Metropolis Monte Carlo
and
Basin Hopping
Global Optimization

• We are currently learning about tools to explore beyond the current local minima and search for the global minimum, or the most stable structure of a chemical system.
Review: Boltzmann Distribution

Probability of finding a molecular configuration in the canonical ensemble (NVT)

\[ P(r) \propto e^{-\frac{V(r)}{k_bT}} \]
Review: Boltzmann Distribution

Probability of finding a molecular configuration in the canonical ensemble (NVT)

\[ P(r) \propto e^{-\frac{V(r)}{k_b T}} \]

If you run a NVT MD trajectory you will sample the Boltzmann Distribution for a fixed temperature
Monte Carlo Methods

• In this course we have talked about some *numerical methods*:
  – Methods for local optimization
    • Gradient Descent and Newton’s Method
  – Euler’s method used in molecular dynamics simulations

• We can also use random numbers to solve quantitative problems
  – Methods for which utilize random numbers are known as *Monte Carlo methods*
How can we use random numbers to solve quantitative problems?

• Procedure for calculating $\pi$:
How can we use random numbers to solve quantitative problems?

- Procedure for calculating π:
- Select two uniform random numbers between 0 and 2 for x and y coordinates
- Label the point as in (red) or out (black) of the circle

(0.46, 1.16)
How can we used random numbers to solve quantitative problems?

• Procedure for calculating $\pi$:
  • Select two uniform random numbers between 0 and 2 for $x$ and $y$ coordinates
  • Label the point as in (red) or out (black) of the circle
  • Repeat many times
How can we use random numbers to solve quantitative problems?

\[
\frac{\text{Area}_{\text{circle}}}{\text{Area}_{\text{square}}} \approx \frac{\text{red \_dots}}{\text{all \_dots}} = \frac{\pi r^2}{(2r)^2} = \frac{\pi}{4}
\]

- This is a way of approximately finding area or doing integration.
- Numerical integration
- Accuracy of the value of Pi depends on the number of random points used
- This is not more efficient for calculating Pi than other numerical methods such as a Reimann Sum
You could solve this problem using a Reimann sum as well and it would be more efficient!

\[
\text{Computed Area} = \pi r^2
\]

\[
\pi = \frac{\text{Computed Area}}{r^2}
\]
Dimensionality

- What would happen to the expense of using a Reimann Sum to find the area of a sphere?
  - Number of samples required scales *exponentially* with dimension

- What would happen to the expense of using a Monte Carlo method like the previous example to find the area of a sphere?
  - Same as a Reimann Sum! Exponential scaling with a dimension!
Dimensionality

- What would happen to the expense of using a Monte Carlo method like the previous example to find the area of a sphere?
  - Same as a Reimann Sum! Exponential scaling with dimension!

\[ N^2 \text{ samples required} \]

\[ N^3 \text{ samples required} \]
Metropolis Monte Carlo

• Metropolis Monte Carlo is a method which exists to tackle the issue of dimensionality!
• It is a method for obtaining a random sequence of configurations from a probability distribution when sampling is difficult (high dimensional systems)
• I will first introduce the procedure of the algorithm.
• Then give an example of how this can be useful
Metropolis Monte Carlo

1. Start with an initial structure $R$ with potential energy $V(R)$ and probability, $P(R) = e^{-V(R)/kT}$

Boltzmann Distribution

$$P(r) \propto e^{-\frac{V(r)}{k_bT}}$$
Metropolis Monte Carlo

1. Start with an initial structure $R$ with potential energy $V(R)$ and probability, $P(R) = e^{-V(R)/kT}$

2. Generate a new structure $R'$ with energy $V(R')$ and probability $P(R) = e^{-V(R')/kT}$
1. Start with an initial structure $R$ with potential energy $V(R)$ and probability, $P(R) = e^{-V(R)/kT}$

2. Generate a new structure $R'$ with energy $V(R')$ and probability $P(R') = e^{-V(R')/kT}$

If $P(R') \geq P(R)$:
   Accept this configuration
   \textit{i.e.} $R = R'$

\textit{Another way of writing this is}
If $V(R') \leq V(R)$:
   Accept this configuration
   \textit{i.e.} $R = R'$
1. Start with an initial structure R with potential energy V(R) and probability, \( P(R) = e^{-\frac{V(R)}{kT}} \)

2. Generate a new structure R’ with energy V(R’) and probability \( P(R') = e^{-\frac{V(R')}{kT}} \)

If \( P(R') \geq P(R) \):  
Accept this configuration  
\( i.e. \ R = R' \)

Another way of writing this is

If \( V(R') \leq V(R) \):  
Accept this configuration  
\( i.e. \ R = R' \)

If \( V(R') > V(R) \) is false  
move to next part of algorithm…

If \( V(R') > V(R) \) is false  
move to next part of algorithm…
1. Start with an initial structure $R$ with potential energy $V(R)$ and probability, $P(R)=e^{-V(R)/kT}$

2. Generate a new structure $R'$ with energy $V(R')$ and probability $P(R')=e^{-V(R')/kT}$

If $P(R') < P(R)$ or $V(R') > V(R)$:
Generate a random number between 0 and 1 called RAND
if $e^{-\left(V(R')-V(R)\right)/kT} > \text{RAND}$:
Accept this configuration
i.e. $R = R'$
1. Start with an initial structure R with potential energy V(R) and probability, \( P(R) = e^{-V(R)/kT} \)

2. Generate a new structure R’ with energy V(R’) and probability \( P(R') = e^{-V(R')/kT} \)

If \( P(R') < P(R) \) or \( V(R') > V(R) \):
   Generate a random number between 0 and 1 called RAND
   if \( e^{-(V(R')-V(R))/kT} > \text{RAND} \):
      Accept this configuration
      \( i.e. R = R' \)

RAND = 0.23
\[ e^{-(V(R')-V(R))/kT} = P(R')/P(R) = 0.07 \]
So,
Reject this configuration!
R’, P(R’)

R, P(R)
1. Start with an initial structure R with potential energy $V(R)$ and probability, $P(R) = e^{-V(R)/kT}$

2. Generate a new structure $R'$ with energy $V(R')$ and probability $P(R) = e^{-V(R')/kT}$

If $V(R') \leq V(R)$:
Accept this configuration
\textit{i.e.} $R = R'$

If $V(R') > V(R)$:
Generate a random number between 0 and 1 called RAND
if $e^{-V(R')-V(R)/kT} > \text{RAND}$:
Accept this configuration
\textit{i.e.} $R = R'$

REPEAT THIS PROCEDURE MANY TIMES TO GET A SERIES OF POINTS FROM A PROBABILITY DISTRIBUTION
Metropolis Monte Carlo

• A method for obtaining a random sequence of configurations from a probability distribution when sampling is difficult (high dimensional systems)
• How could we use this?
• There are many applications but one example could be the average potential energy of a system
• Here is one way of calculating the average potential energy

\[ \langle V \rangle = \sum_{i=1}^{n} P(r_i) \ast V(r_i) \]
Metropolis Monte Carlo

• We could also use Metropolis Monte Carlo and sample the known probability distribution, \( P(r) \).
• Then, we can average the values sampled from this probability distribution

\[
\langle E \rangle = \frac{1}{n} \sum_{i=1}^{n} V(r_i)
\]

Where \( V(r) \) is selected from the Probability distribution \( P(r) \)

Metropolis Monte Carlo is superior for High Dimensional Systems!
Basin Hopping

• Transforms the PES to eliminate barriers between states

• Sample the transformed PES (dashed line) with Metropolis Monte Carlo
1. Start with an initial structure \( R \). Optimize this structure to get \( R_{\text{opt}} \) with potential energy \( V(R_{\text{opt}}) \) and probability, \( P(R_{\text{opt}}) = e^{-V(R_{\text{opt}})/kT} \).

2. Generate a new structure \( R' \). Optimize this structure to get \( R_{\text{opt}}' \) with potential energy \( V(R_{\text{opt}}') \) and probability \( P(R_{\text{opt}}') = e^{-V(R_{\text{opt}}')/kT} \).

If \( P(R_{\text{opt}}') \geq P(R_{\text{opt}}) \):
- Accept this configuration
  - \( i.e. \ R = R' \)

If \( P(R_{\text{opt}}') < P(R_{\text{opt}}) \):
- Generate a random number between 0 and 1 called \( \text{RAND} \)
  - if \( e^{-(V(R_{\text{opt}}')-V(R_{\text{opt}}))/kT} > \text{RAND} \):
    - Accept this configuration
    - \( i.e. \ R = R' \)

REPEAT THIS PROCEDURE MANY TIMES TO GET A SERIES OF POINTS FROM A PROBABILITY DISTRIBUTION.
Basin Hopping

1. Start with an initial structure $R$. Optimize this structure to get $R_{opt}$ with potential energy $V(R_{opt})$ and probability, $P(R_{opt}) = e^{-V(R_{opt})/kT}$
Basin Hopping

- Generate a new structure $R'$. Optimize this structure to get $R_{opt}'$ with potential energy $V(R_{opt}')$ and probability, $P(R_{opt}')=e^{-V(R_{opt}')/kT}$
Basin Hopping

• If $P(R_{opt}') \geq P(R_{opt})$:
  • Accept this configuration $R = R'$
Basin Hopping

1. Start with an initial structure $R$. Optimize this structure to get $R_{\text{opt}}$ with potential energy $V(R_{\text{opt}})$ and probability $P(R_{\text{opt}})=e^{-V(R_{\text{opt}})/kT}$
Basin Hopping

2. Generate a new structure $R'$. Optimize this structure to get $R_{opt}'$ with potential energy $V(R_{opt}')$ and probability, $P(R_{opt}') = e^{-V(R_{opt}')/kT}$
Basin Hopping

If $P(R') \geq P(R)$:

Accept this configuration

Else: Move on
Basin Hopping

Generate a random number between 0 and 1 called RAND

- if $e^{-\frac{(V(R_{opt'})-V(R_{opt}))}{kT}} > \text{RAND}$: Accept this configuration
- Else: reject
Random Moves

• How to create new configurations?
  – $R \rightarrow R'$

• This can be from any distribution you choose!

• Here are a few examples
  – Random uniform displacement
    • Maxstep $*[-1,1]$ where maxstep is the maximum displacement per degree of freedom and $[-1,1]$ is a uniform random number between -1 and 1
  – Gaussian random displacement
  – Plus more ....

• You will get the opportunity to select a distribution of your choice in the next lab

• You will also explore optimal values for the maximum displacement