SIG Proceedings Paper in LaTeX Format

40165151

ABSTRACT

Abstract goes here

ACM Reference format:
40165151. 2018. SIG Proceedings Paper in LaTeX Format. In *Proceedings of Coursework, Edinburgh Napier University, April 2018 (SET10107),* 2 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The objective of this report is to implement and evaluate an evolutionary algorithm that is used to evolve the weights of a neural network controller. The neural network in question is attempting to land a spacecraft using as little fuel as possible. This is to be done by manipulating the way in which new members are created and added to the population as well as the chance and magnitude of any mutations that occur.

1.1 Background

For the following work a description of some background terms is required, namely the operators that were utilised to achieve the result.

1.1.1 Individuals. An individual is any specific genome that is used operated upon by the algorithm. The genome in this algorithm is a permutation of the weights used by the neural network. For this implementation an individual also possesses a fitness value. This is a measure of how well the genome performs at its given task. The lower the fitness is, the better the individual performs.

1.1.2 Population. The population within the algorithm is the collection of individuals that are being operated upon. Individuals within the population can reproduce to create children which may then be introduced into the population. Each time the algorithm completes an evaluation the population should have been changed, and each new population is called a generation. The goal of the algorithm as a whole is to improve the overall fitness of each subsequent generation.

1.1.3 Selection. The selection operator in the context of an evolutionary algorithm is used to select viable parents for the purpose of reproduction. This is generally done in order to ensure children are more likely to be produced by the individuals with better fitness, though they can also be chosen at random.

SET10107, Edinburgh Napier University

DOI: 10.1145/nnnnnnn.nnnnnnn

1.1.4 Crossover. Crossover is the way in which genetic information from parents is combined to create a new individual. This involves taking parts of each parents genome and combining them into a single viable genome.

1.1.5 Mutation. Mutation is used to add genetic diversity to an existing population. This is done by slightly adding to or removing from a a gene. The chance for a gene to be modified is the mutation rate, and the amount the gene can be modified by is the mutation change. The mutation change is generally a small amount so that the mutation is less likely to move beyond the optimal solution.

1.1.6 Replacement. Replacement is the way that new individuals are added into the population. This can be done by evaluating the fitness of new individuals in comparison to those already within the population or by randomly replacing old members of the population.

1.1.7 Overfitting. Overfitting occurs when a neural network is trained too much on a training set. This is where the network adapts to the solving the particular training set more so than to the problem it was created for. The result of this is excellent performance for the training set but poor performance for any testing sets.

1.1.8 Convergence. Convergence is where there is no diversity within the population, meaning each solution is identical. This can occur at a point before the optimal solution is found, causing a local optimum. At this point a better solution cannot be easily found due to the lack of genetic diversity, and only a lucky mutation could potentially improve the solution.

2 METHODOLOGY

A number of methods were used for each operator in the evolutionary algorithm. This was done so that a comparison could be made between each of these methods to compare how suitable each is for this particular problem. The operators in question are the following: select, crossover, mutation and replace.

2.1 Selection

The selection operator as mentioned in the background is used to choose viable parents for reproduction. For this implementation two different methods were used, random selection and tournament selection. Random selection simply selects a random member of the population for each parent. Because this is completely random it is possible that parents with poor fitness that would otherwise be ignored could be selected and add genetic diversity that could improve the overall population. If left unchecked however the population could become less fit as easily as it could become more fit. Tournament selection involves taking N members of a the population and selecting the best of these members as a parent. This prevents the worst of the population from being selected, and the higher the tournament size the better the chance that the best members of the population are selected. The downside of this method is that if the best members of a population breed more often than the others, a premature convergence is more likely to occur. Because of this the size tournament should be limited so that the population is encouraged to improve, but stagnation doesn't occur.

2.2 Crossover

For the crossover operator a number of methods were used, uniform, one point, and N point crossover. In uniform crossover each gene in the genome has a chance to be from either parent. The chance for a gene to be from either parent is equal in this implementation, as fitness is not taken into account. One point crossover involves choosing a random point in the genome, splitting the parents about this point and selecting a part from each parent. In this way a child will contain a significant portion of at least one parents genome. Lastly N point crossover occurs in a similar manner to one point, though instead of a single splitting point, there are N splits. A segment is taken from each parent in an alternating fashion. While one point and N point are more likely to keep traits from a parent, and potentially take the good qualities of each, uniform crossover is more likely to introduce genetic diversity. In addition N point will produce more diversity than one point on average.

2.3 Mutation

For this implementation only a single method of mutation was used. This involves giving each gene in the genome an equal chance to mutate by a static amount. The mutation change that occurs can be positive or negative. This produces a reliable element of randomness to the population, and should help to prevent premature convergence from occurring throughout. The impact of the mutation rate will be evaluated to see how much mutation can occur while maintaining a fit population.

2.4 Replace

Several methods of replacement were used. The first of these was random replacement. This involved selecting a random member of the existing population and replacing it with a random child. This occurs a number of times in order to add diversity to the population. The second method used was tournament replacement. This collected the existing population and the children into a single group, then performed a tournament on this group to add each member to the population. This was done enough times to match the size of the original population. This should select the better members of the combined group and should produce a population with a good fitness. This faces the same problems as tournament selection, where a large tournament size could lead to premature convergence. The last method used was replace worst. For each child the worst existing member of the population was removed and was replaced with the child. This could improve the fitness of the algorithm by removing those with poor fitness, but could also make it worse by introducing children with worse fitness.

3 RESULTS

4 CONCLUSION

https://www.whitman.edu/Documents/Academics/Mathematics/2014/carrjk.pdf