Evolutionary Stategy using Genetic Algorithms

Brent Verpaalen s2158124 b.a.a.verpaalen@umail.leidenuniv.nl
Athanasios Agrafiotis s2029413 a.agrafiotis@umail.leidenuniv.nl

December 15, 2018

1 Introduction

The assignment concept is to implement an evolutionary strategy. An evolutionary strategy is an optimization population based algorithm. The main purpose of evolutionary algorithm is produce good solutions solving mathematical function such as black box functions. The methods of evolutionary algorithms can vary according to the most important operators such as recombination-crossover and mutation operators.

2 Problem Description

A black box function(BBFOP) presents ambiguity between inputs and outputs. The goal is to find the global or local optima a providing good solution. Black box functions is hard to be solved because the relationships of input and output are indistinct. The goal of optimization is to provide solution for the value f(x). For this research 5 different black box function was used but as the name depicts is not to our knowledge what those function contains.

3 Evolutionary Algorithms

The idead of evolutions is based on the Darwin theory that it is randomly generated population depicts that search process which create parents and select the best children. The population evolved over generations. An evolutionary algorithm is metaheuristic algorithm which can provide good solution for an optimization problem. A metaheuristic is a procedure that tries to find good solutions for a problem. The base functions that an evolutionary algorithm uses is **selection**,**crossover**,**mutation**.

4 Selection

The process of selection is based on the fitness function where the fitness function evaluates each of the individuals and assign a score. Next we normalize and sort the score of the fitness function in order to find the individuals with the highest score to select them as Parents. The number of parents is two and represent the best individuals of the whole population,and the process of selection ends.

4.1 Selection

The selection which was used for this experiment is based on the **tournament selection**. This selection method creates random brackets where different individuals compete. Each round/iteration individuals are pared in groups of two. To proceed to the next round a individual needs to have a more optimal score than the other individual in his group. We keep creating rounds until we terminated enough offspring to become parents.

5 Crossover- Recombination

The process of crossover is very important and the main function is that transfers a number of genes which belong to Parent1 to Parent2 to create an offspring. The implementation of the crossover is depend on the probability which is generated as threshold where the upperbound is length of parent genes and the lowerbound is the minimum number of genes.

5.1 Uniform Crossover

According to a probability of the threshold number of the genes will transfer to create the child. Specifically on this assignment uniform crossover was used the new child containing genes of parents. In uniform crossover each gene of the offspring is chosen from the distribution. Uniform crossover exchanges genes and not segments of the genome. This methods purpose is to choose genes from each parent with equal probability at random.

Figure 1: Crossover

6 Mutation

The next phase is the process of mutation where a number of genes in the child will change. Where we will change genes based on step sizes. Each step size is created by taking a random number between the upper and lower bound and multiplying it with the mutation rate. In more details a step size is created which was multiplied by the mutation rate, this gives us an option to make smaller or bigger steps. Small mutation rate has as result smaller mutation step sizes which can be more accurate but take more time to progress. Bigger step sizes take less time but often overshoot which creates worse accuracy. Different mutation rates have been used in order to evaluate the performance. The points of mutations which has been used is [0.001,0.01,0.05,0.1] as a heuristic search. After each mutation we mutate each step size using the third formula below.
$\vec{a} = ((x_1, ..., x_n), (\sigma_1, ..., \sigma_n))$ individual before mutation
$\vec{a'} = ((x'_1, ..., x'_n), (\sigma'_1, ..., \sigma'_n))$individual after mutation
$\vec{\sigma'} = \sigma_i * exp(t' * N(0,1) + t * N_i(0,1))$ mutation of individual step size
$x'_i = x_i + \sigma'_i * N_i(0,1)$ mutation of individual gene
The probability of crossover represents a global search and known as **exploration** and the probability of mutation represent a local search and known as **exploitation** while they try to exploit solutions.

7 Selection best individuals

The next phase is to evaluate the new population according to the old population and to select the best parents. The way that operator works is to combine the parents in this implementation the number of parents was 2 the combination of the parents returns the children . Because of the previous phases the new population can be almost identical or better than the old population. In order to evaluate which children/offspring will be selected to the new population it evaluated by the objective function. In this research the number of selection of the best offspring was **50%**.

```
Result: Write here the result
initialization;
while iteration ¡= maxiteration do
    iteration = iteration +1;
    calculate the fitness of each individual;
    select the individuals according to their fitness;
    perform crossover with probability p_c;
    perform mutation with probability p_m;
    population = selected individuals after crossover and mutation;
end
```

Algorithm 1: Evolutionary Algorithm

8 Research

The research was conducted using a evolution Strategy (ES) the evaluation budget was in 10,000 function evaluation per problem. The fitness functions which have been used were the black box functions and the number of parents were 10, and 20 offspring.The mutation rate was 0.005 which represent the exploitation and the crossover probability was about 0.5 as you can observe in table 8 the values of the average and std dev was used to evaluate our fitness function. The second approach was used using mutation rate 0.01 and cross over probability 0.5. In figure 2,3,4,5,6 the evaluation the curve display the fitness function which depicts an evaluation of the fitness function for 10000 iterations adding the selection function function with 50% selecting the best individuals to create new population.

	ES configuration A 0.005		ES configuration B 0.01	
Benchmark	Avg	Std Dev	Avg	Std Dev
1 bff1	59,61295	4,55407	132,5955	21,46523
2 bff2	434,6525	147,039	575,4835	75,71118
3 bff3	274,0535	152,5399	1684,722	764,9194
4 bff4	45,00735	0,126976	45,04595	0,014688
5 bff5	68,52115	0,091663	68.5498	0.093521

Table 1: Fitness values over 20 runs for mutation rate 0.005 and 0.01

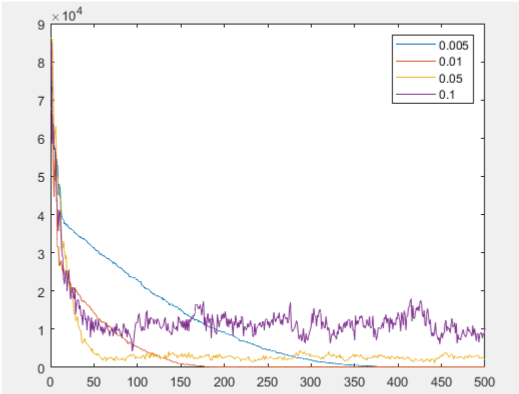

Figure 2: bff1 evaluation fitness function

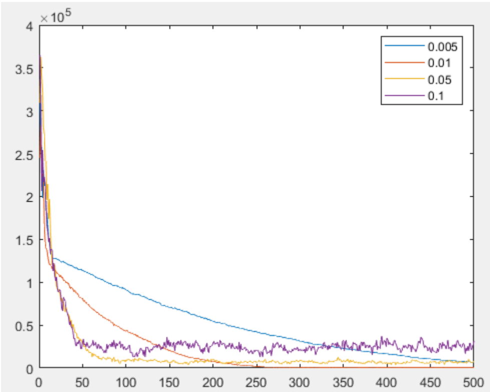

Figure 3: bff2 evaluation fitness function

9 Discussion and Conclusion

Through the research an evolutionary algorithm implemented. Five different black box functions were used. A heuristic implementation was used to evaluate different mutation rate for each of the functions. For the figure 2, 3, 4, 5 the results depict that the best mutation rate was 0.01 in terms of speed with minimal fitness loss. The last function in figure 6 shows that the mutation rate of 0.005 seems be more close to the optimum solution compared with the other functions. Which indicates that lower mutation rate take longer but also create better fitness in the long run. In the table 8 we can observe that the mutation rate of 0.005 gives the best solutions compared with mutation rate of 0.01.

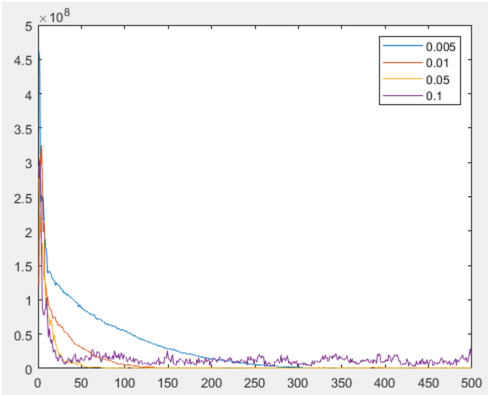

Figure 4: bff3 evaluation fitness function

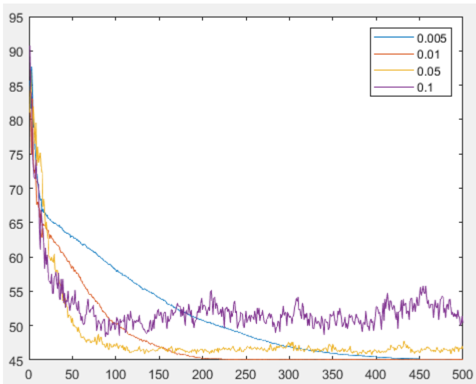

Figure 5: bff4 evaluation fitness function

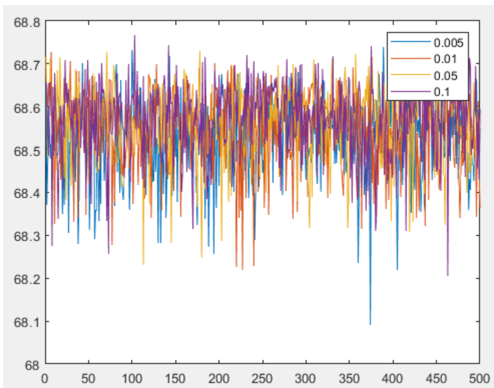

Figure 6: bff5 evaluation fitness function