

Replication Report

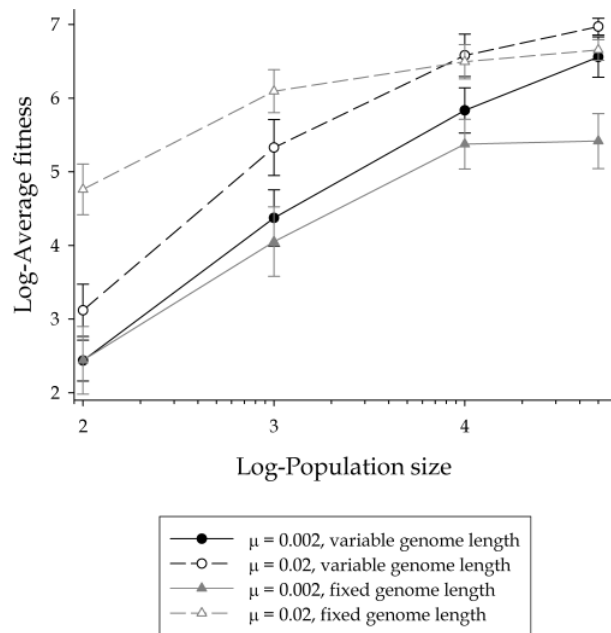
Methodology

Mutational robustness, or the ability of a gene to mutate yet still display the original phenotype, is predicted by Elena et al. to increase as population size increases. To test this, the researchers used Avida, a program that simulates evolution through genetic mutation. Their experiment allowed Avida to run for an unspecified number of updates, then calculated the average fitness across all organisms. Fitness is calculated by “dividing the rate at which an organism acquires the energy used to execute the instructions in its genome by the number of instructions that it must execute to produce an offspring”.

The researchers ran the experiment on 160 populations. They tested the effects of changing population size and the mutation rate. They also tested the effects on variable genome length, but we did not incorporate this into our final project.

Result

The paper confirmed its hypothesis and found that mutational robustness did indeed increase on average as population size increased. The graph showing this is provided below.



Our reimplementaion

For our reimplementaion, we also used Avida to mutate our organisms. We used the library SignalGP-Lite for this. Because the mutation functions are built in, we were able to keep a lot of

things similar to the paper. However, we did change some aspects for simplicity or if the paper did not specify details about their implementation.

We elected to only replicate the parts of the graph that examine a fixed genome length. SGPLite does not yet support insertion/deletion mutations, so it would have been very complicated to create our own genomes to test these on.

Our fitness function was not the same fitness function the paper outlines. The paper describes their fitness function as being “measured by dividing the rate at which an organism acquires the energy used to execute the instructions in its genome by the number of instructions that it must execute to produce an offspring”. We did not understand what was meant by the rate at which an organism acquires energy or how to implement this into our redesign. Our fitness is calculated by counting the number of offspring an organism produces. The more offspring there are, the higher the fitness.

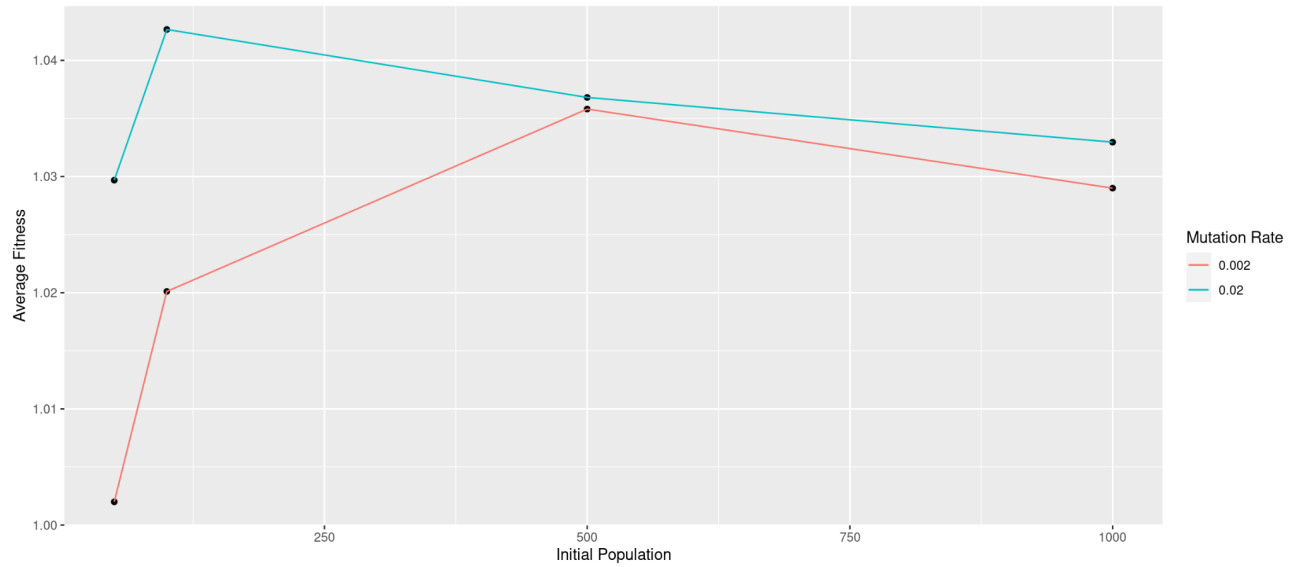
Finally, though the paper does not outline the tasks they have organisms do, it’s probable that our tasks were easier to perform. We had a total of four tasks for our little friends to perform: squaring, cubing, doubling, and halving for 5, 10, 2, and 6 points, respectively. We ran our experiments for four-hundred updates. We suppose that the researchers likely ran theirs longer, but an exact number of updates is not stated.

We tested eight populations rather than the paper’s 160. The eight populations were a mixture of population sizes, different mutation rates, and different amounts of tasks performed.

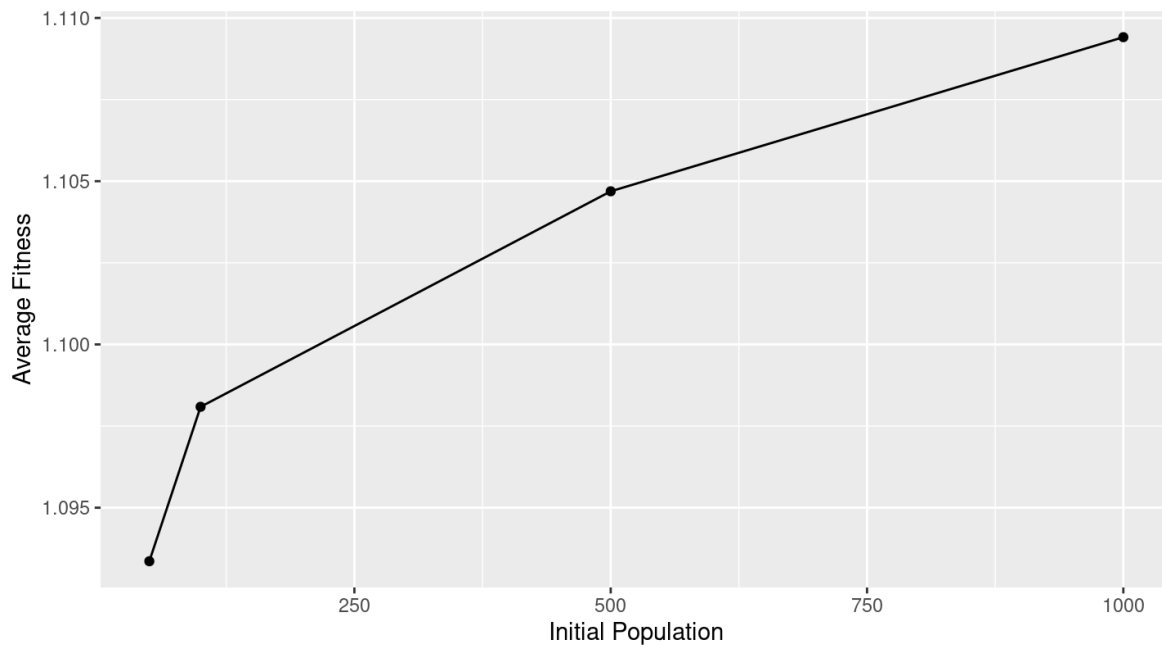
We provide a simple GUI to view the organisms as they replicate. A yellow organism indicates one of fitness greater than 9. There is a configuration panel where the seed, initial population, and mutation rate can be changed.

Our results

Our results somewhat line up with the paper’s findings. We first recorded our results when the organisms were performing just one task and had a mutation rate of 0.02. This resulted in a relatively low average fitness and the fitness actually decreased as the population increased. We also tested a mutation rate of 0.002 to see if there would be a difference. A mutation rate of 0.002 resulted in a lower average fitness compared to a mutation rate of 0.02 but again, the fitness decreased as the population increased. The graph of our findings is below.

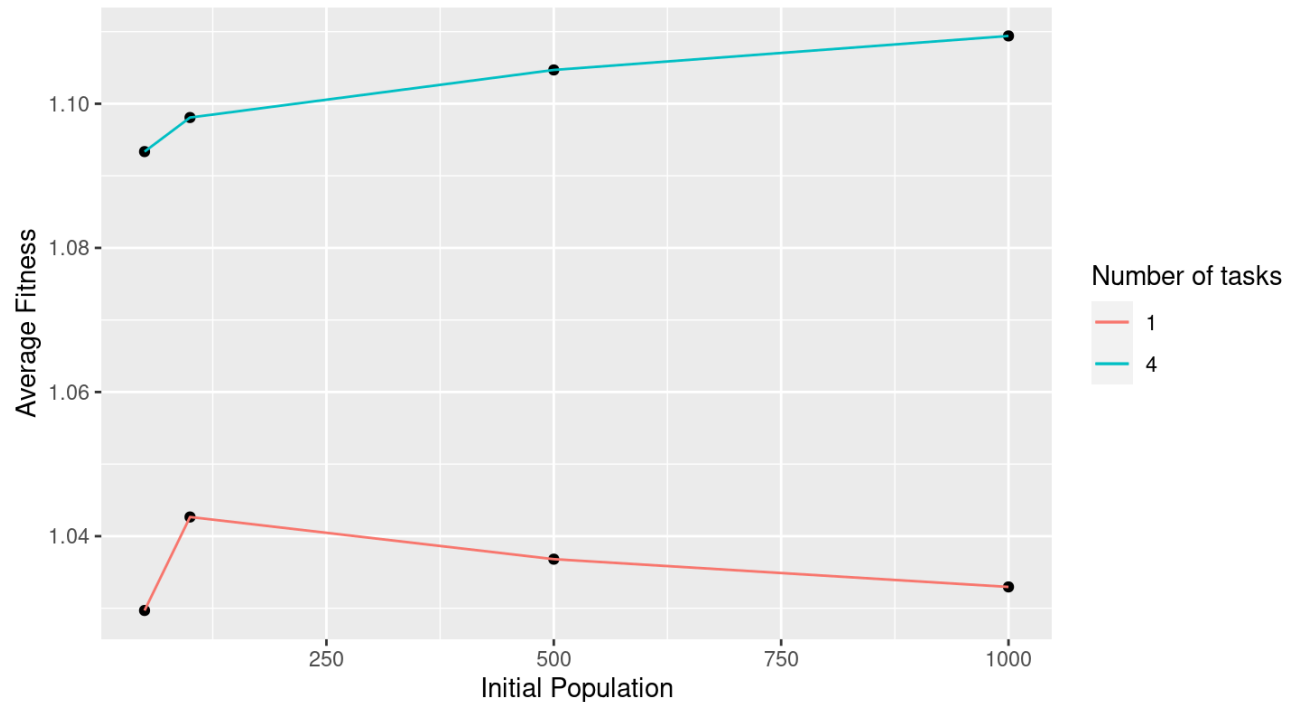


Because these results didn't line up with what the paper got, we decided to increase the number of tasks our organisms performed to see if that matched their results better. The results of that change are in the graph below.



The shape of this graph aligns with the graph from the paper significantly better than with just one task. However, our average fitness is still a good deal less than what the paper found. We will discuss this difference later.

If we focus on the organisms performing four tasks, then we see a correlation between initial population size and average fitness. Like in the research paper, our organisms had higher average fitness (or mutational robustness) as the initial population size increased.



What went wrong?

We had some difficulties comparing our results to the paper's findings. For one, we were confused by the axes provided on the graph we were attempting to replicate. Their plot looks at the log- Average fitness as log-population size changes. We assume that they mean \log_{10} , but it is not explicitly stated. We also couldn't find definitions for acronyms given in their results table so we did not know which of our numbers to compare.

The biggest contributor to differences was likely our fitness function. As we could not implement their fitness function, we think our implementation skewed our results. Our average fitness was close to 1 in all our experiments, where there was a lot more variation in the paper's results.

Our tasks to perform were also likely simpler than those of the paper. As mentioned before, the tasks were not explicitly stated in the paper, but it's safe to assume there were either more or more complicated tasks for their organisms to complete. This may have skewed our results due to it being more likely for our organisms to gain points and reproduce.

Another difference was in the platforms used. We used SGPLite, so there is a slight difference in instructions and task rewards. This may have caused variations in mutation and replication rates, changing our results.

Finally, a potential source of error is the number of updates we ran. As we are college students and time is everything to us, we only ran our populations for 400 updates. Researchers likely have more time than us, and while it's not said how many updates the program was run for, it was likely much higher and took much longer. If their organisms had more updates, it's possible that they would show different results than ours.