

CS 351

Triangle Genome Project

Justin Hall, George Boujaoude

## About

This version of the Triangle Genome Project was written by Justin Hall and George Boujaoude. This data has been compiled to show an overall comparison of how well different thread combinations could produce an image. The Triangle Genome Project (TGP) contains a hierarchy consisting of an overall population. This population is then responsible of differing amount of tribes, in the case of this paper, the smallest amount of tribes consisted of one tribe while the largest amount contained ten tribes. Each tribe then contains two thousand to ten thousand genomes at any time. One genome is capable of closely recreating an image that is fed through the TGP.

## Summary of Testing

In order to test which combination of threads was the most suitable for the TGP project it underwent a series of test. Each thread combination would try to recreate five different images five times each. Our thread combination size consisted of {1, 2, 4, 5, 6, 7, 9, 10}. Once this was completed, the average fitness of those twenty five runs was graphed and compared to the other thread combinations. The fitness is measured from a scale of 0 to 1, where 1 is considered to be identical to the original image.

There were three different computers used to compile the results, however, the method of how we tested over three computers eliminates a thread combination of having an unfair advantage over the others. Computer 1 would do all thread combinations for Images 1 and 2, Computer 2 would do all thread combinations for Images 3 and 5, while Computer Type 3 would perform all thread combinations for Image 4. This is why this method works:

- When compiling the average fitness graph for any thread, we take the average fitness values from Images 1, 2, 3, 4, and 5, and take then the overall average fitness
- If computer 3 were to perform substantially slower than the other computers, that would imply that our values for Image 4 would be much slower than for the other images. This offset would affect any thread combination running Image 4, and because all thread combinations are running Image 4 through the same computer, their overall fitness values should be effected equally.

## Computer Type Specifications

Computer Type 1 – Used to test thread combinations for Images 1 and 2

- Intel® Core™ i7-2600 CPU @ 3.40GHz 3401 Mhz
- 4 Core(s), 8 Logical Processor(s)
- Graphics: AMD Radeon 6450

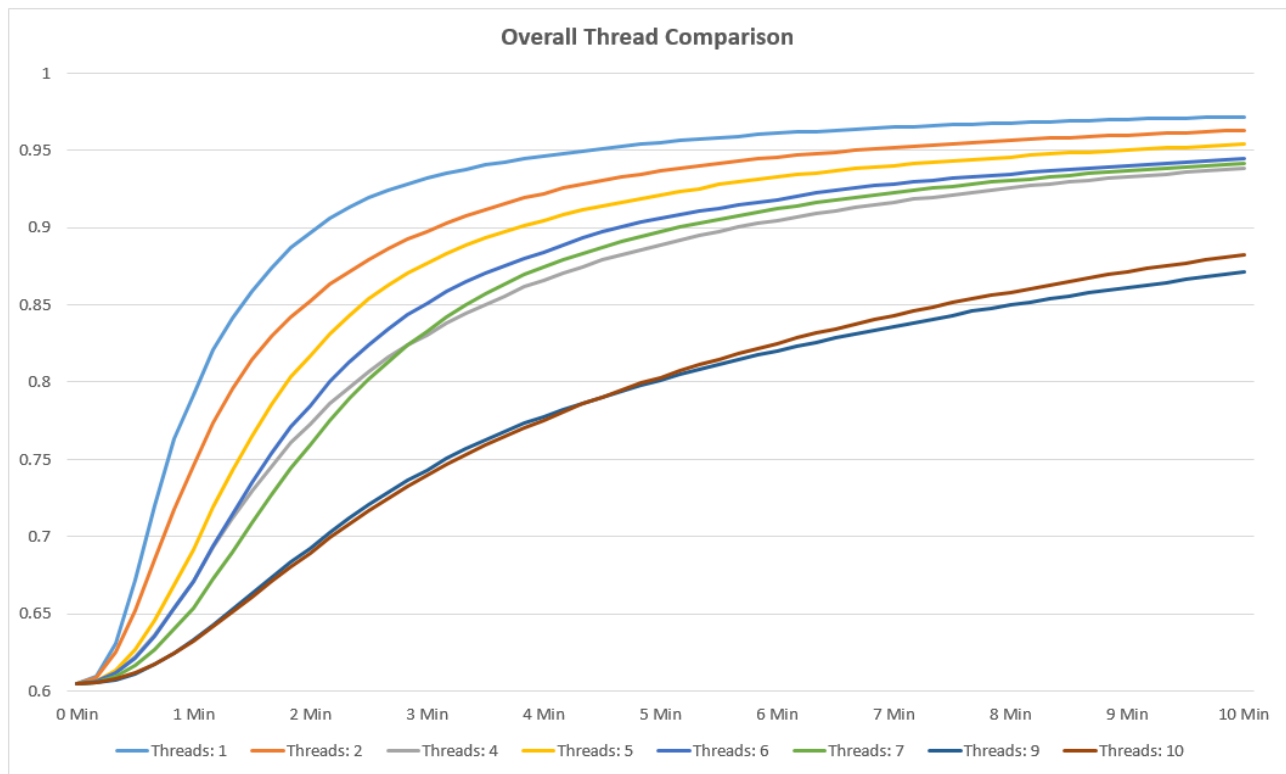
Computer Type 2 – Used for Images 3 and 5

- Intel® Core™ i7-4700MQ CPU @ 2.40Ghz
- 4 Core(s), 8 Logical Processor(s)
- Graphics: Intel® HD Graphics 4600

Computer Type 3 – Used for Images 2

- Intel® Xeon® CPU E31245 @ 3.30 Ghz, 1600 Mhz
- 4 Core(s), 8 Logical Processor(s)
- Graphics: NVIDIA Quadro 2000

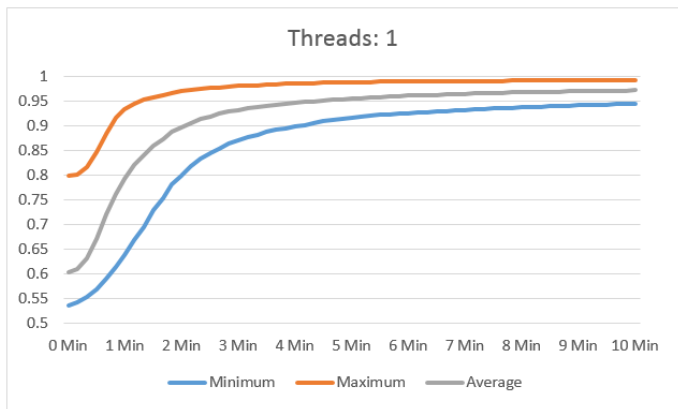
# Overall Thread Comparison: Graph Result



## Conclusion and Breakdown

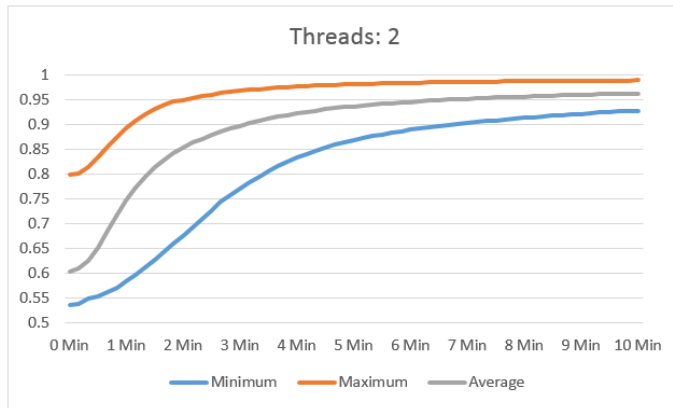
We found that one thread utilizing one tribe had the fastest, and highest quality of results by reaching an average fitness of 0.9717 within ten minutes. Below each thread is graphed individually along with its maximum fitness graph as well as its minimum fitness graph.

### Threads: 1



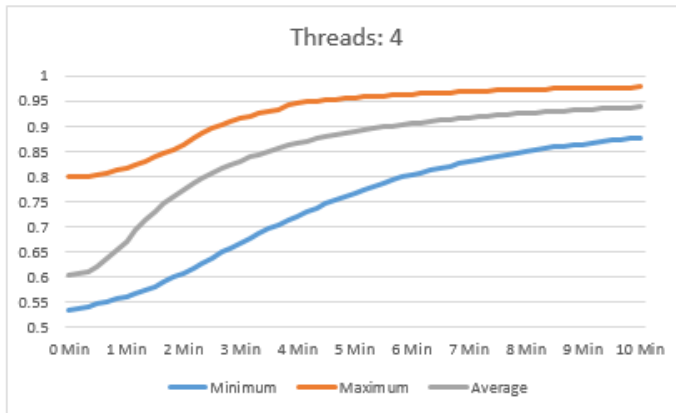
Minimum: 0.9445  
Maximum: 0.9920  
Average: 0.9717

Threads: 2



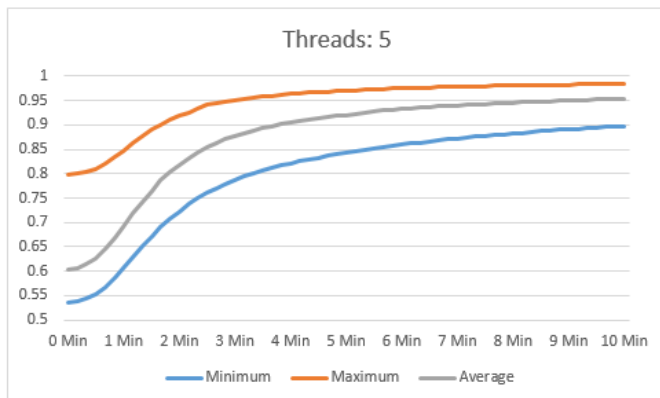
Minimum: 0.9278  
Maximum: 0.9890  
Average: 0.9628

Threads: 4



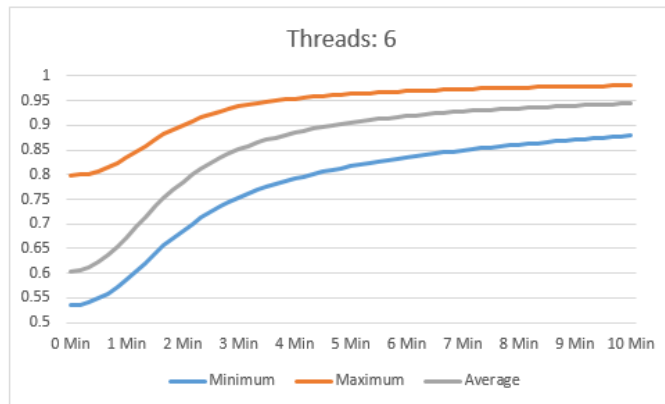
Minimum: 0.8765  
Maximum: 0.9775  
Average: 0.9383

Threads: 5

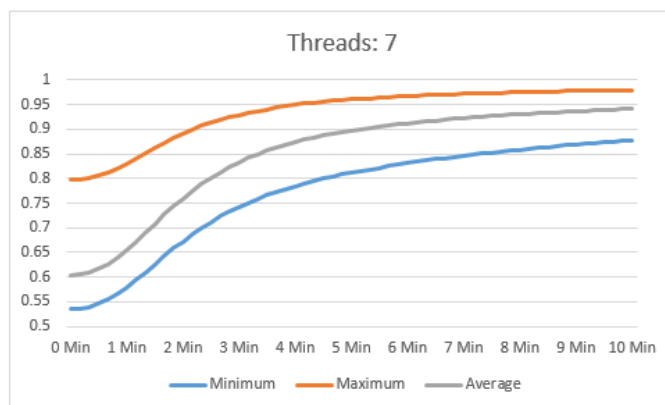


Minimum: 0.8981  
Maximum: 0.9841  
Average: 0.9537

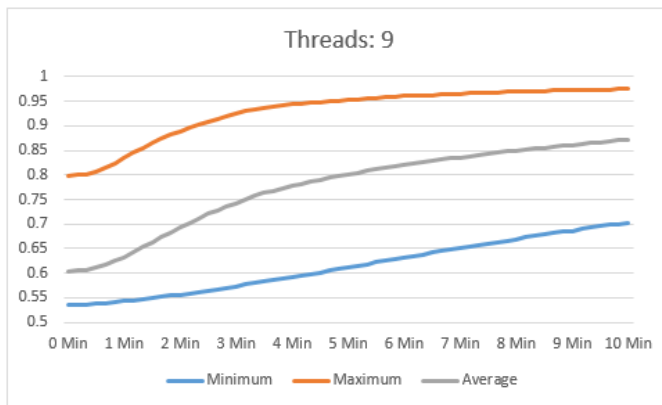
Threads: 6



Threads: 7



Threads: 9

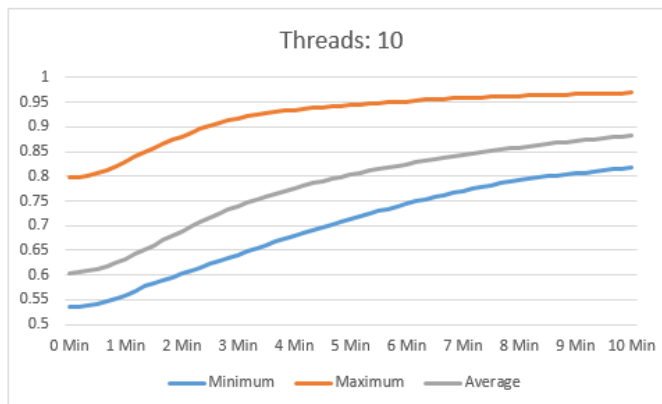


Minimum: 0.7025

Maximum: 0.9751

Average: 0.8714

Threads: 10



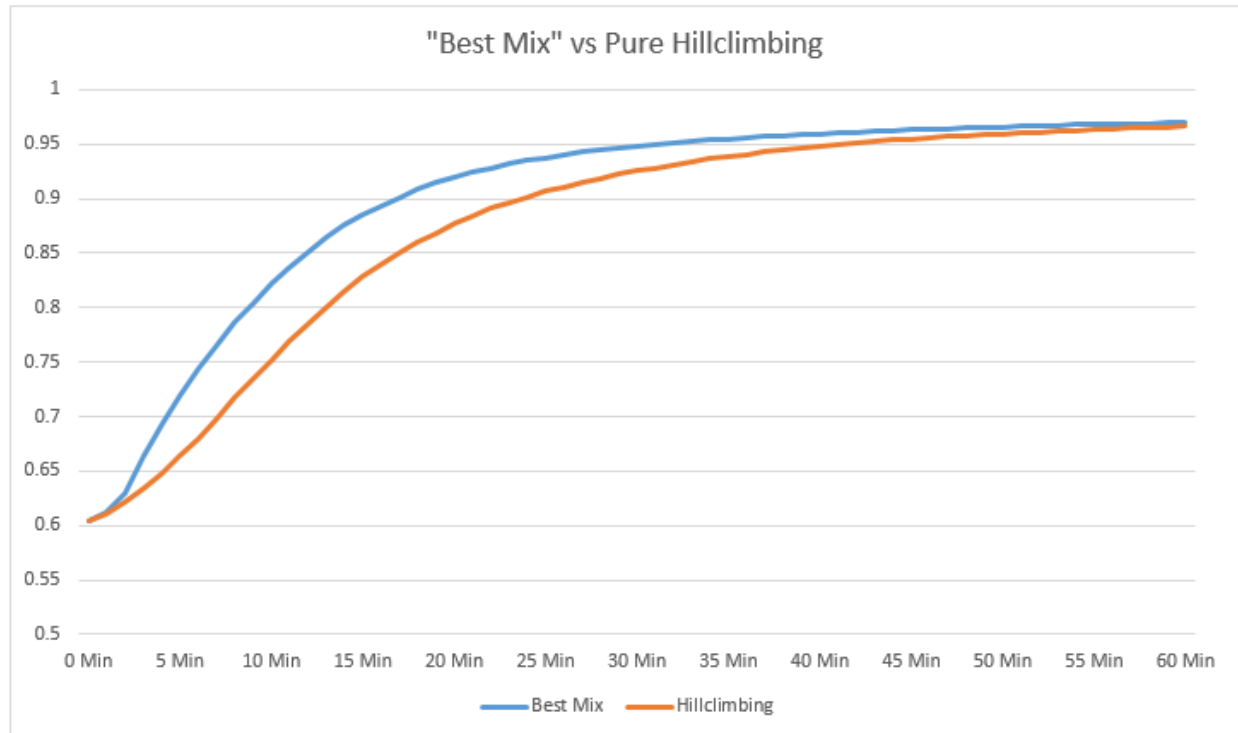
Minimum: 0.8169

Maximum: 0.9689

Average: 0.8823

## Further Analysis:

Upon reviewing the given information, it was found that a thread of one utilizing one tribe was the most efficient result. In, further analysis we selected one thread to run Images 1-5 over the course of one hour utilizing our best mix of crossover and Hill Climbing. We also selected one thread to run the same images over the course of one hour relying only on Hill Climbing. Below are the results:

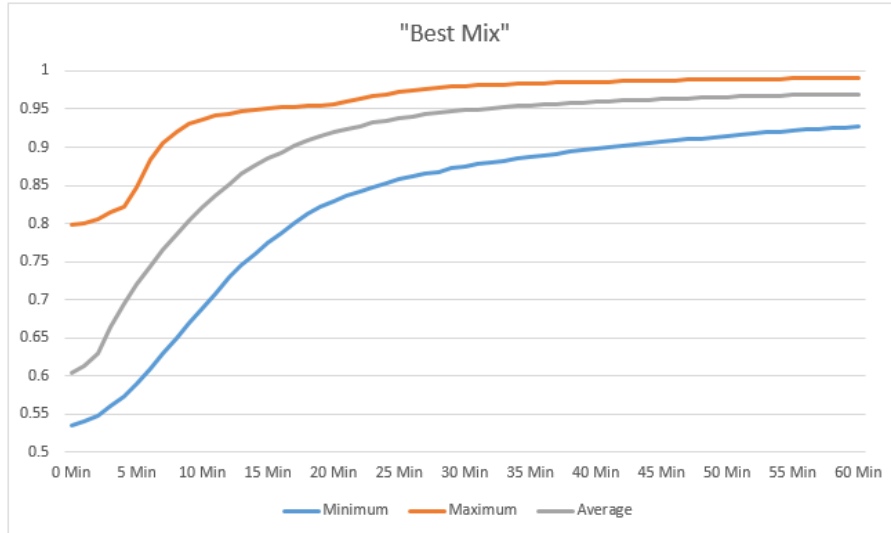


This graph shows that "Best Mix" on average tends to grow quicker, however it slows down and is eventually caught up by Hill Climbing. Specifically at minute 60, "Best Mix" had a fitness greater than Hill Climbing by 0.0036.

## Breakdown

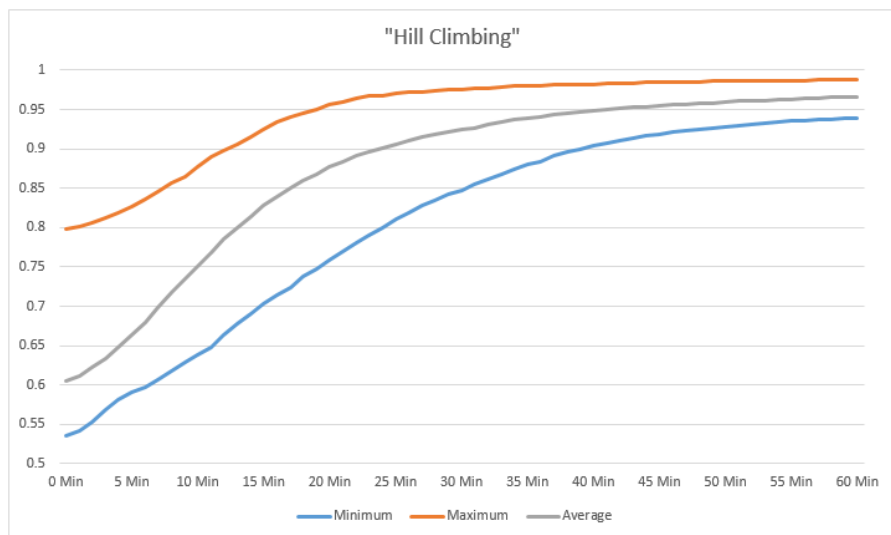
Below is the minimum, maximum, and average plotted for both Best Mix and Hill Climbing

### Best Mix



Minimum: 0.9269  
Maximum: 0.9905  
Average: 0.9697

### Hill Climbing



Minimum: 0.9392  
Maximum: 0.9877  
Average: 0.96638



## Additional Graph Questions

a) What mutations were used in the hill climbing?

Hill climbing selects genes and mutates the gene to a new value. At each hill climbing iteration, it was possible that a hill climbing will mutate the same genome up to ten times without changing the original genome. During this time, if it finds that the mutation has produced a better genome it swaps the original with the new one.

b) What was adaptive about the adaptive hill climbing and what heuristics were used in the adaptation?

Adaptive hill climbing builds a map of the triangles to the genes. Each gene has its own probability of being chosen. In addition, each gene has a step size that dictates the amount of change that can be occurred. The direction of the gene dictates in which direction that step is taken. A change is only applied if it results in a better genome fitness. Also, if the fitness is better, the step size increases, the probability of being chosen increases, while the direction of the step remains the same. If it's worse the step size and probability of being chosen decreases, while the direction is reversed.

c) How often was each type of cross-over used?

Crossover composes of 50% of the generations. Single point cross over is used half of the time while double point is used the second half.

d) Did your cross-over include mutation? If so, how?

When a gene was being crossed over, it had a probability that the gene would mutate during the cross over.

e) How did your cross-over do selection?

At each cross-over step we create a pool of our top 12 genomes. We loop over each genome from the top 12, and cross-over this genome with another genome from within the tribe. Each time that a genome undergoes crossover, it is randomly paired with one of the genomes from this pool. Genomes in the pool with the higher fitness had a higher probability of being chosen.

f) How were genomes shared between tribes?

We instantiate a global ordered genome list that is randomly added to by the different tribes. Whenever a tribe does contribute to this genome list, it will only submit copies its genomes with the highest fitness.

g) What caused switching between hill climbing and cross-over?

Hillclimbing is used to quickly and efficiently bring genomes within a fitness level of 95 percent. At this point, crossover is regularly used to promote diversity.

## Example Output

During this additional analysis, both algorithm ratios were tasked to recreate the Mona Lisa from a 512x413 png file. Seen below is the Mona Lisa and the two recreated images side by side (Not to scale in order to compare images side by side):



Mona Lisa



Mona Lisa using "Best Mix"



Mona Lisa using "HillClimbing"

## Additional Images

Some of our favorite images that we encountered during the development of The Triangle Genome Project.

Piet Mondrian 250x188



Fitness of 0.9669



Great Wave of Kanagawa 250x172



Fitness 0.9798

