Evolution of Complexity – Assignment II

An Analysis of the Benefit of Crossover when Using Building-Blocks

1 Introduction

Within the field of Genetic algorithms it has been widely theorized that using crossover to solve certain fitness landscapes should be more efficient than simple hill-climbers [1]. In the past much time has been spent trying to establish which landscapes will optimize the use of crossover, specifically to overemphasize the ability for building-blocks to flourish [2]. This report is based on the paper written by Richard A. Watson [3], which aims to address the building-block problem by using a two-module approach to idealize the use of building-blocks in conjunction with crossover. What follows is a reimplementation of Richard A. Watsons study, followed by a critical analysis of the effect the number of crossover points on the effectiveness of the algorithm.

1.1 Summary of Paper

The proposed solution to demonstrate the merits of building-blocks revolves around the idea of crossing over high fitness schemata to find the optimum fitness. In order to present this idea, a landscape has been defined such that there is a strong fitness correlation within the building-blocks to encourage development towards them [3]. For this approach to work, a high diversity has to be maintained in order to guarantee that all building-blocks are found using hill climbing before they are crossed over with other building blocks [4]. To ensure that all the building-blocks are present a multi-deme island model is employed to allow various populations to evolve to optimize for different building-blocks. To optimize the use of building blocks, a fitness landscape was designed that once one block was found, there would be various trap functions (generated by noise) to prevent the hill climber form evolving to the maximum fitness, which could only ever be achieved using crossover [3].

2 Reimplementation of Original Paper

2.1 Conceptual Method of Reimplementation

The reimplementation of the two-module problem was completed using a fairly similar approach of the original paper. The genome was an *n* long vector, which is mapped onto a fitness landscape defined as follows:

$$f(G) = N_{(i,j)}(2^i + 2^j)$$

Equation 1 - Function defining fitness landscape where N is noise and i and j gene building blocks.

Where *N* is a random number ranging between [0.5, 1], and *i* and *j* are the fitness of the first and second half of the genome respectively calculated by the number of 1s by using 0 and 1 as the genotype. As seen in Figure 1, for gene one *i* is 2 and *j* is 3, this will then be mapped to the fitness landscape to provide a fitness value. The purpose of splitting the genome into two halves and mapping it onto this specific landscape is to produce a stronger synergy between the same genes rather than different ones promoting development of building-blocks while still being epistatically linked.

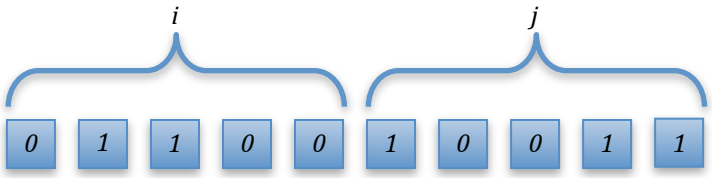

Figure 1 - Example of genome of length 10 subdivided into two genes i and j, the fitness is determined by mapping the number of ones from i and j onto a fitness landscape defined in Equation 1.

The fitness landscape is produced using Equation 1, as the equation dictates the landscape involves creating a noise landscape which is then multiplied by the main landscape to provide the final fitness landscape, as seen in Figure 2. It is important to note that the noise is added in order to prevent the hill climber from simply walking along the smooth edge of the main fitness landscape and thus finding the optimum solution in linear time.

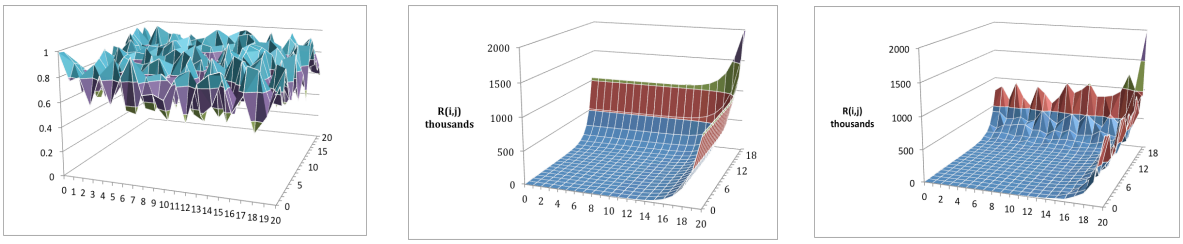

Figure 2 - Displays of construction of fitness landscape. (left) Noise generated between 0.5 and 1, (center) fitness landscape produced by 2^i + 2^j, (right) product of equations one, (noise multiplied by main fitness landscape).

2.2 Technical Method of Implementation

In order to construct the simulation a low level class for chromosomes was designed which held the relevant data such as an array of alleles and functions for mutation and crossover. Using this as a basis the simulation was constructed using a hierarchical approach so that the simulation consisted out of a population, which contained individuals, which had the chromosome properties. Each individual was initialized to have all zeros and which represents the lowest fitness. During mutation an elitism of 1% was decided in order to ensure that deleterious mutations would not be made to the most fit individuals.

In order to implement crossover a generational approach was taken where each population would undergo a crossover phase in which a new generation was born. The reason for this was that in order to debug the system it was easier to observe what changed in each generation rather than over-riding the data in each generation as done in steady state. To select which individual's genes were to be passed onto the next generations by crossover, fitness proportionate selection in the form of a roulette wheel was used. This was to ensure that the fittest individuals had the most chance of reproducing. Migration of individuals was preformed by randomly picking demes that would then exchange individuals with each other to maintain genetic diversity.

In order to ensure the maximum fitness was obtained, by being in the far corner of the fitness landscape, noise was not applied to the largest value of the plain field (*2^imax + 2 ^jmax)*.

The simulation parameters are specified below:

Number of Demes	50
Number of individuals in population	200
Number of Alleles	40
Migration Probability	0.005
Mutation Rate	0.025
Crossover Rate	0.025

Table 1 - Table showing build parameters

2.3 Results

As mentioned previously the simulation consists out of 50 demes each having 200 individuals, with a migration probability to ensure the diversity within the demes are maintained. Each simulation was run 10 times and an average taken in order to get a fair result. The findings are presented in Figure 3 (And Figure 8 in the Appendix using a logarithmic scale). We can clearly see that the time for a one-point crossover technique to reach the peak

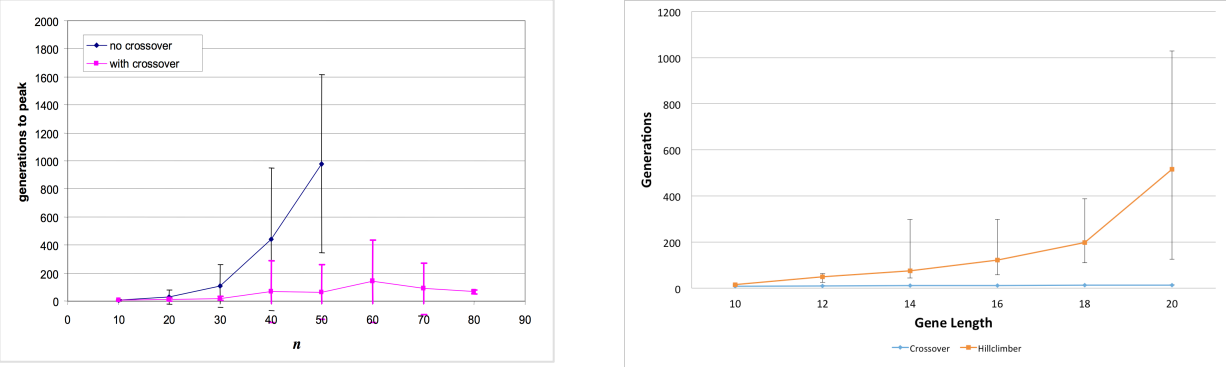

Figure 3 - Graphs comparing data found by original paper and those obtained by simulation, (left) original (right) simulated. Both graphs show the time till optimal fitness has been reached.

consistently relatively few generations when increasing the size of the gene. Where as, when compared with using just hill climbing and no crossover of any sort, the time to reach maximum fitness exponentially increases. Intuitively this is would seem logical, as a one-point crossover has the advantage of being able to use building-blocks from other demes. In contrast with hill climbers which gets caught in a local optima.

2.3.1 Critique

While the simulation clearly shows the advantage of using a one-point crossover when competing against a hill climber technique, the results do not reflect the results obtained by the original paper. When investigating the ability of the hill climber it failed to solve the problem in a reasonable number of generations (less than 2,000) around a gene size of over 20 – 30 giving inconstant results. The obtained results showed that the hill climber was able to solve the problem in exponential time. This is to be expected when using hill climbers; eventually some fortunate combination of mutation loci will allow it to overcome the local optima however, this would be achieved in exponential time. While in theory this is consistent with the original paper, the rate at which crossover outperforms hill climbers is significantly faster than that stated by Richard A. Watson. A possible explanation for this could be due to an increased mutation rate that would allow the genes to mutate more often and thus are more likely to overcome the local optima.

When experimenting with decreasing the size of populations, to understand the effect of reducing the internal diversity of a deme, it was quite clear that a reduction in population size increased the number of generations it took to find the global optimum. This supports the theory that diversity has to be maintained within the simulation in order to effectively reach the global optimum as it takes longer for demes to reach the local optima (i.e. top fitness in a gene) and thus longer to create the essential building blocks.

When implementing uniform crossover, it was apparent this also solved the problem in exponential time when increasing the genome length. This is because building-blocks break down when recombining with other genomes as each allele is chosen from a random parent. However, again, the efficiency at which the uniform crossovers find the optimal solution is much faster than stated in the original paper.

One problem I encountered implementing the simulation was how random numbers were actually generated. It is crucial to ensure truly random numbers are generated, specifically when dealing with uniform crossover. When initially implementing the simulation, the random number generator wasn't truly random and thus made

uniform crossover more into n-point crossovers due to the fact that it was choosing one parent more often than the another.

3 Extended Research

The two-module problem discussed in the paper written by Richard A. Watson (original paper) is aimed at showing how crossover can be beneficial when solving specific landscapes, which favor high fitness schemata. However, it does not analyze the effects of varying the crossover function to multiple points. The specific issue that will be addressed in this research is the effect of adding more crossover points when computing the next generation and observing the effects this has on the effectiveness of finding the peak fitness.

3.1 Hypothesis

The success of building-blocks appear to rely on being able to pass sets of strong genes to the next generation which can then be combine with complimentary sets to preform better. When having two different genes dictating the fitness of a genome, as presented in the original paper, intuitively, the optimum crossover would be at loci, exactly where the genes are separated. This is because it would allow the building-blocks to perfectly pair up to form a highest fitness genome, which is reflected in the re-implemented simulation. Crossovers where building-blocks are not preserved are not able to solve the problem when exposed to genomes with a high rank. This is reflected in the uniform crossover; it becomes exponentially more difficult to work to a solution as the algorithm intrinsically breaks down building-blocks by randomly selecting genes from both parents.

This trend, of where it becomes exponentially more difficult to solve the two-module problem as seen in the uniform crossover, is expected to be observed when increasing the number of crossover points. This is due to the building-blocks not being given to the chance to form, as recombination in each generation separates them. It is also expected that as the number of crossover points approaches the number of loci in a genome, the amount of time required to solve for global top fitness is going to move towards that of uniform crossover.

3.2 Motivation

At the heart of building-blocks lies the idea of modularity, i.e. building-blocks in the form of modules are passed on to the next generation [4]. Investigating at what point modularity breaks down into randomly selecting genes from two parent genomes gives us insight into to what extent modularity can be used effectively to solve high fitness schematic problems.

3.4 Method of Implementation

In order to perform simulations that require multi-point-crossover, a new function will be added which accepts a parameter that enables the crossover points to be varied. This will allow for the simulation of various crossover types when changing the number of loci present in a genome. Data will be collected as done previously; an average of ten simulations will be made of each specification in order to ensure a fair result.

The method of measuring the effect of changing the number of crossover points will be attained by calculating the change in number of generations to find the optimum solution between different genome lengths.

5 Extended Research Results

The initial research involved varying the number of crossover points over a set number of gene lengths (10 to 40 with increments of 10), in order to get a basis what would happen. The results are displayed in Figure 6 in the appendix, which shows little to no affect when varying the genome length from ten to thirty alleles. However, take a massive leap in time to solve from 30 to 40 generations. This suggests an exponential relationship between the number of crossover points and number of genes present.

Following this, an investigation between thirty and forty genome length was conducted with the purpose of having a closer look at the relationship between crossover and genome length, of which the results are seen in Figure 4. The simulations shows a mixed set of results where the lower crossover numbers fit linear than exponential relationship, however as the number of crossover points are increased a clear exponential curve can be observed. A possible explanation for this might be that with the lower number of crossover points is there isn't enough data to be able to observe and an exponential trend and hence appears linear.

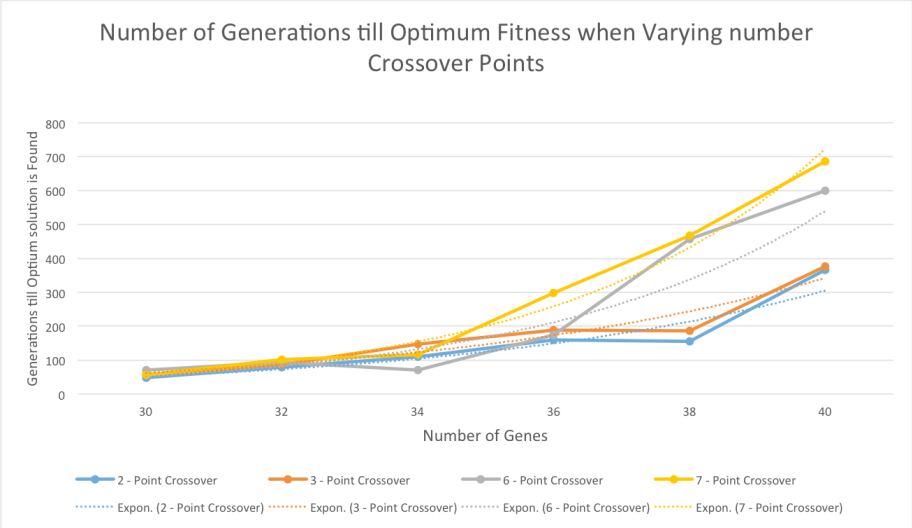

Figure 4 - Graph showing number of generations till optimal fitness has been reached when varying the number of crossover points within the parameters of 30 to 40 loci. Exponential trend lines have been added in to display and highlight the exponential increase in completion time.

Further analyses of the simulation shows that as the number of crossover points increases, the time required for them to find the optimal solution dramatically increases. This phenomenon can be best observed in Figure 5, which shows the time for forty loci to find the optimal solution when varying crossover points. The results attained were mixed and can be best described to have a linear fit. To test this theory of linearity, another simulation was conducted. To see how fast the effect of number of crossover points affected solving time, the rate of change was measured by analyzing the increase of time to finish a simulation from thirty to forty loci (as this is when the most increase can be observed). As can be observed the data is fairly similar, and shows roughly the same trend.

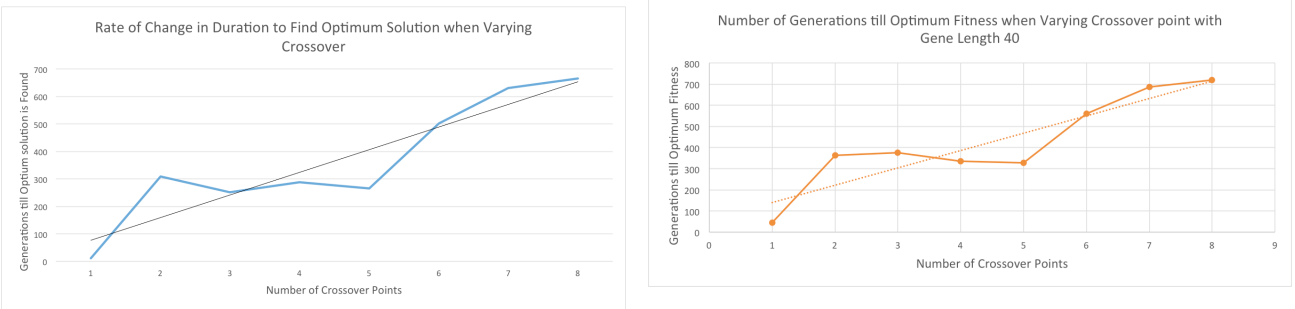

Figure 5 - Graphs showing two different methods of measuring the rate at which increasing the number of crossover points has on the completion time for a genome of length 40. (Left) absolute time till completion (Right) Rate of change from 30 to 40 loci.

6 Conclusion

This research assignment successfully re-implemented the "Simple Two-Module Problem to Exemplify Building-Blocks Assembly Under Crossover". It has reiterated the use of one-point crossover as a tool to solve high schematic fitness landscapes, and more importantly shown that a simple hill-climbing algorithm can be outperformed by crossover. Moreover, an investigation into multi-point crossover has shown a linearly degenerative relationship between the solving time for this fitness landscape and the number of crossover locations.

The results obtained verified the principles argued by Richard A. Watson regarding the advantages of crossover versus hill-climbing. Some differences were seen in the time scale when using a hill-climbers approach and using uniform crossover, however this could be due to varying mutation, migration and crossover rates and would required further research and simulations to confirm the assumption.
It is conclusive that even if the number of crossover points far exceeds the optimal amount it can still outperform a simple hill-climber, which suggests that when operating in a fitness landscape which encourages modularity, crossover is still preferred. Further more, results suggest (with research conducted limited to fitness landscape which optimizes two-modules), straying away from the optimum number of crossover points reduces the efficiency of the algorithm in a linear fashion. This again, supports the need for an effective method of constructing building-blocks.

When comparing multiple crossover points with uniform crossover (see Figure 6), it can be seen that my hypothesis was on the way to being correct. While increasing the number of cr

6.1 Critique and Further Work

While the results obtained reflect the theories set out, it is limited to a genome of forty loci. In order to conclusively say the correlation between the reduction in solving speed and number of crossover points, more simulations need to be conducted where the genome size exceeds forty. This will allow the results to be clearly analyzed and concretely establish the type of relationship (linear or exponential) the increase in crossover points has on the time taken to solve for optimal fitness.

Further research could be made to verify whether it is correct to speculate that by straying away from the optimal crossover point (ie. The numbers of modules minus one, in order to ensure building-blocks can be passed on). This can be investigated by extending simulations into multi – module problems instead of two. This can be achieved by adding another dimension to the fitness landscape; it would be possible to have a three-module problem, which theoretically have an optimal algorithm of using two-point crossover.

Works Cited

[1] J. H. Holland, Adaptation in Natural and Artificial Systems, MIT, 1975.

[2] T. J. D. a. L. T. Richard A. Watson, "A Building-Block Royal Road Where Crossover is Provably Essential," 2007.

[3] X. a. a. e. Richard A. Watson, "A Simple Two-Module Problem to Exemplify Building-Block Assembly Under Crossover," 2004.

[4] A. S. P. R. A. W. Simon T Powers, "The Efficacy of Group Selection is Increased by Coexistence Dynamics within Groups," 2008, pp. 498-505.

[5] R. A. W. Rob Mills, "Variable Discrimination of Crossover Versus Mutation Using Parameterized Modular Structure," 23 May 2007.

[6] G. S. H. a. J. B. P. Richard A. Watson, "Modeling building-block interdependency," in *Parallel Problem Solving for Nature*, Springer Berlin Heidelbreg, 1998, pp. 97 - 106.

Appendix

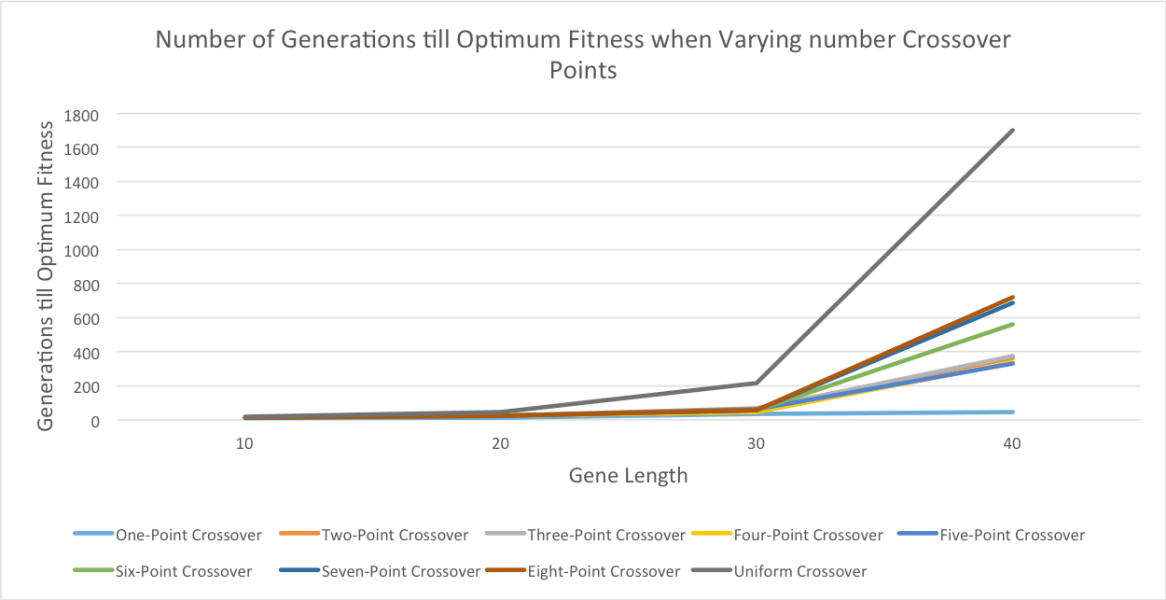

Figure 6 - Graph showing time till optimal fitness has been found for genomes of length 30 to 40 when varying crossover techniques.

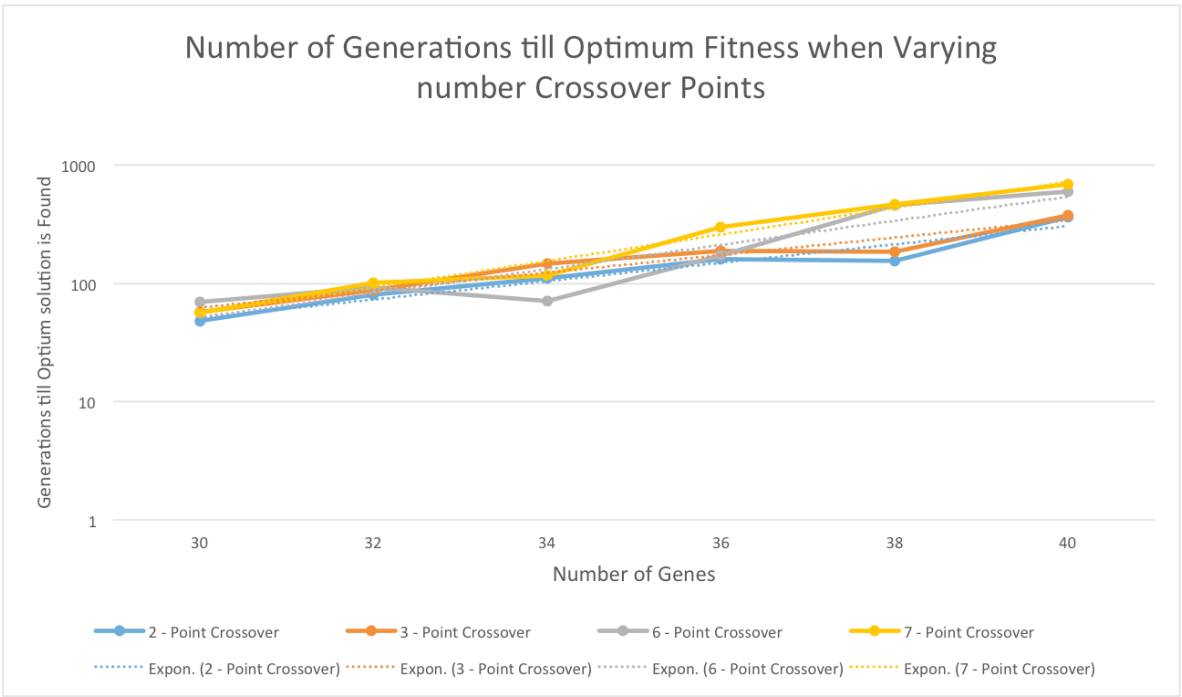

Figure 7 - Graph showing time to reach optimal fitness over from 30 to 40 loci when varying number of crossover points.

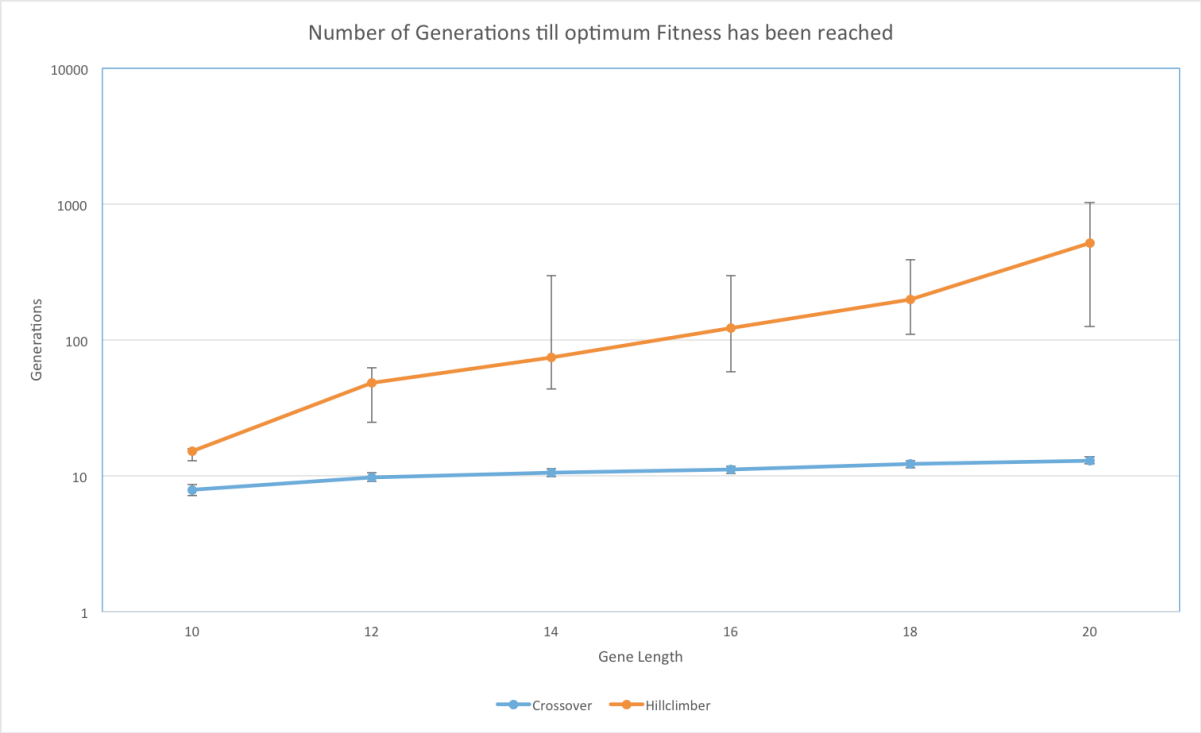

Figure 8 - Logarithmic graph showing time to reach optimum fitness over a length 10 to 20 genome when with two-point crossover and hill-climbing.

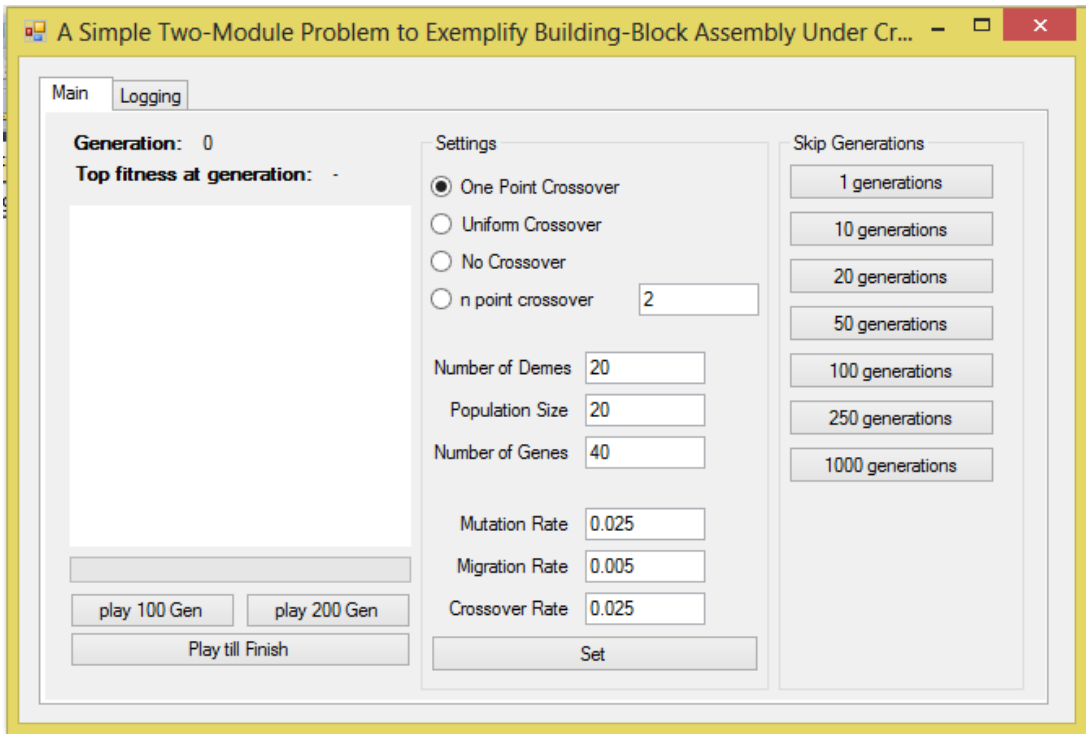

Figure 9 - Screenshot of simulation environment.