Combining Roulette wheel and Chromosome differentiation methods in Genetic Algorithm

D. O. Okhonko

Genetic algorithm (GA) is a heuristic algorithm that use idea of natural evolution in order to solve optimization and search problems. One of the most popular selection method in GA is "Roulette wheel" method. Our approach use hybrid of this and Chromosome differential technique. In addition, switching between this two methods is dynamic. This paper contain research about efficiency of this method and examples of application.

Introduction

Genetic Algorithm [1] is widely used in finding maximum or minimum of some function. This problem may be solved easily using brute force approach with approximation but often it is impossible in high-dimensional space. In Roulette wheel method [3] main idea is that as higher value of fitness function of chromosome as probably it will be selected for crossover. This approach use the same principle as Monte Carlo method. But if function has some fast-growing local extremum value, hybrid method will be much more effective.

Problem in Roulette wheel method

Let's look at simple two-dimension example of fitness function in Figure 1. Suppose, A1-A5 is our random generated start chromosomes. As we can see, fitness value of A2, A3 and A4 chromosome are bigger then fitness value of A1 and A5 chromosome. It means that most likely A2, A3 and A4 will be selected for crossing. Eventually, our child chromosome will be close to 0.5 value. But as we can observe it is just local maximum of function. Mutation can help to avoid this problem, but as smaller radius of "pit" of this function, as difficult it will be to get there. At first it seems that such functions with fast-growing local extremum value are not popular in real task. But indeed they are pretty natural. For instance, we can meet some in semantic analysis of the text.

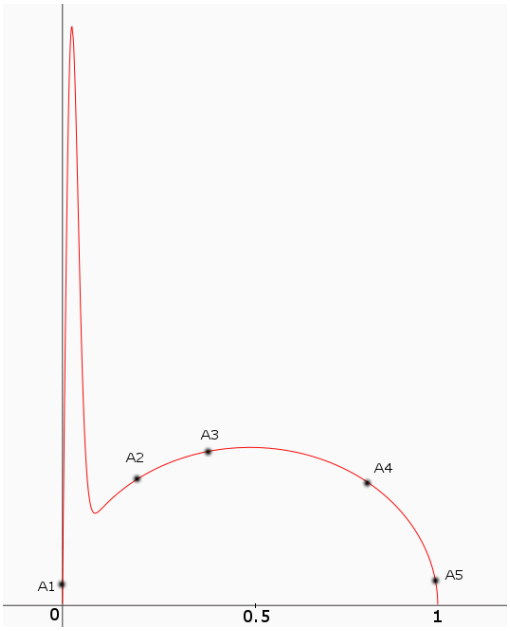
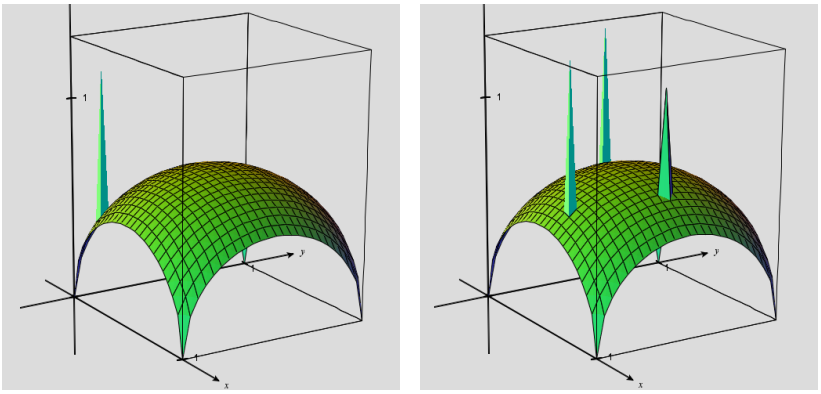


Figure 1

Proposed Solution

The proposed selection schemes are performed in the environment where all child individuals are divided on two groups. First group will be generated by Roulette Wheel method and second by Genetic algorithm with chromosome differentiation (GACD) [2]. The main idea of GACD cames from biological module where individuals looking for the most most dissimilar partner. It lead to absolutely new offspring that may be healthier. In our case we will calculate difference between chromosomes using Hamming distance.



(a) n=1 **(b)** n=3

Figure 2. Experimental function

Let's k is percent of population that will be generated using GACD and $100 - k$ percent of Roulette Wheel based generated population. Often we can't classify our function and set constant k value for all functions. That is why we should apply dynamic approach in this schemes. In this case k value will increase if GACD based population gave good result in next population and decrease otherwise. In addition we should set maximum and minimum value of k. We set $5 < k < 15$ in our experiment.

Experiment

For our experiment we wrote Java program. Three-dimensional space were chosen for fitness function.

First step was to create class of fitness functions that has some fast-growing local extremum value.

$$F(x,y) = \sqrt{\left|\left(x-\frac{1}{2}\right)^2 + \left(y-\frac{1}{2}\right)^2 - \frac{1}{2}\right|} + \sum_{k=1}^{n} e^{-\left((x-x_k)^2+(y-y_k)^2\right)*10000} \quad (1)$$

, where $x, y \in [0,1], n \in N$ and $x_k, y_k \in [0,1]$ is the point where functions has fast-growing extremum value.

For instance, in Figure 2 (a) $n = 1, x_1 = 0.1, y_1 = 0.1$ and in Figure 2 (b) $n = 3, x_1 = 0.1, y_1 = 0.6, x_2 = 0.6, y_2 = 0.1$ and $x_3 = 0.8, y_3 = 0.5$.

Then we estimated number of generation that is necessary to get close to maximum of function for both hybrid and clear Roulette Wheel methods. Mostly, hybrid approach gave better results.

Conclusion

Combining Roulette wheel and GACD selection methods can optimize maximum searching Genetic Algorithm. Our local goal is create algorithm that can classify fitness function and adjust hybrid proportion automatically. Then we would like to create an open source Java framework for Genetic Algorithm with possibility to turn on our hybrid feature. In addition everyone will have the opportunity to adjust or change GA selection, crossover or mutation approach according to task requirement.

References

[1] David E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1989

[2] R.Sivaraj, Dr.T.Ravichandran, A review of selection methods in Genetic Algorithm, (IJEST), Vol. 3 No. 5 May 2011.
[3] David E. Goldberg, Kalyanmoy Deb, A Comparative Analysis of Selection Schemes Used in Genetic Algorithms, Morgan Kaufmann Publishers, Inc, 1991.

Authors

Dmytro Oleksandrovych Okhonko — the 4nd year bachelor, the faculty of mechanics and mathematics, Taras Shevchenko national university of Kiev, Kiev, Ukraine; E-mail: *diametralis@gmail.com*