Lecture 5, April 13, 2015
MD1: minimization

William A. Goddard III, wag@wag.caltech.edu
316 Beckman Institute, x3093
Charles and Mary Ferkel Professor of Chemistry,
Materials Science, and Applied Physics,
California Institute of Technology

Special Instructor: Julius Su <jsu@caltech.edu>
Teaching Assistant: Daniel Brooks daniel.brooks@caltech.edu
Teaching Volunteer: Yufeng Huang yhuang5@caltech.edu
Homework and Research Project

First 5 weeks: The homework each week uses generally available computer software implementing the basic methods on applications aimed at exposing the students to understanding how to use atomistic simulations to solve problems.

Each calculation requires making decisions on the specific approaches and parameters relevant and how to analyze the results.

Midterm: each student submits proposal for a project using the methods of Ch121a to solve a research problem that can be completed in the final 5 weeks.

The homework for the last 5 weeks is to turn in a one page report on progress with the project

The final is a research report describing the calculations and conclusions
Now that we have a FF, what do we do with it?

Calculate the optimum geometry
Calculate the vibrational spectra
Do molecular dynamics simulations
Calculate free energies, entropies, phase diagrams

.....
Expanding the energy expression about $x_0$, we can write

$$E(\delta) = E_0 + \delta E_0' + \frac{1}{2} \delta^2 E_0'' + O(\delta^3)$$

Where $E' = (dE/dx)_{x_0}$ and $E'' = (d^2E/dx^2)_{x_0}$ at $x_0$.

At an energy minimum we have

$$E'(\delta_{\text{min}}) = E_0' + \delta_{\text{min}} E_0'' = 0$$

and hence

$$\delta_{\text{min}} = -\frac{E_0'}{E_0''}$$

Thus give $E_0'$ and $E_0''$ at point $x_0$ we can estimate the minimum.

If $E(x)$ is parabolic this will be the exact Minimum.

This is called **Newton’s method**

Of course, $E(x)$ may not be exactly parabolic.

But then we recalculate $E_1'$ and $E_1''$ at point 1 and estimate the minimum to be at $x_2 = x_1 - E_1'/E_1''$.

This process converges quadratically. That is the error at each iteration is the square of the previous error.
Newton’s method will rapidly locate the nearest local minimum if $E'' > 0$.

Note in the illustration, that this may not be the global minimum, which is at $x_{2\text{min}}$.

Also, if $E'' < 0$ (eg. Between the green lines) Newton’s method takes us to a maximum rather than a minimum.
Now consider that we only know the slope, not the 2\textsuperscript{nd} derivative. We start at point $x_0$ of a one dimensional system, $E(x)$, and want to find the minimum, $x_{\text{min}} = x_0 + \delta x$

Given the slope, $E' = (dE/dx)_{x_0}$ we know which way to go, but not how far.

Clearly we want $\delta x = - \lambda E'$

That is we move in the direction opposite the slope, $E'$, but how far? With just $E'$, we cannot know. Thus we could pick some initial value $\lambda_1$ and evaluate the energy $E_1$ and slope $E_1'$ at the new point. If the new $E_1'$ has the opposite slope, then we have bounded the minimum, and we can fit a parabola to estimate the minimum

$x_{\text{min}} = x_0 - E_0'/k$ where

$k = (E_1' - E_0')/(x_1-x_0)$ is the curvature of this parabola.
Energy minimization – one dimensional

We would then evaluate $E(x_{\text{min}})$ to make sure that both points were in the same valley (and not $x_2$).

In the case that the 2$^{\text{nd}}$ point was $x_3$, no change in slope, then we want to jump farther until $E'$ changes sign so that the minimum is bounded.

In the case that the 2$^{\text{nd}}$ point was $x_2$, then the new energy would not match the prediction based on the parabola.

In this case we would choose whichever point of $x_0$, $x_{\text{min}}$, and $x_2$ had the lowest energy and we would use the $E'$ to choose the direction, but we would choose the jump, $\lambda$, to be much smaller, say a factor of 2 than before (some people like using the Golden Mean of $2/\sqrt{5} - 1 = 1.6180$)
Energy minimization - multidimensional

Consider a molecule with $N$ atoms ($J=1,N$),
It has $3N$ degrees of freedom ($J_\alpha$, 1x to Nz) where $\alpha = x,y,z$
Denote these $3N$ coordinates as a vector, $\mathbf{R}$.
The energy is then $E(\mathbf{R})$
Starting with an initial geometry, $\mathbf{R}_0$, consider the new geometry, $\mathbf{R}_{\text{new}} = \mathbf{R}_0 + \delta \mathbf{R}$, and expand in a Taylor series

$$E(\mathbf{R}_{\text{new}}) = E(\mathbf{R}_0) + \sum_k (\delta \mathbf{R})_k (\partial E / \partial \mathbf{R}_k) + \frac{1}{2} \sum_{k,m} (\delta \mathbf{R})_k (\delta \mathbf{R})_m (\partial^2 E / \partial \mathbf{R}_k \partial \mathbf{R}_m) + \mathcal{O}(\delta^3)$$

Writing the the energy gradient as $\nabla E = (\partial E / \partial \mathbf{R}_k)$ and the Hessian tensor as $\mathbf{H} = \nabla^2 E = (\partial^2 E / \partial \mathbf{R}_k \partial \mathbf{R}_m)$
This becomes

$$E(\mathbf{R}_{\text{new}}) = E(\mathbf{R}_0) + (\delta \mathbf{R}) \cdot \nabla E + \frac{1}{2} (\delta \mathbf{R}) \cdot \mathbf{H} \cdot (\delta \mathbf{R}) + \mathcal{O}(\delta^3)$$
Newton Raphson method in 3N space

Given $E(R_{\text{new}}) = E(R_0) + (\delta R) \cdot \nabla E + \frac{1}{2} (\delta R) \cdot \mathbf{H} \cdot (\delta R)$

The condition that $R_{\text{new}} = R_0 + \delta R$ lead to a minimum is that

$\nabla E + \mathbf{H} \cdot (\delta R) = 0$

Bearing in mind that $\mathbf{H}$ is a matrix, the solution is

$(\delta R) = - (\mathbf{H})^{-1} \cdot \nabla E$ where $(\mathbf{H})^{-1}$ is the inverse of $\mathbf{H}$

This is exactly analogous to Newton’s method in 1D

$\delta_{\text{min}} = - \frac{E_0'}{E_0''}$

and in multidimensions it is called the Newton-Raphson (NR) method.

There are a number of practical issues with NR. First for Hb, with 6000 atoms the Hessian matrix would be of dimensions 18000 by 18000 and hence quite tedious to solve or even to store.

First we must be sure that all eigenvalues of the Hessian are nonzero, since otherwise the inverse will be infinite.

Such zero eigenvalues might seem implausible. But for a finite molecule with 3 or more atoms there are always 6 zero modes.
**Hessian problems**

For example, translating all atoms of finite molecule by a finite distance in the x, y, or z directions cannot change the energy.

In addition rotating a nonlinear molecule about either the x, y, or z axis cannot change the energy.

Thus we must remove these 6 dof from the Hessian, reducing it to a (3N-6) by (3N-6) matrix before inverting it.

For a linear molecule, there are only 2 rotational modes, but if there are more than 2 atoms (say CO$_2$) the dynamics will almost always lead to nonlinear structures, so we must consider 3N-6 dof. However for diatomics, there is only one nonzero eigenvalue (Bond stretch).

Also all (3N-6) eigenvalues of the Hessian must be positive, otherwise NR will lead to a stationary point that is a maximum for some directions.
Steepest descents – 1st point

Generally it is impractical to evaluate and diagonalize the Hessian matrix, thus we must make do with just, $\nabla E$, the gradient in 3N dimensions (no need to separate out translation or rotation since there will be no forces for these combinations of coordinates).

Obviously, it would be best to move along the direction with the largest negative slope, this is called the **steepest descent** direction, $u = -\nabla E/|\nabla E|$, where $u$ is a unit vector parallel with the vector $\nabla E$ but pointed downhill.

Then $\delta \mathbf{R} = \lambda \mathbf{u}$, where $\lambda$ is a scalar.

Just as in 1D, we do not know how far to move, so we pick some value, $\lambda$, and evaluate $\nabla E_1$ at the new point. Of course $\nabla E_1$ will generally have a component along $u$, $(u \cdot \nabla E_1)$ plus a component perpendicular to $u$.

Here we proceed just as in the 1D case to bound the minimum and predict the minimum energy along the path.
Steepest descents – more on 1st point

Regarding the first $\lambda$, Newton’s method suggests that it be

$$\lambda = 1/k$$

where $k$ is an average force constant (curvature in the $\nabla E$ direction). In biograf/polygraf/ceriusII I used a value of $k=200$ as a generally good guess.

Also note from the discussion of the 1D case, we want the first point to overshoot the minimum a bit so that the slope is positive. This way we can calculate $k$ from the two slopes and predict a refined minimum.

At the predicted minimum, we evaluate the slope in the original steepest descent direction and if it is small enough (significantly smaller than the other 2 slopes) and if the new energy is lower than the original energy, we use the new point to predict a new search direction.
Steepest descents – 2nd point

Starting from our original point $x_0$ with slope $\nabla E_0$, we moved in the direction, $u_0 = -\nabla E_0 / |\nabla E_0|$ to find a final new position $R_1$, along unit vector $u_0$.

Now we calculate $\nabla E_1$ and a unit vector $u_1 = -\nabla E_1 / |\nabla E_1|$. If $R_1$ was an exact minimum along unit vector $u_0$ then $u_1$ will be orthogonal to $u_0$.

In the Steepest Descents (SD) method, we continue as for the 1st point to find the minimum $R_2$ along unit vector $u_1$, and then the minimum $R_3$ along unit vector $u_2$.

The sequence of steps for SD is illustrated at the right. Here we see that the process of minimizing along $u_1$, can result in no longer having a minimum along $u_0$.
Conjugate Gradients

Even for a system like Hb with 18000 dof, the behavior shown in the SD figure is typical, the system first minimizes along \( u_0 \) then along \( u_1 \), then back along \( u_0 \) then back along \( u_1 \), ignoring most of the other 18000 dof until the minimum in this 2D space is reached, at which point SD starts sampling other dof.

The **Conjugate Gradient** method (Fletcher-Reeves) dramatically improves this process with little extra work.

Here we define \( \nu_1 = u_1 - \gamma u_0 \) where \( \gamma = (\nabla E_1 \cdot \nabla E_1) / (\nabla E_0 \cdot \nabla E_0) \)

(note the use of the dot or scalar product of the gradient vectors)

Thus the new path \( \nu_1 \) combines both directions so that as we optimize along \( u_1 \) we simultaneously keep the optimum along \( u_0 \).

(The ratio \( \gamma \) is derived assuming that the energy surface is 2\(^{nd}\) order.) This process is continued, with

\[ \nu_2 = u_2 - \gamma u_1 \]

where \( \gamma = (\nabla E_2 \cdot \nabla E_2) / (\nabla E_1 \cdot \nabla E_1) \)

The new \( \nabla E_2 \) is perpendicular to all previous \( \nabla E_k \).
Conjugate Gradients

In 2D, for a system in which the energy changes quadratically, CG leads to the exact minimum in 2 steps, whereas SD would take many steps.

First large systems, CG is the method of choice, unless the starting structure is really bad, in which case one might to SD for a few steps before starting CG. There reason is that CG is based on assuming that the energy surface is quadratic so that the point is already in a valley and we want to find the optimum in that valley.

SD is most appropriate when we start high up in the Alps and want to jump around to find a good valley, after which we can convert over to CG.
Fine points on CG

To ensure convergence, one must never choose a final point along a path higher than the original point defining the path.

If the final point is higher than the original point then it necessary to take the two lowest energy points and predict a third along the same path that is lower.

One must be careful if the first step does not find a change in sign of the slope along that path. One should predict the new minimum but if the predicted step is much larger (more than 10 times) than the original step, one should jump more cautiously.

With CG the fewest number of points along a pathway would be 1, so that the predict point is close enough to the minimum that a second checking point is not needed.

R. Fletcher and C.M. Reeves, Function minimization by conjugate gradients, Computer Journal 7 (1964), 149-154

Inverse Hessian or Quasi-Newton methods

In Newton-Raphson we choose the new point from

$$\delta \mathbf{R} = - (\mathbf{H})^{-1} \cdot \nabla \mathbf{E}$$

where $$(\mathbf{H})^{-1}$$ is the inverse of the Hessian $$\mathbf{H}$$. For a system for which the energy is harmonic, this takes us directly to the minimum in one step.

Generally, it is too expensive to actually calculate and save the Hessian. However each time we search a path, say in CG, to find the minimum we derive an average force constant in that direction,

$$k = (E_1' - E_0')/(x_1 - x_0)$$

where the gradients are projected along the path.

Thus we can construct an approximate inverse Hessian which we assume initially to be the unit matrix (SG) but each time we minimize along a direction we use $$1/k$$ along this direction to improve our approximation to $$(\mathbf{H})^{-1}$$.
Updating the Inverse Hessian

The inverse Hessian, $H$ is built up iteratively, AVOIDing explicitly inverting the Hessian matrix

$$\lim_{k \to \infty} H_k = H_k^{-1}$$

$$r_{k+1} = r_k - H_k \nabla E(r_k)$$

$H_k = \text{Inverse Hessian (H) approximation}$

$$H_{k+1} = H_k + \frac{(r_{k+1} - r_k) \otimes (r_{k-1} - r_k)}{(r_{k+1} - r_k) \cdot (g_{k+1} - g_k)} - \frac{H_k (g_{k+1} - g_k) \otimes H_k (g_{k-1} - g_k)}{(g_{k+1} - g_k) \cdot H_k \cdot (g_{k+1} - g_k)}$$

Popular versions:
Davidson-Fletcher-Powell (DFS),
Broyden-Fletcher-Goldfarb-Shanno (BGFS), and
Murtaugh-Sargent (MS), are common [see Leach] -
These methods use only current and new points to update the inverse $H$
Better convergence achieved using more points (QM programs)
1st and 2nd Order Methods: Which to use?

general rule of thumb:

– For large molecular systems with available analytical Force Field functions (1st and 2nd derivatives)
  • Start with Steepest descent and switch to conjugate gradient after system is behaving rationally

– For small QM systems that are computationally expensive but for which second derivatives are unavailable
  • Quasi-Newton methods
So Far ... Gradient Based Methods

- Hold one solution at a time
- Look locally to see what direction to move in (gradient of the function at the current solution)
  - Solution structure to closest minimum
- Select the new current solution after deciding how far to move along that path
- These methods are great for finding the local minimum but are not very useful
  - The objective function is not smooth (i.e., not differentiable).
  - There are multiple local minima.
  - There are a large number of parameters.
  - When the global minimum is desired
Global Minima Searching
without an exhaustive search !!!

• **Note:** No guaranteed analytical solution exists for most real (multivariate) applications in Molecular Simulations (MS) ! … but, closest (time-constrained) alternatives are:

• Random search algorithms
  – Monte Carlo
    • Simulated Annealing (adaptation of Metropolis-Hastings)
  – Genetic Algorithms

• Molecular Dynamics

Stochastic Search Algorithms

The Totally Random Algorithm:
- generates random parameter vectors
- evaluates each one, and
- saves the best one that it finds

Monte Carlo and Simulated Annealing (SA) approaches:
- From one solution or time take a random step away from it.
- If step results in a better solution, then it becomes the new solution about which random steps are taken.
- As optimization proceeds, average size of steps decreases (system “cools” down).

Genetic Algorithms (GAs)
- GAs contain a “population” of solutions at any one time.
- 3 Step process:
  - A way to select parents (pop. generation and selection)
  - A mating ritual between the parents (genetics)
  - A survival of the fittest mechanism (a fitness measure)
Simulated Annealing

• A generalization of MC method for examining EOS and frozen states of n-body systems [metropolis et al, 1953]

• By Thermodynamics Analogy (liquids or metals): a melt, initially at high T and disordered, is slowly cooled (i.e. approximately at thermodynamics equilibrium at any time).
  
  – As cooling proceeds, system becomes more ordered and approaches a “frozen” ground state at T=0. This reduces defects (induces softness, relieves internal stresses, refines the structure and improves cold working properties of metal).
  
  – If initial T is too low or cooling is done insufficiently slowly the system may become quenched forming defects or freezing out in metastable states (i.e. trapped in a local minimum energy state).

• Concept: at higher T more configuration become available!!

S. Kirkpatrick; C. D. Gelatt; M. P. Vecchi, Optimization by Simulated Annealing
Start with system at a known configuration \((E)\)

\(T=\text{hot}\)

Frozen=False

While (!Frozen):

Do Until Thermal Equilibrium @T

- Perturb system slightly (e.g. move particles)
- Compute \(\Delta E=E_{\text{new}} - E_{\text{current}}\) due to perturbation

If \((\Delta E < 0)\) THEN

- Accept perturbation (new system configuration)
Else

- Accept “maybe” with probability \(= \exp(-\Delta E / K_B T)\)

If \(\Delta E\) decreasing over last few T’s THEN:

- \(T = (1-\epsilon)T\) (Cooling schedule)
Else Frozen=True

Return final configuration as low E solution

As SA progresses the system approaches ground state \((T=0)\)
Genetic Algorithms (GAs)

1. Start global minima search with GA:
   a) Establish numerical range for conformation variables (e.g. Angles, torsions, etc.)
   b) Divide range by $2^{(n-1)}$ intervals ($n \rightarrow$ desired resolution)
   c) Associate each slot with a configuration value (real) or use it to binary code the value (bin)
   d) Generate binary representations of the individual variables stochastically, and combine them in a single binary string (position in string indicates parameter associated)
   e) Generate a whole population
   f) Calculate $E$ for each state and determine conformation fit number
   g) Optimize (adapt)
      • Generate new fitter offspring set (populations) by exchanging bits from parents - crossover
      • To avoid suboptimal solution (local minima) induce “mutations” by inverting randomly selected bits (every so often)

2. When GA pop. stops improving, switch to CG

Implementations: Gromos, CMDF
### EM: Convergence criteria

- **Exact location of minima and saddle points** are rare in real molecular modeling applications - so minimization would continue indefinitely.

- **Need to terminate the process!**
  - Energy difference between successive iterations
  - RMS coordinates (change in successive conformations)
  - **RMS of gradient** (recommended)

<table>
<thead>
<tr>
<th>N_iter</th>
<th>N_func</th>
<th>Energy(kcal/mol)</th>
<th>Energy_diff</th>
<th>RMS_coords</th>
<th>RMS_force</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>6.5678589836e+02</td>
<td>0.0000000000e+00</td>
<td>0.0000000000e+00</td>
<td>2.0609801246e+01</td>
</tr>
<tr>
<td>10</td>
<td>21</td>
<td>3.2995541076e+02</td>
<td>-6.8361848793e-04</td>
<td>6.3709030512e-05</td>
<td>1.6146775176e+00</td>
</tr>
<tr>
<td>20</td>
<td>42</td>
<td>3.2181491846e+02</td>
<td>-4.0985875833e-02</td>
<td>7.9228816836e-04</td>
<td>6.8419176497e-01</td>
</tr>
<tr>
<td>30</td>
<td>62</td>
<td>3.1691279419e+02</td>
<td>-1.3937305058e-02</td>
<td>5.2899529600e-04</td>
<td>7.0901724185e-01</td>
</tr>
<tr>
<td>40</td>
<td>82</td>
<td>3.1430948951e+02</td>
<td>-1.4732246578e-02</td>
<td>6.4474518504e-04</td>
<td>6.0462436581e-01</td>
</tr>
<tr>
<td>50</td>
<td>102</td>
<td>3.1251982226e+02</td>
<td>-7.6510768008e-04</td>
<td>1.6604142969e-04</td>
<td>4.9186883205e-01</td>
</tr>
<tr>
<td>60</td>
<td>122</td>
<td>3.1094106944e+02</td>
<td>-1.9026408838e-03</td>
<td>3.2189517910e-04</td>
<td>4.8629507807e-01</td>
</tr>
<tr>
<td>70</td>
<td>142</td>
<td>3.0969619774e+02</td>
<td>3.1197365510e-04</td>
<td>1.1929026568e-03</td>
<td>3.2359649746e-01</td>
</tr>
<tr>
<td>80</td>
<td>162</td>
<td>3.0863355352e+02</td>
<td>-1.318847153e-03</td>
<td>2.7051239481e-04</td>
<td>3.5778184685e-01</td>
</tr>
<tr>
<td>90</td>
<td>182</td>
<td>3.0774713732e+02</td>
<td>9.9227506212e-06</td>
<td>8.0820179633e-05</td>
<td>3.3577681153e-01</td>
</tr>
<tr>
<td>99</td>
<td>201</td>
<td>3.0715578558e+02</td>
<td>-2.4492005913e-03</td>
<td>2.6292140614e-04</td>
<td>3.9285171778e-01</td>
</tr>
</tbody>
</table>

Total energy = 3.0715578558e+02
nflag = 1, total N_iterations = 99, total N_function_calls = 201

\[ \text{RMS} = \sqrt{\frac{g^T \cdot g}{3N}} \]
MM: Transition Structures and Reaction Pathways

- From a chemical process we require:
  - Thermodynamics (relative stability of species) -> minimum points on PES
  - Kinetics (rate of conversion from one structure to another) -> nature of PES away from minimum points (e.g. path between 2 minima = “reaction” pathway).
  - Example: Gas-phase reaction between chloride ion (Cl\(^-\)) and methyl chloride (CH\(_3\)Cl).
    - As the chloride ion approaches the methyl chloride along the C-Cl bond the E passes through an ion-dipole complex which is at a minimum.
    - The energy then rises to a max at the pentagonal transition state.

Adapted from Chandrasekhar J, S.F. Smith and W.L. Jorgensen. JACS, 107, 1985
Example function
• \( f(x,y) = x^4 + 4x^2y^2 - 2x^2 + 2y^2 \)
• \( f'(x,y) = [(4x^2 + 8y^2 - 4)x, (8x^2 + 4)y] \)
• \( f' = 0 \) at \((1,0);(0,0);(-1,0)\)

\[
\frac{\partial f}{\partial \mathbf{x}} = 0 \rightarrow \{(0,0),(1,0),(-1,0)\}
\]

Saddle point

\[
H(x,y) = \begin{bmatrix}
12x^2 + 8y^2 - 4 & 16xy \\
16xy & 8x^2 + 4
\end{bmatrix}
\]

\[
H(0,0) = \begin{bmatrix}
-4 & 0 \\
0 & 4
\end{bmatrix}
\]

\[
H(1,0) = H(-1,0) = \begin{bmatrix}
8 & 0 \\
0 & 4
\end{bmatrix}
\]

Both eigenvalues are positive: \((1,0)\) and \((-1,0)\) are a minima

One positive and one negative eigenvalue: \((0,0)\) is a saddle point
Summary

- Energy Minimization and Conformational Analysis (use FF)
- Transition structures and Reaction pathways
  - distinguishing minima, maxima and saddle points
- Demos with Lingraf, CMDF and LAMMPS
MM: Recap and Highlights

- Each particle assigned: radius (vdW), polarizability, and constant net charge (generally derived from quantum calculations and/or experiment)
- Interactions pre-assigned to specific sets of atoms.
- Bonded interactions are conventionally treated as "springs" (equilibrium -> experimental or QM)
- Interactions determine the spatial distribution of atom-like particles and their energies.
- PES leads to Force Fields (FF)
- Which FF to use depends on:
  - Type of bond modeled (E.g. metallic, covalent, ionic, etc)
  - Desired precision (E.g. Chemistry vs. Statistical Mechanics)
  - Desired transferability (E.g. Describe multiple bond types)
  - Size and time of system simulation
  - Available computational resources
- structures of isolated molecules can lead to misleading conclusions (full interactions MUST be considered, e.g. solvent)
Related Reading Material and Molecular Simulation Codes

• Books and Manuals
    – Chapters: 4, and 5
    – Chapters: 4 and 6
• Polygraph (Reference Manual, Appendix G, Force Fields)
• Software:
  • LAMMPS (http://lammps.sandia.gov/)
  • Lingraf (https://wiki.wag.caltech.edu/twiki/bin/view/Main/LingrafPage)
  • CMDF (https://wiki.wag.caltech.edu/twiki/bin/view/CMDF/WebHome)
  • Cerius2 (http://www.accelrys.com/products/cerius2/).
Energy minimization is used to determine stable states for a molecular structure. Using Lingraf, minimize a tripeptide (e.g. Glutathione or a Thyrotropin-releasing hormone) to an RMS force of 0.01 molecule with:

- Steepest Descent (v145)
- Conjugate Gradient (v200)
- Fletcher-Powell Conjugate Gradient (v200)
- Annealed Dynamics with: 1 annealing cycle using standard microcanonical dynamics per cycle, and a temperature profile starting at 100K and ending at 0K. Minimize after annealed dynamics.

Write down results for:

- Number of minimization steps
- Converged (if) Energy Value

Explain your results