# Project 2: Implementation of a Genetic Algorithm for Determining the Global Energy Minimum of a Lennard-Jones Cluster

Eric Ross

***Abstract*—This paper describes the implementation of a genetic algorithm to solve for the minimum potential energy of various sized Lennard-Jones clusters. Using this algorithm, only a relatively small number of energy evaluations and minimizations are required to find the global minimum. By contrast, a simple random search algorithm often has issues finding the global minimum and instead finds a local minimum, especially with larger clusters.**

## I. Introduction[1]

The genetic algorithm is a class of algorithms based on Darwin's idea of natural selection. It uses operators that correspond to the intermixing of DNA from two parents mating to produce an offspring, introduces random mutations in the DNA, and selects children to survive based on 'survival of the fittest.' Genetic algorithms can be used to optimize any problem where the variables (genes) can be encoded to form a string (chromosome).

In this paper we wish to minimize the potential energy of a cluster of atoms whose potential is given by

$$U = \sum_{i=1}^{N-1} \sum_{j>i}^{N} (r_{ij}^{-12} - r_{ij}^{-6} \tag{1}$$

where $r_{ij}$ is the interatomic spacing between two atoms and $N$ is the number of atoms in the cluster.

Genetic algorithms are often more effective than similar global-finding algorithms such as simulated annealing and basin-hopping. They also drastically outperform local search methods such as quasi-Newton methods, because as cluster size is increased the number of local minimums increases exponentially, making a random search nearly impossible. However, utilization of a local minimum search technique is used in this paper to effectively turn the search space into a stepped surface. This stepping can be seen clearly in the results presented later.

## II. Methods

A flowchart of the genetic algorithm is shown in Figure 1.

### A. Initial Population

An initial population of clusters is generated by randomly choosing each atom's x, y, and z coordinates in the cluster to be in the interval [0,3.5]. This is repeated for however many clusters are desired; for this paper each population consisted of $N_{pop} = 10$. In general, smaller populations decrease computation time, especially as the cluster size is increased.

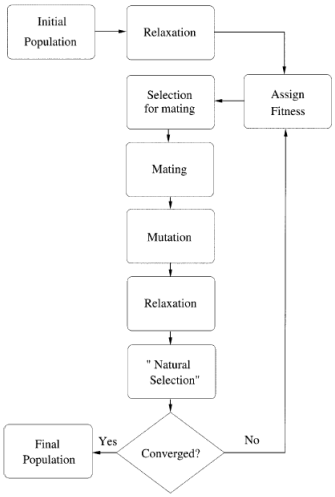


Fig. 1. Schematic flow chart for the cluster geometry optimization GA program.

Next all the clusters in the initial populations are relaxed to their nearest local minima by minimizing the cluster potential energy as a function of the cluster coordinates, using a quasi-Newton routine.

### B. Fitness

To determine which cluster arrangements are best, we introduce the concept of fitness. This value is based on the potential energy of the cluster and is normalized based on the range of potential energies over all clusters. This fitness scaling ensures that for every generation, the fittest (lowest energy) cluster has a fitness of one. The fitness ($F_i$) of the ith member of the population (with potential $V_i$) is given by

$$F_i = \exp\left(-\rho\left[\frac{V_i - V_{min}}{V_{max} - V_{min}}\right]\right) \tag{2}$$

where $V_{min}$ and $V_{max}$ are the values of the highest and lowest energy clusters in the current population and $\rho$ is a scaling factor which was set to 3.

### C. Selection of parents for mating

After fitness has been assigned to all members of the initial generation, a new generation is formed by picking two parents

and generating an offspring. This is repeated until a suitable number of offspring are generated and these offspring are then either admitted to the next generation or allowed to die off, based on their fitness.

To select two parents for mating, a random cluster is selected from the population. Next a random number is chosen between 0 and 1. If the randomly selected cluster has a fitness greater than the random number, it is accepted as one of the parents to mate. If the candidate cluster is rejected for mating, another is picked and the process is repeated. The second parent is chosen in the same method and once both parents are selected they are subjected to the mating operation.

### D. Mating

The mating step is very important and determines the overall direction that the GA algorithm takes. It is important to have operators that pass on important genes will removing genes that trend toward lower fitness. This project utilized three different mating operators: an arithmetic mean, a geometric mean, and the cut-and-splice method.

*1) Arithmetic Mean:* The arithmetic mean simply loops through each coordinate and takes the average position using an arithmetic mean of atoms between each parent. That is

$$C_i = (P1_i + P2_i)/2 \tag{3}$$

where $C_i$ is the atomic coordinates of the ith atom in the child, and $P1_i$ and $P2_i$ are the atomic coordinates of the ith atom in the two mating parents.

*2) Geometric Mean:* The geometric mean operator acts the same way except the geometric mean of each parent's atom position is taken instead of the arithmetic mean.

$$C_i = \sqrt{P1_i \times P2_i} \tag{4}$$

*3) Cut and Splice Method:* The cut and splice method is a much more involved operator but it has shown to be very effective in converging towards the global minimum. It is outlined as follows:

1) Select a random atom CP (the cut point) in P1.
2) Select a random number $S \in [1, N-1]$.
3) Create a list $L_{P1}$ consisting of the $N$ atoms that belong to P1 ordered according to an increasing distance from the location of CP.
4) Copy the first S atoms from $L_{P1}$ to $C1$.
5) Repeat step 3 for P2.
6) Remove from $L_{P2}$ atoms that are too close (i.e. at a distance smaller than 0.5) to particles already copied to $C1$.
7) Copy the first $(N-S)$ atoms from $L_{P2}$ to $C1$.
8) If less than $N$ atoms were coppied to $C1$ then the child is completed with particles placed at random locations.

Figure 10 shows this process more clearly.

Mating continues until a predetermined number of offspring are generated. The number of matings was set to 80% of the total population for reasons that will be discussed in later sections.

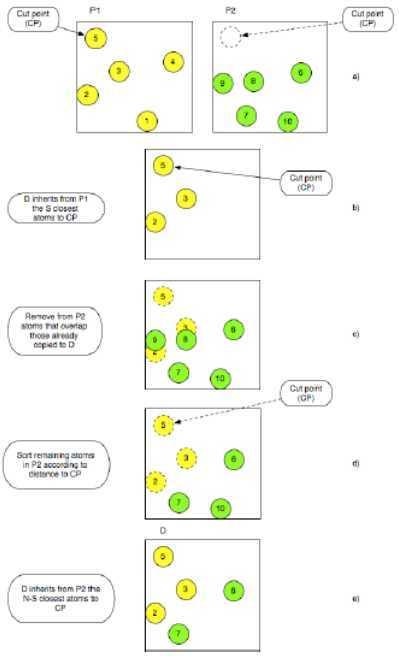


Fig. 2. Application of the generalized cut and splice mating operator between two clusters with $N = 5$ and $S = 3$.

Once all the offspring are generated, the child clusters are subsequently relaxed to their nearest local minima using the same techniques used for the initial population.

### E. Mutation

Although the mating operations do lead to a mixing of the genetic material in the offspring, oftentimes this mixing does not provide much new genetic material to be introduced into the population. This leads to stagnation and early convergence on an energy value that is only a local minimum instead of a global minimum. To avoid this happening, a mutation operator is introduced. The mutation is only performed on a fixed proportion of children, however, otherwise we lose many of the strengths of the overarching genetic algorithm. For the Lennard-Jones clusters used in this project, a mutation percentage of 20% was found to be optimal.

If a child is chosen to be mutated, all of the atoms within the child cluster are reassigned with new coordinates in the interval [0,3.5], similar to how the initial clusters were formed.

### F. Natural Selection

Now that we have our initial population and offspring equal to about 80 percent of the initial population, it is time to decide which genes will be admitted into the next generation.

The algorithm employed for this project uses a constant population size between generations. The set of clusters in the previous generations and the set of offspring clusters are first ranked by their potential energy. The clusters with the lowest energy between the two sets are chosen to make up the next generation. Since the number of offspring is less than the size of the population, we ensure that the lowest energy clusters from the previous generation are transferred to the next generation. This is known as elitism.

### G. Convergence

The algorithm can be stopped in a few different ways but the main convergence criterion was the difference between the maximum energy in the population and the minimum energy in the population. If this value reaches a certain threshold, most of the clusters in the population are the same energy so more generations are going to look approximately the same.

## III. Results

The GA was run for clusters of $N = 5, 7, 10, 13$. Plots of the potential energy are shown below.

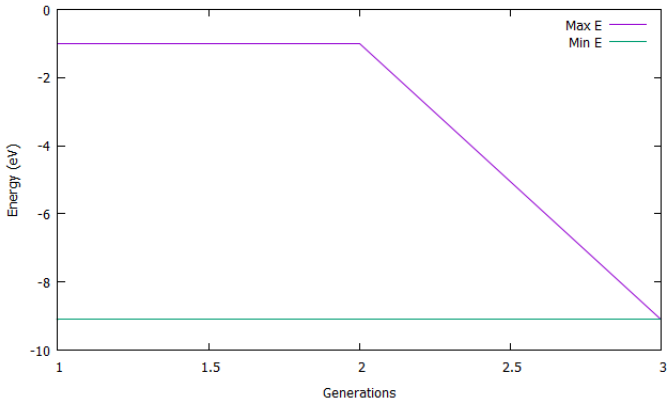


Fig. 3. Maximum and minimum potential energy of a Lennard-Jones cluster of $N = 5$ as a function of generations.

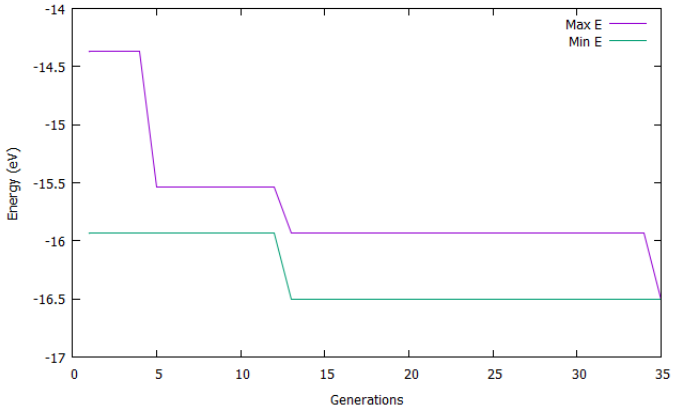


Fig. 4. Maximum and minimum potential energy of a Lennard-Jones cluster of $N = 7$ as a function of generations.

The smaller clusters converge rapidly and the cluster with $N = 5$ even contains the global minimum in its first generation. As the cluster size is increased it takes longer to reach the global minimum but overall the computational effort is much less than a simple random search. These figures clearly show the step-wise fashion that the energy proceeds in. All the minimums found were validated and match previously published values. The structures of each cluster are shown as well.

All the structures are highly symmetric which is why the structures are more stable and lower in energy than other possible structures.

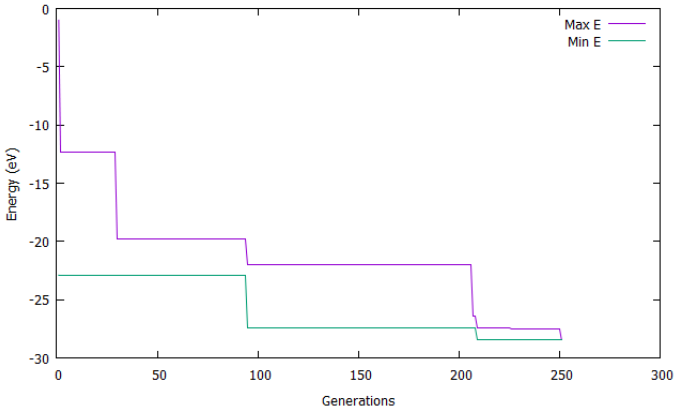


Fig. 5. Maximum and minimum potential energy of a Lennard-Jones cluster of $N = 10$ as a function of generations.

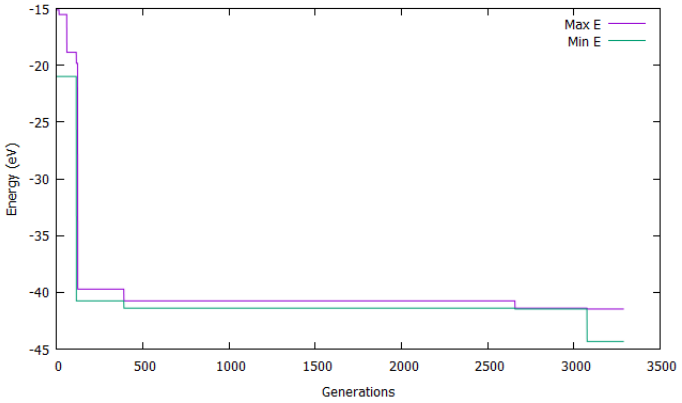


Fig. 6. Maximum and minimum potential energy of a Lennard-Jones cluster of $N = 13$ as a function of generations.

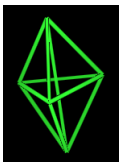

Fig. 7. Structure for global minimum for $N = 5$.

Fig. 8. Structure for global minimum for $N = 7$.

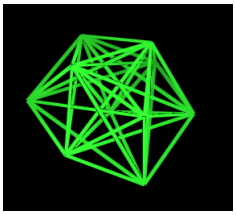

Fig. 9. Structure for global minimum for $N = 10$.

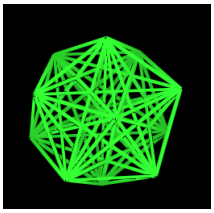

Fig. 10. Structure for global minimum for $N = 13$.

## IV. Conclusions

The genetic algorithm is an extremely powerful technique for finding unbiased global minimums of different-sized Lennard-Jones clusters. There are countless ways to employ the algorithm and choice of mating and mutation operators determine how and if the algorithm converges. Starting parameters such as the initial population size, the number of offspring to produce, and the mutation rate also play an important role in the algorithm. These choices are especially important as the size of cluster is increased to $N > 50$.

## References

[1] Roberts, Chris, Roy L. Johnston, and Nicholas T. Wilson. "A Genetic Algorithm for the Structural Optimization of Morse Clusters." Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta) 104, no. 2 (2000): 123-30. doi:10.1007/s002140000117.

# APPENDIX