The Thirteenth International Symposium on Artificial Life and Robotics 2008(AROB 13th '08),
B-Con Plaza, Beppu, Oita, Japan, January 31-February 2, 2008

Adaptive Crossover, Mutation and Selection Using Fuzzy system for Genetic Algorithms

Soung-Min Im and Ju-Jang Lee
Electrical Engineering and Computer Science
Korea Advanced Institute of Science and Technology(KAIST)
373-1, Guseong-dong, Yuseong-gu, Daejeon, 305-701, Korea
taronism@kaist.ac.kr, jjlee@ee.kaist.ac.kr

Abstract-Genetic algorithms use a tournament selection or a roulette selection to choice better population. But these selections couldn't use heuristic information for specific problem. Fuzzy selection system by heuristic rule base help to find optimal solution efficiently. And adaptive crossover and mutation probabilistic rate is faster than using fixed value. In this paper, we want fuzzy selection system for genetic algorithms and adaptive crossover and mutation rate fuzzy system.

Index Terms-Genetic algorithms(GA), fuzzy logic system, adaptive genetic algorithm.

1 Introduction

GENETIC ALGORITHMS(GA) are revolutionary random guided search techniques based on a Darwinian survival-of-the fittest and genetic mechanism [1]. In other words, we select genes which adapted given environment, crossover each genes and sometime mutate for new chromosome. Good gene characteristic is reproducted new generation. After several evolution their remain better adaptive gene to given environment. GAs are parallel-search procedures and stochastic less likely to get trapped in local minima. This paper is organized as follows. In section 2, a brief review to GA and explanation about important parameter. In section 3, simulation about each adaptive experiments, and the conclusion is made in section 4.

2 A BRIEF REVIEW TO GA

GA have parameters and variables for control the algorithms. There are evolution operation, genetic operations and parameter settings. Simple GA described in fig. 1. [2]

```
Simple Genetic Algorithm()
{
initialize population;
evaluate population;
while (not termination-condition) do
   {
   select population for next population;
   perform crossover and mutation;
   evaluate population;
   }
}
```

Figure 1: Simple Genetic Algorithm

First, evolution operation is selection. Typical method for selection are roulette, tournament, stochastic uniform, remainder and Uniform selection. Selection is only operation for evolution. So choosing a selection method is quit important.

Second, genetic operations have three steps, crossover operation, mutation operation and reproduction. If chromosomes presented bit strings, crossover operator can be one point crossover or n point crossover, also mutation operator can be one point mutation or n point mutation. Real number lists case scattered, intermediate, heuristic and arithmetic crossover operator can be used, and gaussian, uniform and adaptive feasible mutate operator can be used for genetic operations. Reproduction method is used to apply elitism.

Third parameter settings, there are two main parameter, crossover rate and mutation rate[3]. The value rely on what kind problems given. In simple problem, increasing crossover rate will reduce convergence time, and decreasing crossover rate will sometimes have better solution. And increasing mutation rate also effect convergence time, and accuracy about solution. But each problem has their own's characteristic, and use different values for crossover and mutation rate.[4]

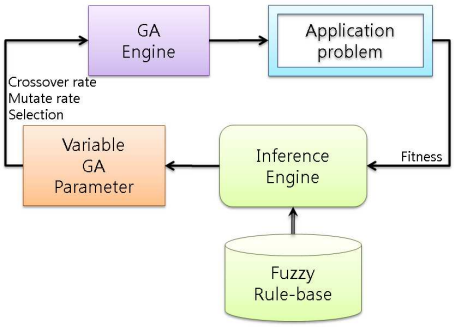

Figure 2: Adaptive Crossover, Mutation and Selection Using Fuzzy system for Genetic Algorithms

```
Adaptive Parameter Genetic Algorithm()
{
initialize population;
decode population;
evaluate population;
keep the best population;
while (not termination-condition) do
   {
   select population for next population;
   perform crossover and mutation;
   evaluate population;
   elitist
   alter Px, Pm, selection parameter
   }
}
```

Figure 3: Adaptive Parameter Genetic Algorithm

3 Adaptive Selection, Crossover and Mutation

There exist trade-off with parameter values. For example, at starting time their need big crossover rate and small mutate rate, and when generation reach to termination condition need small crossover rate and big mutate rate for improve solution [5]. So using that kind of heuristic information can be used fuzzy logic system, in fig. 2. Algorithm is described in fig. 3.

3.1 Adaptive selection

Test function-Schaffer 2 problem (SF2) (Michalewicz, 1996) [6]. Schaffer's function used for modeling several real-world problem like emerging areas of wireless networks. The function defined in (1) and two-dimension(2-D) case plotted in fig. 4.

$$f(x) = \left(\sum_{i=1}^{n} x_i^2\right)^{1/4} \cdot \left[sin^2 \left(50 \cdot \left(\sum_{i=1}^{n} x_i^{2q} \right)^{1/10} \right) + 1.0 \right]$$

$$x_i \in [-32.767\ 32.767]\,,\ n = 5,\ offset = 1000. \qquad (1)$$

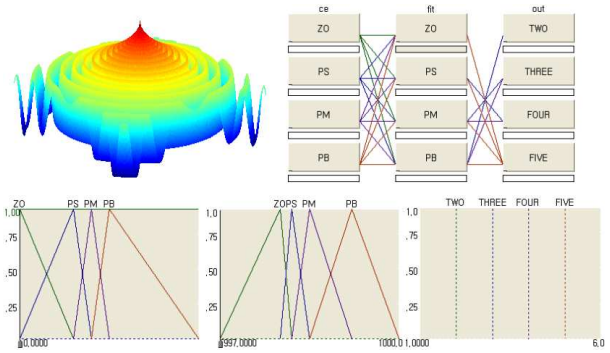

Figure 4: Fuzzy membership functions and rule base for adaptive tournament candidate's number

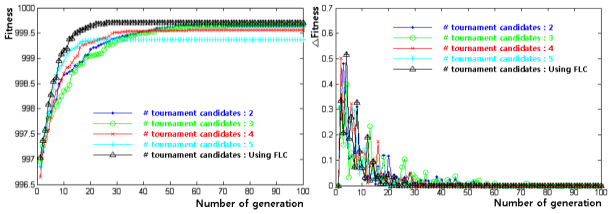

Figure 5: Comparison of the number of tournament candidates change and adaptive number of tournament candidates

The global optimum x*=(0,...,0) with f(x*)=1000, offset=1000. To find characteristic about given problem we use 300 population, 50 chromosomes, fixed crossover rate 1.0, mutation rate 0.0 and repeat 100. In fig 5. several number of tournament selection candidates bring about different result. Large number of candidates raise fast convergence. But accuracy wasn't good. So using this heuristic information we design fuzzy logic system. fig. 4.

3.2 Adaptive crossover rate

Test function-Rastrigin problem (RG) (Storn and price, 1997) [6]. The function defined in (2) and two-

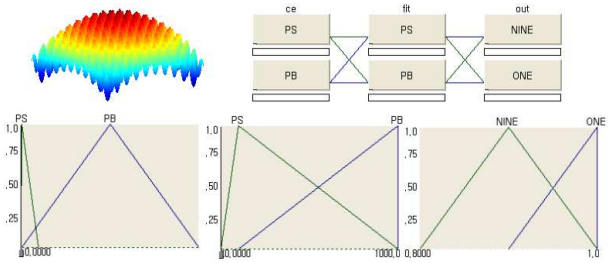

Figure 6: Fuzzy membership functions and rule base for adaptive crossover rate

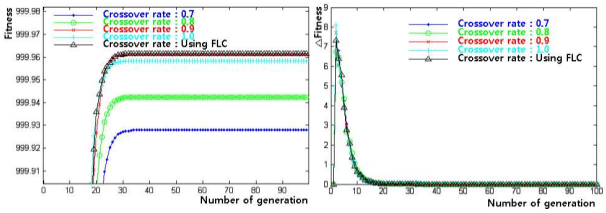

Figure 7: Comparison of the different crossover rate

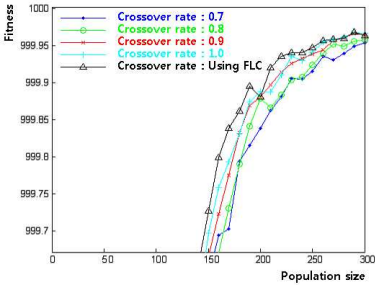

Figure 8: Comparison of the different crossover rate when population size change

dimension(2-D) case plotted in fig. 6. The global optimum x*=(0,...,0) with f(x*)=1000, offset=1000.

$$f(x) = 10n + \sum_{i=1}^{n} \left(x_i^2 - 10\cos\left(2\pi x_i\right)\right)$$

$$x_i \in [-5.12\ 5.12]\,,\ n = 5. \tag{2}$$

We use 300 population, 50 chromosomes, fixed mutation rate 0.0, repeat 100, and change the crossover rate from 0.7 to 1.0. In fig 7, crossover rate 1.0 is faster than Crossover rate 0.9, but Crossover rate 0.9 is more accurate. So we can design to mix that kind of performance in fig 7. After adding fuzzy logic system there exist improve performance. And this fuzzy logic

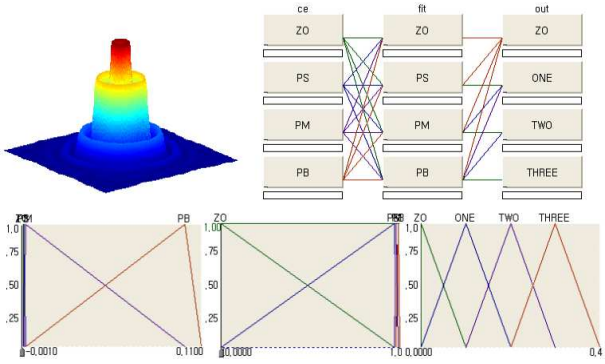

Figure 9: Fuzzy membership functions and rule base for adaptive mutation rate

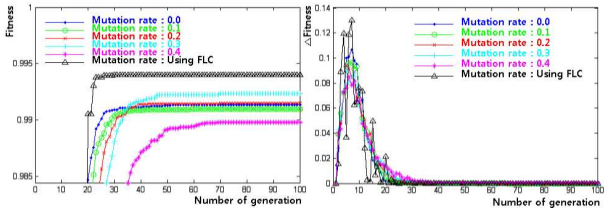

Figure 10: Comparison of the different mutation rate

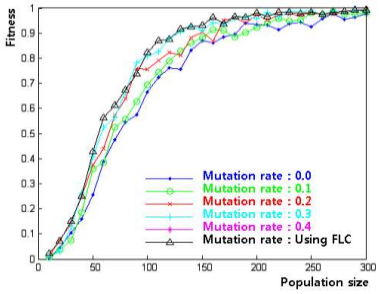

Figure 11: Comparison of the different mutation rate when population size change

system also same influence when the population size change from 10 to 300. In fig 8.

3.3 Adaptive mutate rate

Test function-Schaffer problem (SF) another version [7]. The function defined in (3) and two-dimension(2-D) case plotted in fig. 9. The global optimum with x*=(near zero), f(x*)=0.99400693.

$$f(x) = \frac{sin^2\left(\sqrt{\sum_{i=1}^{n} x_i^2}\right)}{1.0 + 10^{-3} \cdot \left(\sum_{i=1}^{n} x_i^2\right)^2}$$

$$x_i \in [-16.383\ 16.383]\,,\ n = 5. \tag{3}$$

We use 300 population, 50 chromosomes, fixed crossover rate 1.0, repeat 100, and change the mutation rate from 0.0 to 0.4. In fig 10, lower mutation rate rapidly convergence, bigger mutation rate find accurate solution, except mutation rate over 0.4. Using these heuristic information we can design fuzzy logic system like fig. 9. And this fuzzy logic system also same influence when the population size change from 10 to 300, in fig. 11.

4 Conclusions

This paper has presented adaptive crossover, mutation and selection using fuzzy system for GA. Three different 5-dimensional problems used to implement

each adaptive algorithms. The aim is better performance by heuristic information implemented fuzzy logic system. After finding characteristic given problem through changing major parameter, design inference engine to avoid trade off about parameter. So the result is faster and more accurate than simple GA. Furthermore studies showed the result after change population size.

References

[1] J. R. Koza, "Genetic Programming". Bradford Book, 1993.

[2] D. E. Goldberg, "GeneAlgorithms in Search, Optimization, and Machine Learning". Reading, Ma: Addison-Wesley, 1989.

[3] Zbigneiw Michalewicz, "Genetic Algorithms + Data Structures = Evolution Programs, Third, Revised and Extended Edition", Springer, 1999.

[4] L C Jain, R K Jain, "Hybrid Intelligent Engineering systems", World Scentific, Advances in Fuzzy Systems-Applications and Therory Vol.11, 1997.

[5] Shengxiang Yang, "Adaptive Crossover in Genetic Algorithms Using Statistics Mechanism", in Artificial Life VIII. Standish, Abbass, Bedau (eds)(MIT Press) 2002, pp182-185.

[6] M. Montaz Ali, Charoenchai Khompatraporn and Zelda B. Zabinsky, "A Numerical Evaluation of Several Stochastic Algorithms on Selected Continuous Global Optimization Test Problems", Journal of Global optimization, Springer, 2005.

[7] Chang Wook Ahn and R. S. Ramakrishna, "Elitism-Based Compact Genetic Algorithms", IEEE Transactions on Evolutionary computation, Vol. 7, No. 4, August 2003.