Method of Global Optimization of Lennard Jones Potential Cluster using Basin Hopping

Dustin S. Carter
University of Nevada, Las Vegas

Abstract

Basin Hopping is a algorithm for finding the Global Minima for the Lennard-Jones Potential Energy Surface for clusters of atoms. The program transforms the PES into a collection of interpenetrating staircases. This effectively removes transition state regions and leaves only the Energy minima for the algorithm to optimize. Cluster sizes of n=13, n=15, and n=19 are evaluated. n=13 and n=15 are found to be in good agreement with literature without changing the step size or temperature parameters. n=19 was not in good agreement because the temperature and step size parameters were not altered.

Introduction

"Optimization is the selection of a best element (with regard to some criterion) from some set of available alternatives."[1] In many contexts this consists of maximizing or minimizing a real function by systematically choosing input from an allowed set. Global optimization is the process to find the global maximum or global minimum over the total allowed set of input values. The applications of global optimization are countless and thus is a great area of interest that impact all areas of life. Finding better and faster methods to optimize a system has lead to a variety of analytical, numerical, and computational methods. Oddly enough there is no one best way to optimize a system, and is highly dependent on the application.

The application that this paper concerns itself with is finding the global energy minima for the Lennard Jones Potential Energy Surface for clusters of atoms. The Potential is :

$$V(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

where ε and $2^{1/6}\sigma$ are the pari equilibrium well depth and separation, respectively.
The main problem for this application is to effectively and efficiently finding the minima for the system and then determine which of them is smallest. This might seem like a trivial matter philosophically but in practice it becomes much more difficult as the total number of possible energy configurations and thus minima as a result of the increased number of atoms increases. Not only does the complexity of the energy potential surface increase, the number of minima become far more abundant. So traversing the landscape becomes much more difficult, i.e., higher abundancy of minima as well as changing well size and depth make it difficult to reliably finding the minima. One solution that has been developed is known as basin hopping.

Theoretical Methods

"The present approach has been guided by previous work on energy landscapes which has identified features that enable the system to locate its global minimum efficiently.44 In particular, analysis of model energy landscapes, using a master equation approach for the dynamics, has provided good evidence that such a surface should have a large potential energy gradient and the lowest possible transition state energies or rearrangement barriers.53 These results immediately suggest a simple way to transform the PES which does not change the global minimum, nor the relative energies of any local minima. We consider the transformed energy $\tilde{E}$ defined by

$$\tilde{E}(X) = \min\{E(X)\}$$

where X represents the 3N-dimensional vector of nuclear coordinates and min signifies that an energy minimization is performed starting from X."[3] This essentially creates a set up step functions or plateaus of energy minima, without

the energy barriers to overcome. This is equivalent to hopping from one energy minima to the next or as the name implies, to basin hop.

The exploration of the potential energy surface is explored using a canonical Monte Carlo simulation at constant reduced temperature of 0.8. At each step, all the coordinates were displaced by a random number within [-1,1] times the step size, which is adjusted to give an acceptance ratio of 0.5. This allows large step sizes between 0.36 - 0.40. In each cluster range considered seven separate runs were conducted. Five of these seven consists of 5000 Monte Carlo steps starting from different randomly generated configurations of atoms confined to a sphere of radii 5.5 reduced units. The following geometric optimizations employs a container of radius one plus the value required to contain the same volume per atom as the fcc primitive cell.

The Monte Carlo convergence criterion is the RMS gradient that is less than 0.01 in reduced units and the energy to change by less than 0.1 ε between consecutive steps in the conjugate gradient search.

Basin-hopping philosophy is that the steps are taken directly between minima using eigenvector-following to calculate pathways. This adds computational expense but for the reduced configuration space it leads to an advantage for finding the minima.

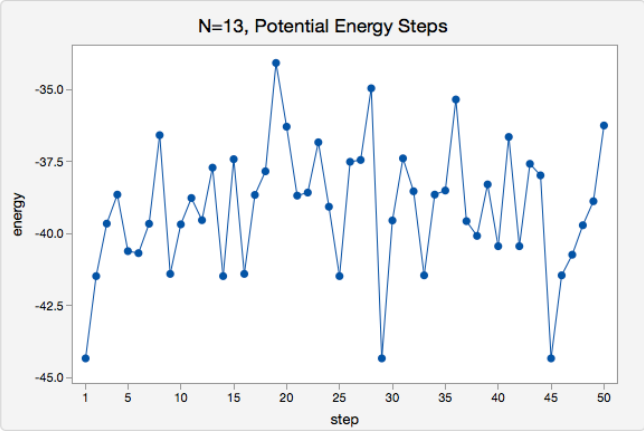
In this particular study, we looked at n=13, n=15, and n=19. The parameters of interest to optimize are temperature and step size. Each of these parameters correspond to the basin depth, and basin width. If the step size is too large you might miss a minima, if it is too small it adds to the computation time. For temperature, the size of the well is increased or decreased and therefore the size of the hop might get the algorithm stuck in a well, without ever being able to find the rest of the minima.

The code for the basin hopping is the following:

```
"""
Created on Wed Nov 22 21:48:12 2017

@author: dustinscarter
"""
import numpy as np

def LJ(r):
    r6 = r**6
    r12 = r6*r6
    return 4*(1/r12 - 1/r6)

def total_energy(positions):
    """
    Calculate the total energy
    input:
    positions: 3*N array which represents the atomic positions
    output
    E: the total energy
    """
    E = 0
    N_atom = int(len(positions)/3)

    #positions = [x0, y0, z0, x1, y1, z1, .....  , xn, yn, zn]
    for i in range(N_atom-1):
        for j in range(i+1, N_atom):
            pos1 = positions[i*3:(i+1)*3]
            pos2 = positions[j*3:(j+1)*3]
            #print('pos1:  ', pos1)
            #print('pos2:  ', pos2)
            dist = np.linalg.norm(pos1-pos2)
            #print(i,j, dist)
            E += LJ(dist)
    return E

N_atom = 19

def init_pos(N, L=5):
    return L*np.random.random_sample((N*3,))

def neighbor(pos_now, kT):
    N = len(pos_now)
    return pos_now + kT*np.random.random_sample((N,))

def acceptance_probability(dE, kT):
    if dE<0:
        return 1
    else:
        return np.exp(-dE/kT)



from scipy.optimize import basinhopping


pos = init_pos(N_atom)
res = basinhopping(total_energy, pos, niter=50, T=2.0, stepsize=3.0, disp=True)
```

Results and Discussion

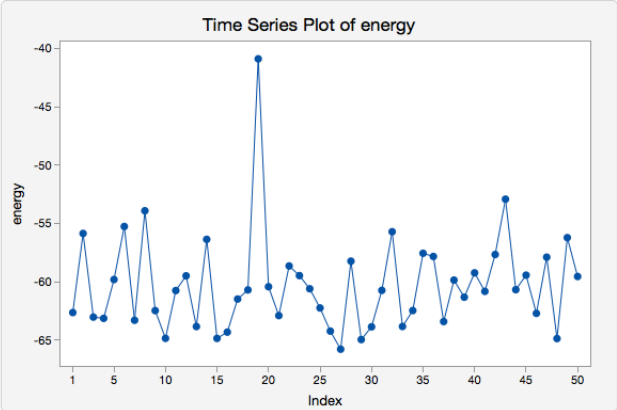
Time Series Plot of energy



Summary Statistics

N	Mean	StDev	Minimum	Maximum
50	-39.126	2.237	-44.327	-34.072

Time Series Plot of energy

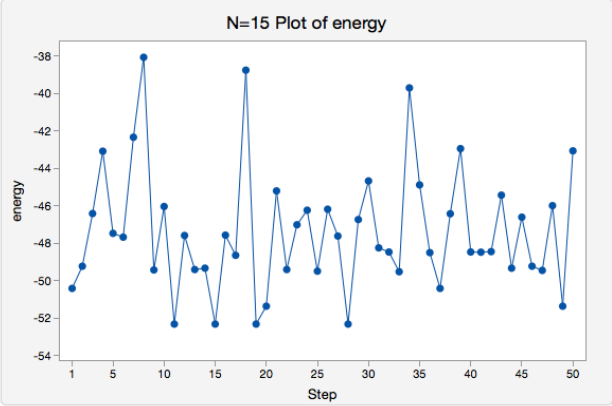


Summary Statistics

N	Mean	StDev	Minimum	Maximum
50	-60.265	4.191	-65.787	-40.891

The three clusters of atoms are as follows n=13,15,19.

Time Series Plot of energy



Summary Statistics

N	Mean	StDev	Minimum	Maximum
50	-47.391	3.293	-52.323	-38.056

The energy minima that was calculated are as follows:

N	Energy	Established	% Diff
13	-44.327	-44.326801	0.00044893
15	-52.323	-52.32263	0.00070715
19	-65.787	-72.65978	9.45885054

For n=13 and n=15 without adjusting the step size or temperature the basin hopping algorithm converged quickly to the established values found in [3]. We did not adjust the temperature or step size of n=19 and as such the algorithm jumped over the smallest minima, or was stuck in one of the wells, preventing it to basin hop into the global minima.

Conclusion

Global optimization continues to be an area of promising research due to its enumerable applications. Finding efficient and effective algorithms to find the global minima especially in the physical sciences will continue to be of great interest. The basin hopping algorithm does a good job at finding the global minima if the appropriate step size and temperature parameters are tuned for the application of interest. As a result of not tuning the error of n=19 is quite significant. This

demonstrates the need to have follow up optimizations, to try a band of step sizes and temperatures. So this algorithm is not an algorithm that can tune itself, and requires continual supervision. Developing the program further to adjust itself stochastically in regards to the parameters themselves instead of only the energy minima could lead to less supervision of the program and a program that can be used for higher order atomic clusters, but computational time may go up as a consequence.

References

[1] https://en.wikipedia.org/wiki/Mathematical_optimization

[2] http://www-wales.ch.cam.ac.uk/pdf/JPCA.101.5111.1997.pdf

[3] http://doye.chem.ox.ac.uk/jon/structures/LJ/tables.150.html