Problem Statement

To construct the optimal structure for supporting a surface reaching the furthest possible distance over an open space with only connection to one horizontal surface using Lego® bricks. The goal is to use a genetic algorithm, parallelized for the super computer, to search for the optimal structure given that there is no direct way to calculate the best structure. This support system could be used in various design scenarios: docks, balconies, surfaces (tables, chairs), and bridges. This system could also be more economical and environmentally friendly with optimized resources.

Procedure

The structure of the genetic algorithm follows that of evolution in Biology, with each generation of a population undergoing: selection, recombination "mating," and then mutation. For this problem one support structure was defined as a chromosome, and a population was a group of chromosomes.

Before proceeding into evolving the chromosomes, an initial population must be constructed to undergo the procedure. This initial population was randomly generated by starting with a ledge brick of an inputted length, and then randomly placing bricks on any openings from that brick, not including the bottom of the ledge. The randomly placed bricks consisted of an infinite supply of a discrete group of Lego shapes, in this case: 2x1, 3x1, and 4x1. Then the population undergoes the generation process for an inputted number of generations.

1. Selection: From the population an intermediate population is selected based on each chromosome's fitness level. The fitness level of a chromosome is defined as the center of mass over the ledge, if it's not over the ledge it is unfit and dies off, scaled by the x-distance to the right, as in this experiment the right is the desired direction. All unfit chromosomes are removed from the population, and a population of the same size is sampled using weighted selection with replacement from the prior population.
2. Recombination: The intermediate population then undergoes recombination as to share desired qualities to potentially find a better structure. The chromosomes are spliced, where a cluster of an inputted number of bricks is removed from a shared location in the plane in two chromosomes, and then exchanged so that they have the splice from their matched chromosome. They are spliced with the chromosome next in line in the population.
3. Mutation: The chromosome each then have a certain inputted probability of receiving a mutation, which is where a brick is randomly added to the chromosome.

After this process is completed for the desired number of generations a final population is produced, and the most fit chromosome is selected as the solution.

Solution

The solution that was produced during the experiment was a two barred support with an open inner frame that extended off the edge to a point where the surface would be attached.

Challenges

The first challenge was determining from the vast amount of possible Lego pieces to use, which would be included for this experiment. There are very complex pieces featuring curves and connections on all 6 faces. To simplify the computation only one dimensional Legos were considered and the subset of lengths 2-4 were selected. It was then chosen that there would be an infinite supply of all of the Legos that were being used with equal likelihood of any particular piece being chosen, instead of weighting the selection.

During splicing of Lego pieces the exchanged pieces weren't likely to be able to fit like a puzzle piece into the alternate chromosome, so to avoid two pieces occupying the same spaces the pieces the conflicting piece in the receiving chromosome was removed.

It was also the case that upon splicing pieces would result in having no connection to the ground, they were floating. It was chosen that all floating pieces would be removed, like in a real Lego structure where they would drop due to gravity.

The computations required a lot of searching of arrays of arrays so it became computational intensive, and the program was met with memory and walltime limitations upon initial trials. This was remedied by adding save points of the populations, and running the job with higher memory and walltime requirements.

Improvements

Given more time the project would have taken on larger and more complex parameters in order to derive more useful large scale structures; however, there wasn't enough time to allow this computations to perform.

There also existed some phenomena in which the super computer wouldn't process the job and return no errors or clues as to why it was unable to perform any tasks. Further time would allow for the analysis and resolving of these mysterious errors by inquiring with the HPCC staff.

Resources

1http://www.mabuno.com/simple-yet-sophisticated-modern-home-design-inaustria/

shining-house-by-the-lake-with-glass-windows-and-long-balcony/

2http://www.doc.ic.ac.uk/~sgc/teaching/pre2012/v231/lecture16.html

3http://en.wikipedia.org/wiki/File:Mutation_and_selection_diagram.svg

4http://www2.estrellamountain.edu/faculty/farabee/biobk/biobookmeiosis.html

5http://3greatacts.com/the-nature-of-the-beast-ii-emergence/

6http://thebrickblogger.com/2012/07/lego-super-heroes-batman-2-video-game/

7http://textflow.mcgraw-hill.com//parser.php?secload=11.2&fake&print

8http://www.cs.colostate.edu/~genitor/MiscPubs/tutorial.pdf

9http://www.otlet-institute.org/wikics/2D_Genetic_Algorithms.html#toc-Section-6