Evolutionary Algorithms: Peer review report

For the group of Andru Onciul, Alexios Konstantopoulos, and Alexandros Kyrloglou

Carpentier Louis, Derwae Helena, Goris Michael, and Vlekken Daan

November 17, 2021

1 One strong aspect

We were really impressed with the stopping criteria of the algorithm. At first it seemed really advanced to apply linear regression to the mean fitness of the past few generations, but once you think about it, it really makes sense and seems logical. The criteria itself seemed really well worked out. You have really thought it trough (eg thinking about normalization of the slope). At some point we doubted whether applying the linear regression model wouldn't cause any computational overhead, but when we saw that "only" 5 generations were taken into account, the operation is likely very fast.

So to summarize: the stopping criteria is advanced and offers a really good intuition about when the population converges, without giving a major computational overhead.

2 Three weak aspects

2.1 k-values in k-tournament are quite low

The k-values for the k-tournament in both the elimination and the selection are pretty low. A k of 2 for selection would mean you might have too much exploration and not enough exploitation. A k-value of 3 for elimination might be good, but 2 for selection does seem too low and might have to be increased. To fact that you would go to a local optimum too fast can be changed in other ways, for example by increasing randomness in the recombination or mutation step of the algorithm.

2.2 offspring creation

We didn't really understand how the offspring is created. The report mentioned that only the current population is stored. Does that mean that whenever crossover occurs, the parents are replaced by the offspring? If this is the case, we think it might be useful to keep both the parents and the offspring to do elimination on.

2.3 Population size

Currently the population size is big compared to the number of cities, which will work for a small number of cities. However we believe when scaling up to a dataset of a bigger size, increasing the population in conformity will lead to heavy load on memory.

If you are not planning on upscaling the population size conform the number of cities, the performance of the algorithm will change.

3 One suggestion

It might be useful to implement a way to get out of a local optimum by introducing more randomness. If you would do this you could increase your k-value in the selection and more exploitation could be introduced.