하였다. EGA의 경우에는 변이 확률을 0.5로 고정하였으며, ANT-GA의 경우 μ값을 0.8로 고정한 뒤 실험을 수행하였다. 각각 100회의 수행 후 평균값을 구하여 수치화 하였다. 실험에 사용된 함수들은 전체 최적화 문제(global optimization problem) 중에서 많은 부분 최적값(local optimal)을 가짐으로써 전체 최적(global optimal)을 찾아내는 데에 어려움이 있는 대표적인 함수들 중에서 선택되어 실험되었다. 지면의 부족으로 인하여 함수 전부에 대한 결과를 나타내지는 못하였으나, 모두 그림 1~4와 비슷한 그래프를 보여주었다. 실험에 사용된 함수 중 대표적으로 선정된 몇 가지에 대한 설명은 다음과 같다.

F1) De Jong의 4번째 함수(4th De Jong's function, quartic model with noise)[8]

$$F=\sum_{i=1}^{2}ix_i^4+Gauss(0,1)$$
$$x_i\in[-1.28,1.28],\ F_{\min}=0$$

F2) V-sinc 확장 함수(stretched V-sinc function, Ackley's)[9]

$$F=(x^2+y^2)^{0.25}(\sin^2(50(x^2+y^2)^{0.1})+1)$$
$$x,y\in[-10,10],\ F_{\min}=0$$

F3) Griewank의 함수(Griewank's function)[10]

$$F=-\prod_{i=1}^{2}\cos\left(\frac{x_i}{\sqrt{i}}\right)+\sum_{i=1}^{2}\frac{x_i^2}{4000}+1$$
$$x_i\in[-20,20],\ i=1,2,\ F_{\min}=0$$

그림 1~3에서는 ANT-GA와 EGA의 초기 20세대까지의 해당 세대의 군집에서의 최적 해를 표시하였다. F1~F3까지의 3개의 함수 모두에서, ANT-GA가 EGA보다 더 빠른 속도로 전체 최적 값에 가까워져 가는 것을 알 수 있다.

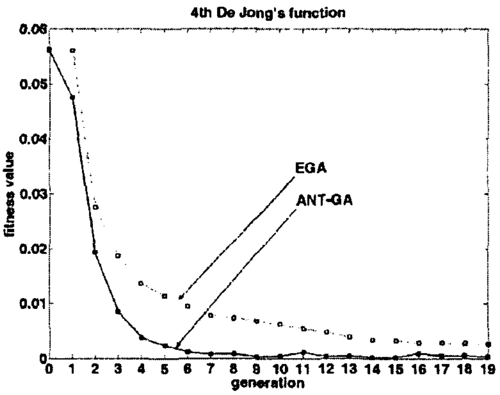

그림 1 : F1에 대한 실험 결과(최적 해)

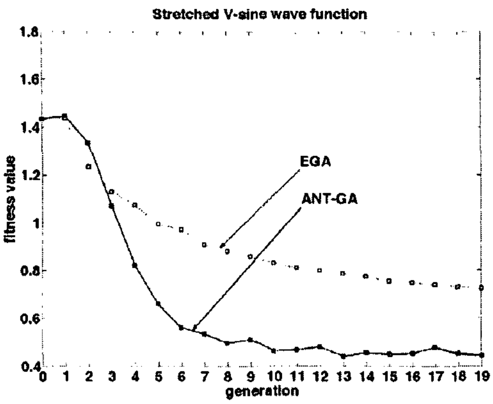

그림 2 : F2에 대한 실험 결과(최적해)

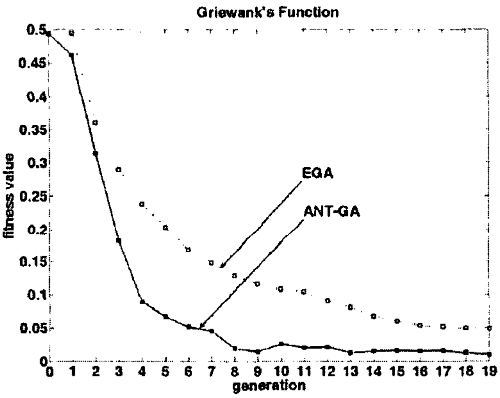

그림 3 : F3에 대한 실험 결과(최적해)

그림 4에서는 F3의 함수를 가지고 ANT-GA를 실험하되, 식(2)에서의 μ값을 0, 0.3, 0.6, 0.9로 차례로 변화시키면서 100세대까지의 해당 세대의 최적해를 구하였다.

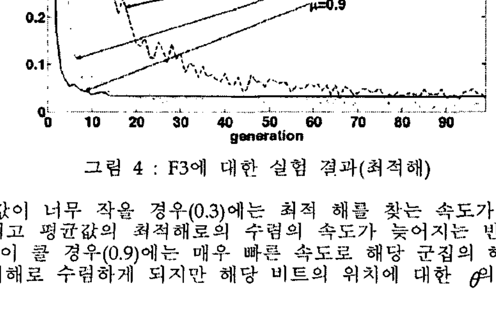

그림 4 : F3에 대한 실험 결과(최적해)

μ값이 너무 작을 경우(0.3)에는 최적 해를 찾는 속도가 매우 느리고 평균값의 최적해로의 수렴의 속도가 늦어지는 반면에, μ값이 클 경우(0.9)에는 매우 빠른 속도로 해당 군집의 해들이 최적해로 수렴하게 되지만 해당 비트의 위치에 대한 θ의 값이 초기에 너무 커지거나 작아지기 때문에 최적 해를 찾는 것이 어려워지는 경향을 보인다. 이러한 점으로 볼 때 μ값은 해당 개체의 해들이 현재의 최적 해로 얼마나 빨리 수렴할 것인지(혹은 현재의 최적 해보다 더 나은 해를 찾아다닐 것인지)를 결정하는 중요한 매개 변수가 된다는 것을 알 수 있다. 실제 실험에서는 μ값이 0.4~0.8 사이일 때 가장 좋은 결과를 보였다.(0.6)

3. 결론

우리는 군지능의 아이디어를 변이 과정에 적용한 새로운 적응형 유전 알고리즘을 개발하였다. 흥미로운 문제들에 대하여, 개미 집단 최적화가 CGA보다 좋은 결과를 보이는 것은 여러 논문에서 다루어져 왔다.[11][12][13] CGA의 구조에 군지능 최적화의 방법론을 병합함으로써, 결과인 본 논문의 혼성(hybrid) 알고리즘은 CGA의 약점을 보완할 수 있게 되었다.

본 논문에서의 여러 가지 전체 최적화 문제에 대하여, 새롭게 제안된 우리의 알고리즘은 EGA에 비해 더 빠른 수렴 속도를 보이는 것이 확인되었다. 또한 적당한 μ값을 정함으로써 수렴 속도와 최적해의 탐색 가능성을 조절할 수 있는 유연성 있는 알고리즘임을 확인할 수 있었다.

4. 참고 문헌

[1] G. Rudolph, "Convergence Analysis of Canonical Genetic Algorithms," *IEEE Trans. on Neural Networks*, vol. 5, no. 1: 96-101, 1994.
[2] C. R. Reeves, "Genetic Algorithms for the Operations Resercher," *INFORMS J. on Computing*, vol.9, no. 3: 231-250, 1997.
[3] W. J. Gutjhar, "A Graph-based Ant System and Its Convergence," *Future Generation Computer Systems*, vol. 16, no. 8, 2000.
[4] W. M. Spears and K. A. De Jong, "An Analysis of Multipoint Crossover," in *Proc. 1990 Workshop of the Foundations of Genetic Algorithms*, 1991: 301-315.
[5] M. Srinivas and L. M. Patnaik, "Adaptive Probabilities of Crossover and Mutation in Genetic Algorithm," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 24, no. 4: 656-667, 1994.
[6] D. Bhandari, C. A. Murthy, and S. K. Pal, "Genetic Algorithms with Elitist Model and Its Convergence," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 10: 731-747, 1996.
[7] K. Najim, A. S. Poznyak, and E. Ikonen, "Optimization based on a team of automata with binary outputs," *Automatica*, vol. 40: 1349-1359, 2004.
[8] K. De Jong, "An Analysis of the Behaviour of a Class of Genetic Adaptive Systems," PhD thesis, University of Michigan, 1975
[9] D. H. Ackley, *A Connectionist Machine for Genetic Hillclimbing*, Kluwer Academic Publishers, 1987
[10] A. Griewank, "Generalized Descent for Global Optimization," *J. of Optimization Theory and Applications* , vol. 34: 11-39, 1981
[11] M. Dorigo, V. Maniezzo, A. Colorni, "The Ant System: Optimization by a Colony of Cooperating Agents," *IEEE Trans. Systems Man Cybernet.*, vol. 25:29-41, 1996.
[12] R. C. Eberhart and J. Kennedy, *Swarm Intelligence*, Morgan Kaufmann, 2001.
[13] M. Dorigo, G. Di Caro, and T. Stützle, Special Issue on "Ant Algorithms," *Future Generation Computer Systems*, vol. 16, no. 8, 2000.