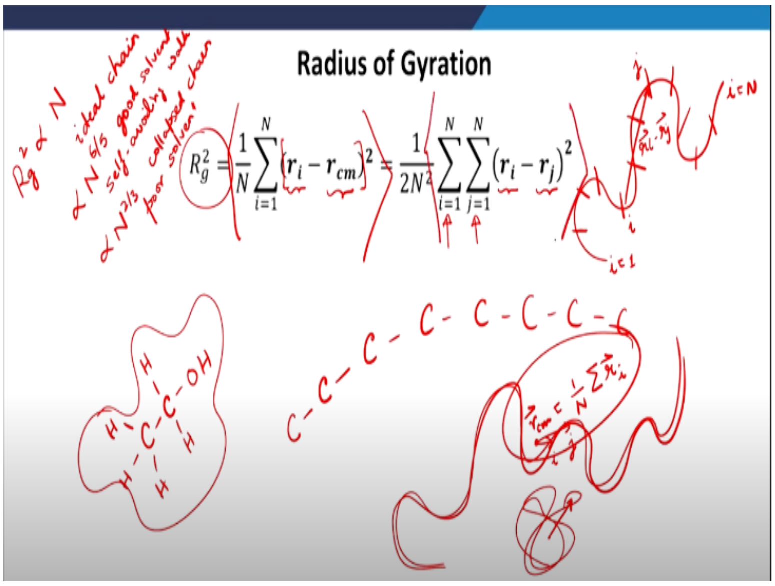
Advanced Thermodynamics and Molecular Simulations
Prof. Prateek Kumar Jha
Department of Chemical Engineering
Indian Institute of Technology, Roorkee

Lecture – 48
MD Simulations – Analysis of Simulation Trajectories (Continued), Case Studies I

Hello all of you. So, the last lecture we have been discussing the analysis tools for MD simulation results. I will continue from that point in today's lecture and then we will come to the case studies, where I will show you the applications of MD simulations. So, the next thing that we can evaluate from the MD simulation is the radius of gyration. So, this particular quantity is important if I want to assess the size of a molecule.

(Refer Slide Time: 00:57)



Radius of Gyration

$$R_g^2 = \left\langle \frac{1}{N}\sum_{i=1}^{N}(r_i - r_{cm})^2 \right\rangle = \frac{1}{2N^2}\left\langle \sum_{i=1}^{N}\sum_{j=1}^{N}(r_i - r_j)^2 \right\rangle$$

$R_g^2 \propto N$ ideal chain
$\propto N^{6/5}$ good solvent self-avoiding walk
$\propto N^{2/3}$ poor solvent collapsed chain
j
$i=N$
$\vec{r}_i - \vec{r}_j$
i
$i=1$
H
C
C
OH
H
H
H
H
C - C - C - C - C - C - C - C - C
$r_{cm} = \frac{1}{N}\sum \vec{r}_i$

So, let us say if I am looking at for example a simple molecule let us say ethanol. This molecule is not clearly of any standard shape like a spherical shape or something else. In fact, if I look at the molecule itself what you will see is, some electron cloud around the molecule and that is going to be somewhat arbitrarily same and the same thing if I extend to larger molecules it is becoming even more complicated more than that the molecule may also change its conformations.

So, one of the possibilities is for example a double bonded molecule may go from cis state to trans state and therefore, we need to have some mechanism to understand what is a change in the shape of the molecule and for larger molecules, let us say proteins or macromolecules or polymers the molecule can also change its conformation in many more possible ways. Let us say for example if I look at a long polymer chain, containing many-many carbons and of course connected with hydrogens and other functional groups this molecule if you look at it will look as something like some sort of a curve and depending on the solvent depending on the temperature this curve that represents the molecule is going to change its shape and R_g therefore is a useful measure as it provides some measure of the size of the molecule.

So, first we will look at the size and then we will look at the same. So, for the size we can use the R_g and size here does not quite represent the length of the polymer chain but it represents in some way the space that is occupied by the polymer chain and the way it is done is we locate the center of mass of the molecule and the center of mass may be located within the molecule or outside the molecule but essentially what you are doing is you are simply taking an average of positions of different units within the molecule. It can be different carbons or it can be whatever functional block you have in the molecule and if you average over that take an arithmetic mean what you get is the center of mass coordinate.

And then you basically measure the distance of different units from this center of mass and now if you square it and sum over all the molecules and then divide by the number of such units what you get is a measure of the size of the polymer chain in this case or any other molecule that you have and this size is going to depend on the conformation.

The same molecule with the same number of carbons and other atoms can for example change conformation to like this and in that case clearly the R_g is going to be smaller because the units in the molecule will be closer to its center of mass.

So, there are two formulas we can use for the center of mass calculation. The first is I can compute the center of mass itself and then compute basically r_i - r_{cm} for every unit in the molecule and there are various ways we can define the units as I have said. It can be every carbon or it can be every monomeric unit or it can be a collection of monomeric units that we can define as a building block and then once we have done that then we can square it and then divide by the number of repeat units after doing the sum of the squares.

The same quantity can also be written as basically two sums where I am summing over all the units with index i and then I sum over again the same units with index j and essentially we compute the distance between the two units on the molecule.

$$R_g^2 = \frac{1}{N}\sum_{i=1}^{N}(r_i - r_{cm})^2 = \frac{1}{2N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(r_i - r_j\right)^2$$

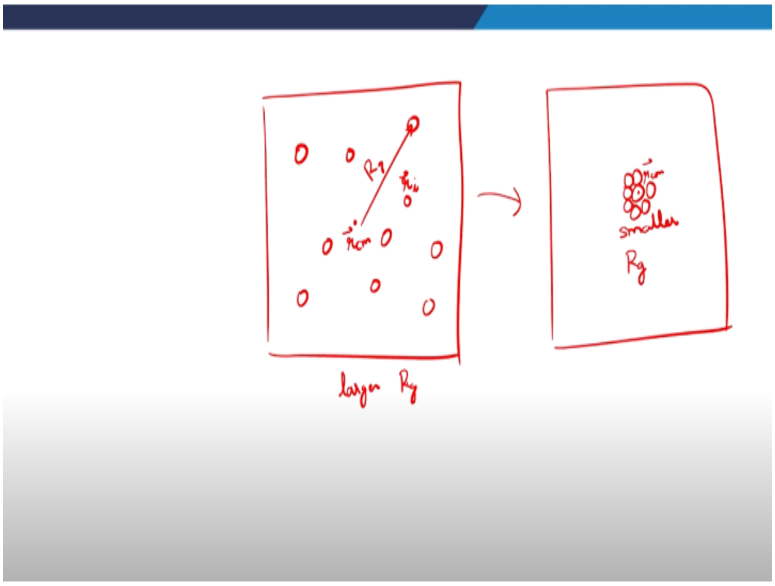
So, let us say for example we start from say i = 1 to i = n here. So, we have some n units along the chain and this can be my ith unit this can be my jth unit and I am basically summing over all the possibilities. So, essentially I am summing over all the possible pairs of units and I compute the distance between these two that is the vector distance r_i - r_j. But when I square it what I have is actually a scalar distance squared and then sum over it and when both of them you can see will give you the same answer that is the magnitude of the R_g. So, this is actually square of R_g.

So, we have to take a square root of that and there are theories in place particularly for the case of polymeric systems that tell you how R_g squared should change with n under different conditions for different polymer chains. One of that is for example R_g squared is proportional to n for what is known as an ideal chain and R_g square goes like something like $N^{6/5}$ for what is known as a self-avoiding walk, where the units tend to avoid each other or it goes like $N^{2/3}$ for what is known as the poor solvent or collapsed chain. Self-avoiding walk is also referred to as the good solvent and essentially you can change the conformation of the polymer chain by varying the solvent so the same polymer chain can show all these different kinds of scaling depending on the solvent and also depending on the temperature. And therefore computation of the R_g^2 will give you the size of the particular configuration that we are dealing with.

And clearly as we are doing the MD simulation or a Monte Carlo simulation the R_g^2will not remain constant. It is going to fluctuate and therefore we are always concerned about the average of these quantities over many-many trajectories and we have discussed that we have to sample so frequently that is the samples must be independent and the sampling time should be more than the autocorrelation time.

So, we have to really do all that stuff whenever we are doing any kind of an average in molecular dynamics we have to ensure that the quantity we are averaging are referring to uncorrelated samples clearly it may not be always possible or it may not be always done that I compute the autocorrelation time and figure out the sampling time. But as a good practice we choose a sampling frequency that is larger in comparison to the autocorrelation time and this we can do for some preliminary simulations done on the same molecules or similar types of molecules or previous literature data on the similar kinds of molecules can provide information regarding what should be my sampling time. This R_g information is also useful when for example we are not dealing with a single molecule but we are dealing with for example the aggregation phenomena of molecules.

(Refer Slide Time: 09:02)

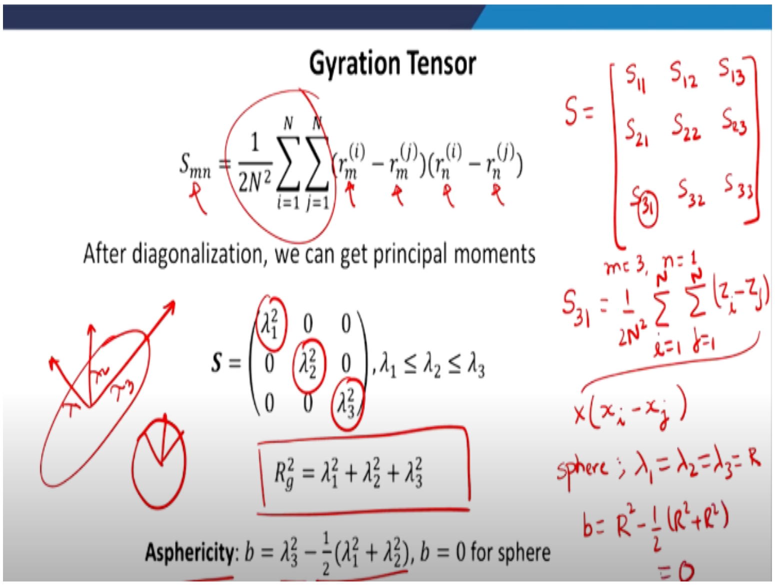


So, let us say for example, you have some molecules comprising a system and they have some attraction between the pairs of molecules. So, in these cases provided that the concentration is large enough and provided that attraction is much larger in comparison to thermal energy, because if thermal energy is dominant the system will tend to go to an ideal gas like state. But if attraction is higher than thermal energy and if the concentrations are reasonably large in that case they will tend to form some sort of an aggregate.

And now instead of looking at the R_g of a single molecule we can look at the R_g of the entire system. And in this case we naturally will find that R_g is going to be large because the molecules will occupy the entire volume you can see that by defining a center of mass and clearly the positions of the molecules. The R_i are farther from the center of mass in the left case then compared to the right case in here all the molecules will be closer to the center of mass so in this case you will have a smaller R_g in comparison to this case where you have the larger R_g.

So, not only a molecule size if I do it over the entire system the Rg can also contain information regarding aggregates that are forming when the R_g is decreasing, that means that we are having some kind of aggregation happening in the system.

(Refer Slide Time: 10:49)



Gyration Tensor

$$S_{mn} = \frac{1}{2N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(r_m^{(i)} - r_m^{(j)}\right)\left(r_n^{(i)} - r_n^{(j)}\right)$$

After diagonalization, we can get principal moments

$$S = \begin{pmatrix} \lambda_1^2 & 0 & 0 \\ 0 & \lambda_2^2 & 0 \\ 0 & 0 & \lambda_3^2 \end{pmatrix}, \lambda_1 \le \lambda_2 \le \lambda_3$$

$$\boxed{R_g^2 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2}$$

Asphericity: $b = \lambda_3^2 - \frac{1}{2}(\lambda_1^2 + \lambda_2^2)$, $b = 0$ for sphere

$$S = \begin{bmatrix} S_{11} & S_{12} & S_{13} \\ S_{21} & S_{22} & S_{23} \\ S_{31} & S_{32} & S_{33} \end{bmatrix}$$

$m = 3,\ n = 1$

$$S_{31} = \frac{1}{2N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}(z_i - z_j) \times (x_i - x_j)$$

sphere ; $\lambda_1 = \lambda_2 = \lambda_3 = R$

$$b = R^2 - \frac{1}{2}(R^2 + R^2) = 0$$

So, apart from the radius of gyration what we can also compute is known as a gyration tensor that is actually a generalization of the idea of radius of gyration in the tensor kind of a space. So, what we do here is now I again do the summation over all i and all j as I did in the formula, the second formula right here. But now I basically look at a particular component of the position vectors. So, here I am looking at the mth component of r_i and mth component of r_j and then I am multiplying with the nth component of r_i and nth component of r_j and that gives me basically a matrix and the m and n component of that matrix.

$$S_{mn} = \frac{1}{2N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(r_m^{(i)} - r_m^{(j)}\right)\left(r_n^{(i)} - r_n^{(j)}\right)$$

Just to give you an example let us say for example we are working in three dimensional space. So, in that case you will have a three dimensional matrix and the index that you see here is basically my m and n index.

$$S = \begin{bmatrix} S_{11} & S_{12} & S_{13} \\ S_{21} & S_{22} & S_{23} \\ S_{31} & S_{32} & S_{33} \end{bmatrix}$$

Let us say for example S 31 corresponds to the case where m is equal to 3 and n is equal to 1 and therefore it is 1 by 2N square i equal to 1 to n, j equal to 1 to n and then I am looking at r_m that is the third component that is a z component. So, it will be z of i minus z of j multiplied with the r_n that is the n equal to 1. So, it is x component multiplied with x of i minus x of j essentially what we get. So, this is by the way the product with that this gives me the z x component of the matrix or the 3 1 component of the matrix.

$$S_{31} = \frac{1}{2N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}(z_i - z_j)x(x_i - x_j)$$

And therefore this represents not only the size but also gives me basically the size in three dimensions, how it is changing in the three dimensions in fact if we diagonalize this as a matrix what we essentially get are the principal components that is every molecule I can define in a way where you have a principal axis along which the dimensions are maximum and then two other axes along which the dimensions are lesser. If it is a spherical particle then all the three axes will have the same dimension. If it is a non-spherical object then you will have one of these components higher than the other components.

$$S = \begin{pmatrix} \lambda_1^2 & 0 & 0 \\ 0 & \lambda_2^2 & 0 \\ 0 & 0 & \lambda_3^2 \end{pmatrix}, \lambda_1 \le \lambda_2 \le \lambda_3$$

So, after diagonalization we get essentially the three principal movements that correspond to the dimensions along the three principal axes where I have defined in a manner that λ3 is higher than or equal to λ2 higher than equal to λ1 and now for this case again I can define the R_g^2 as sum over the squares of the principal components.

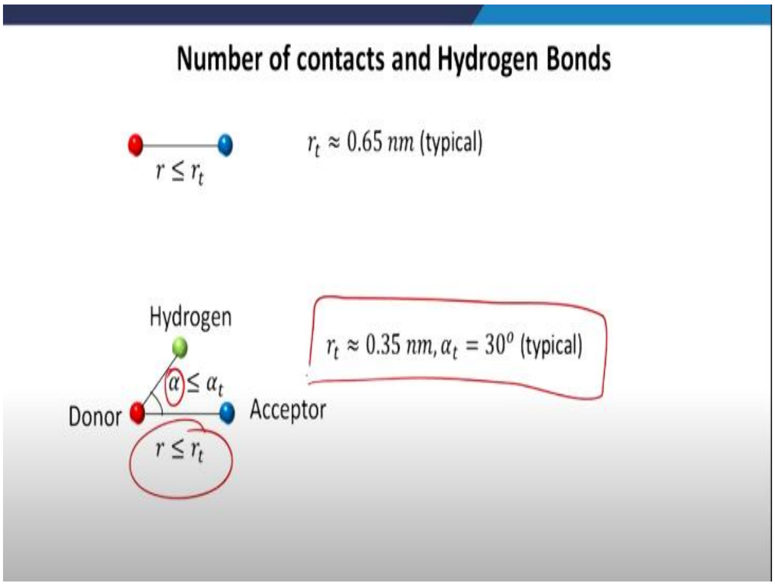
$$R_g^2 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2$$

But I can also define a quantity let us call the asphericity that is defined as-

$$Asphericity, b = \lambda_3^2 - \frac{1}{2}(\lambda_1^2 + \lambda_2^2), b = 0\, for\ sphere$$

Again for the case of sphere all these components are going to be the same because for the sphere no matter what direction I look at the size is R. So, it is λ1 = λ2 = λ3 = R and therefore for that case we will get R^2 - half of R^2 + R^2 and that gives me 0. And any deviation from 0 tells me that we are not forming any sphere and the larger the magnitude of the b is or it can be negative as well so the larger the magnitude is it tells me that we are departing from the sphere.

(Refer Slide Time: 15:31)



Number of contacts and Hydrogen Bonds

$r_t \approx 0.65\ nm$ (typical)

$r_t \approx 0.35\ nm, \alpha_t = 30^o$ (typical)

So, next thing that we can compute and that is again particularly useful when there is some aggregation or some bonding happens between the molecules. So, the first thing we can define is something called the number of contacts. So, two particles in the system, may want to come together for whatever force that is present between them. So, we can define some threshold distance and say that if the distance between the two particles is less than or equal to the threshold I will call that a contact. So, as you lower the threshold distance you are looking at contacts which are very close to each other as we increase that we are forming contacts even when the particles are farther apart and depending on the context you may want to use different values of the threshold distance depending on what kind of contacts are forming there but one typical value for example can be say 0.65 nanometer. So, I can look at how many contacts are there from a molecule. So, this is not really any kind of a bond forming between them. I am simply finding whether two particles are coming together within that particular distance then I will call that a contact.

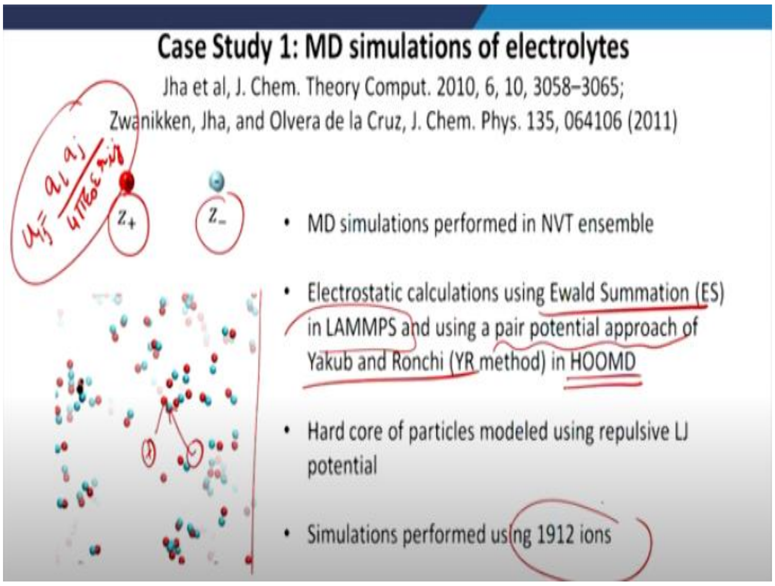
There is not really a bond forming in fact in molecular dynamics since we are not doing chemical reactions no bonds can really form so whatever bonds are there in the system they will remain in the system. There is no bond breaking or anything else happening in the classical molecular dynamics. But nonetheless the atoms may come close together or the molecules may come close together and I can define contacts in this particular way depending on the context that we are in.

So, one particular example of the contact is the hydrogen bond and for the case of hydrogen bond it is not only that the distances are important but also the orientation of the hydrogen atom with respect to the donor and acceptor is also important. So, in that case we can put basically two conditions one for the distance between the donor and acceptor that is again that if distance is less than a threshold distance we will call it a hydrogen bond but in addition to that we can also put an angle condition that is the orientation condition and for example here the acceptor donor hydrogen angle is α and that is assumed to be less than equal to αt.

So, we can add conditions of the spacing and we can also add conditions for the orientation and typically the values that are used in MD to characterize hydrogen bonding is something like 0.35 nanometer for the distance threshold and 30 degrees for the angle threshold.

So, with this background now I want to discuss some case studies taken from my own research to illustrate the application of molecular dynamics. These are only very limited examples of what MD can do. But I have taken from my own research because I have the complete information regarding how the simulations have been conducted and if required we also have the input files that you can borrow from me.

(Refer Slide Time: 18:45)



Case Study 1: MD simulations of electrolytes

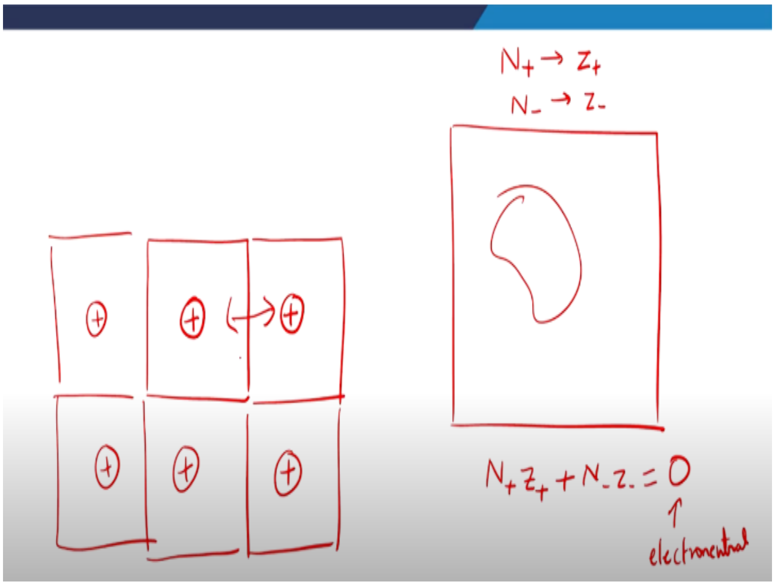
Jha et al, J. Chem. Theory Comput. 2010, 6, 10, 3058–3065;
Zwanikken, Jha, and Olvera de la Cruz, J. Chem. Phys. 135, 064106 (2011)

- MD simulations performed in NVT ensemble
- Electrostatic calculations using Ewald Summation (ES) in LAMMPS and using a pair potential approach of Yakub and Ronchi (YR method) in HOOMD
- Hard core of particles modeled using repulsive LJ potential
- Simulations performed using 1912 ions

So, the first example that I will start with is the MD simulation of simple electrolytes. So, what the system is consisting of is essentially positive and negatively charged particles. So, you can have positive particles of certain valency Z_+ and negative particles of certain valency Z_- and typically they are present in a solution but we are not really modeling solutions in this particular case but they can be modeled.

In this case we are only modeling the interactions between the plus and minus particles using molecular dynamics and of course we have plenty of such ions in the system. As a rule we should make sure that whenever we are doing this kind of charge systems the whole system has to be electro neutral and there is a reason why we do that and the reason is the following.

(Refer Slide Time: 19:39)



So, let us say for example we have some N_+ ions of valency Z_+ and let us say you have N_- ions of valency Z_-. So, I can define the total charge of the system as something like N + Z_+ + N - Z_- and what I am telling you is that we should keep that overall charge of the system to be equal to 0. That is the condition of electro neutrality and that is applying on the large system that I am simulating clearly you will have regions in the system that can be positively charged, regions in there that can be negatively charged that is completely possible. But the overall system has to be electro neutral and there is a strong reason for doing that and that is that since we typically use periodic boundary condition and I am also using in this particular case and actually all the case studies that I will show.

So, in that case what is going to happen is if the electro neutrality was not valid then the system will have some net positive charge or negative charge and then since the images contain the same conformation they will therefore also have some net positive charge actually this of the same magnitude. So, what will then happen is that this simulation box and its images will have electrostatic repulsion between them so they tend to repel each other because both have a net positive charge. So, we know that the like charges repel and that would basically lead to an explosion of the system because you have some repulsion force that is pretty much telling me that the system should try to push the other system apart or the other system may try to push it below.

So, this gives me some artificial force and that is not really present in the system that I am trying to simulate because periodic boundary condition was only introduced as a matter of convenience this is not, I would say a correct representation of the system this has been just done for the sake of convenience and therefore that repulsion between the system and its images should not be present in our simulation because it will give me some artificial repulsion that will give me some artificial force on these walls that should have been avoided and therefore we always ensure that the system overall is electroneutral.

And that is also true in most conditions at equilibrium if the system is carrying an excess charge that means that there is some external field present only then there can be some excess charge in the system when I am talking about at a macroscopic level. So, therefore the condition of electro-neutrality should also be valid for equilibrium systems when there is no external field being applied and therefore to simulate the equilibrium behavior we should by default make sure that my system overall is electro neutral and that is the condition we can ensure by choosing N_+ and N_- accordingly.

So, whatever number of particles we are simulating that is my N equal to $N_+ + N_-$.those number of particles should be distributed between the positive and negative particles in a manner that the system overall is electroneutral and we always do that in all the simulations that I will discuss.

So, this is the first thing here. So, in this case I am showing you one snapshot of my MD simulation. So, you can see the red particles which are positively charged and the blue particles which are negatively charged and there is nothing else in the system. So, the only interaction between the systems is the interaction between the plus and minus particles and in this case we are introducing two kinds of interaction. One is the standard Coulomb interaction that is given by the energy as-

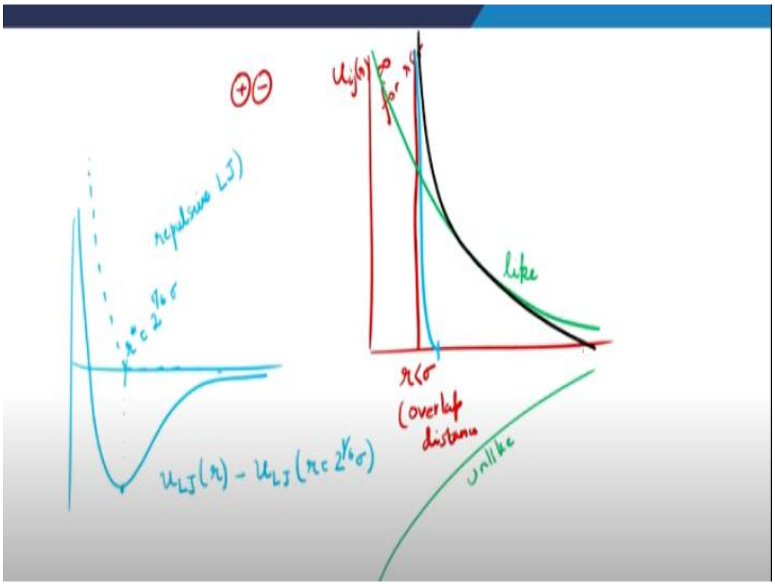
$$U = \frac{q_i q_j}{4\pi\varepsilon_0\varepsilon\, r_{ij}}$$

where r_{ij} is the distance between particles i and j and q_i and q_j refers to two particles.

Now if i and j are both positively charged in that case q_i equal to Z_+ q_j is Z_+. If i and j are both negatively charged then you have qi equal to Z_- q_j is Z_- and when i and j have opposite charges in that case we will have Z_+ multiplied with Z_-. So, by putting that particular force form the simulation automatically takes care of all the pair computations.

So, once we have done that there is something else that remains and that is the positive and negative particles have an attraction. So, they want to come close together but they can only come closer to until a certain point because after that the particles will start to overlap and that overlap has to be prevented. So, therefore in addition to the Coulomb interactions we also add some hard core interactions that will prevent basically the overlap of particles.

(Refer Slide Time: 25:35)



Now I have been telling you that if I want to really put the hard code in like a very strict manner that is if I say that my U_{ij} (r) for r less than some σ or the overlap distance goes like this that is it goes to infinity for r less than σ this will give me discontinuity in the forces. So, therefore I use a soft core and our method to construct the soft core is such that it really goes to zero at very short distance from σ.

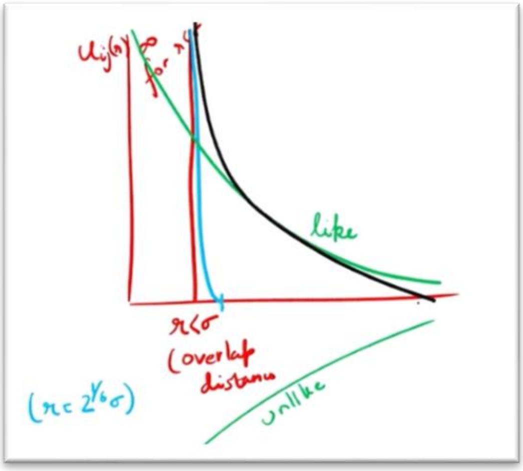
So, in principle although this allows for overlaps but then as soon as the distance becomes less than σ the energy shoots really fast. And that means that the energies are very high, it is not infinity but energies are still very high and they increase suddenly as soon as we get to the overlap distance. So, the way to construct this kind of a potential is to use the Lennard Jones form that we have discussed in a very smart manner. So, I already have said that my Lennard Jones potential is given by something like this.

Now I do not want the attraction part. I only want the repulsion part. So, I know and we have shown that the minima happens at something like r* is equal to $2^{1/6}$ σ. So, what we can do then is shift this entire potential at r* to the value of zero. So, the whole thing is shifted in a manner that the potential becomes 0 at r* and this can be done by doing something like-

$$U_{LJ}(r) - U_{LJ}(r = 2^{\frac{1}{6}}\,\sigma)$$

and this is what is known as a repulsive LJ potency.

And clearly beyond r* we say that the interactions are zero. The repulsive LJ interactions are zero. We can still have Coulomb interactions. So, Coulomb interactions can be for example like this for like charged particles or it can be like that for unlike charged particles that is attraction for unlike charges and repulsion for like charges. So, we are basically adding the coulomb interactions with the Lennard Jones interactions. So, together we will have some form of like charge example can be something like that.



So, at longer distances it behaves like pretty much the standard Coulomb interaction. But at shorter distances there is a correction because of the repulsive Lennard Jones interaction and that takes care of the fact that two particles cannot really overlap. If we would have not done that then all the plus minus would have sat on each other because it is then the r is equal to zero and at r equal to zero for unlike charges the energy is minus infinity. And we want to go towards negative energy so the system will pretty much collapse into basically a point where all the plus and minus come together this should be really avoided and therefore we add the repulsive Lennard Jones potential.

So, I have also told you that doing this pure Coulomb interaction is actually hard in molecular dynamics because not only we have to look at the particle - particle interactions but we also have to look at the particle - image interactions or the image - image interactions and the reason for that is that these interactions are not really short range so we cannot cut them off. If we cut them off then we make large errors. The tail correction actually diverges and therefore we need to have some smarter mechanism to handle this long range interactions and we have already discussed briefly that we can use this Ewald Summation method that is we divide the electrostatic energy into two parts. One is a short range part that I handle in real space and one is the long range part that I handle in Fourier space.

So, in this particular paper what we have done is that we have used the Ewald Summation to do molecular dynamics in software called LAMMPS. But then we were also interested in using GPUs for molecular dynamics and it turns out that Ewald Summation schemes in GPU becomes somewhat complicated and the reason is because the Ewald Summation expressions are pretty cumbersome and the memory of the GPUs are somewhat limited than in comparison to CPUs. So, although Ewald Summation is the method of choice whenever we are doing molecular dynamics particularly on CPUs. In fact what we use is the particle mesh Ewald scheme that is also basically a version of Ewald Summation but somewhat more efficient but the same schemes in GPUs are somewhat more cumbersome at least at the time when we did this work.

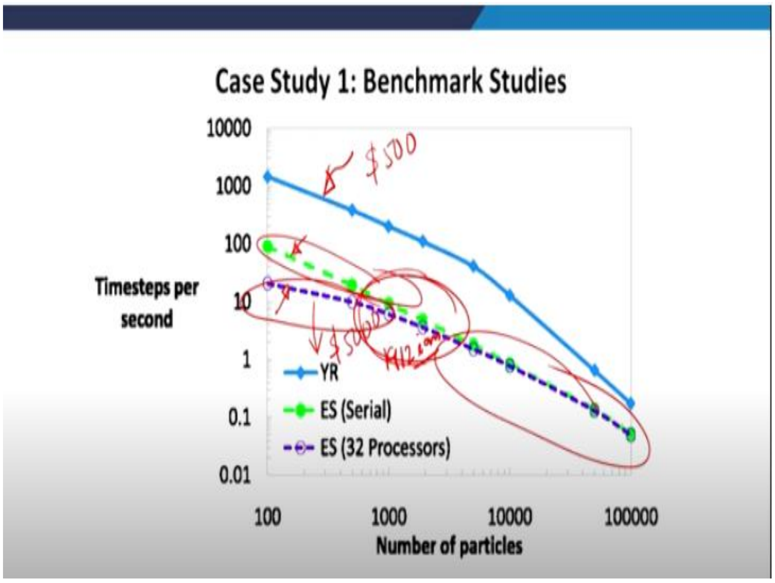
Now of course the GPUs also are getting better. So, how long this is going to be valid remains to be seen. So, what we did is we implemented another method that is based on the idea of pair potentials. I will not go into details of the method. You can look at the paper but the basic idea was something like this. So, the real space part of the Ewald remains as it is but the Fourier space part was basically averaged along the θ direction that is we assumed that that term is somewhat isotropic.

So, basically did an isotropic averaging of the Fourier space part and using that we have been able to write this particular Ewald Summation in the form of pair potential and that pair potential has to be computed only between the particle with other particles or the minimum image of the other particle we do not have to worry about the particle with other images or images with other images and all that. So, this was one smart scheme that we used this due to scientist named Yakub and Ronchi and we refer that as YR method and we implemented that in software called HOOMD that is I would say one of the best softwares to do molecular dynamics on GPUs

And then we did simulations for like 1900 ions now this number has to be chosen to be like large enough such that it gets the averages correct. At the same time it has to be small enough that we can simulate within reasonable time and we figured that using these many ions gives me a good value of the averages that we can work with. The fluctuations in quantities are relatively small now clearly this number is not something that is standard we really have to work with different number of particles at least in preliminary runs to make sure that the results will not be affected if I double the number of particles or make the particles 10 times. Because keep in mind that we are trying to actually mimic a real system that contains like moles of ions. So, 1900 ions in this case is only a small representation of that system. So, if for example the simulation using 1900 ions is very different from say double of that or 10 times of that then in that case we cannot really rely on our results. So, we really have to make sure that there is no dependence of system size on the results that we try to compute and we definitely have ensured that in this particular work.

So, the next thing that we did is I have been telling you that whenever you are working with HPC platforms we have to check how exactly this code scales as we increase the number of processors. And what I have been telling you is that as we increase the number of processors our efficiency of parallel simulation actually improves and this is kind of demonstrated here so for this particular case we did simulations for a larger number of particles as well and what you can see is these green lines are basically time steps per second taken by an Ewald Summation on a serial processor and this is what is taken by the Ewald Summation on a parallel processor of course we correct for the number of processors so it is like the total number of time steps taken by all the processors put together.

(Refer Slide Time: 34:44)



Case Study 1: Benchmark Studies

And what we can see clearly is the efficiency of the parallel program actually is worse than in comparison to the serial program in fact there is a huge drop in efficiency as we go from serial to parallel you should not get confused here that the computation time has decreased because we are using more codes the computation times could have been lesser but if I basically add over the computation time of all the processors that quantity reduces.

So, let us say for example if I use two codes and it takes one second I count the time as two seconds as opposed to one seconds and it is that time that I am comparing and that time actually is much worse in comparison to the serial program when we are using a smaller number of particles. But as soon as we go to a larger number of particles that difference pretty much goes away that means simulation starts scaling like if I double the number of codes my computation time basically becomes half that is every code is also taking the same time as it would have taken in a serial code and this is the ideal limit we want to work the parallel simulations in.

In fact somewhere over here where I am working at 900 ions is also not bad we are clearly not as the serial code but since we are using more number of cored the total computation time is less. Clearly if I combine over all the course the time is more but what we are able to do is we can run the simulation in lesser time for ourselves because the cores are running together but the time it takes for us to see the results is clearly given by the total time taken by all the codes put together and that quantity is decreasing. So, this is a regime where we can easily work.

So, as long as the parallel program is not really cutting down on the computing time heavily we are good. But as soon as we are let us say doubling the number of codes and our computation time is decreasing by a little bit in those cases we have to go back to the serial implementation. This is the same thing for the YR method and now you can see for this particular case that we are doing the YR method on GPUs is actually outperforming the Ewald Summation on the CPUs.

Now clearly it depends on the CPUs I am working with and the GPUs I am working with. But just as a matter of comparison, I want to say that the cost of the CPUs if I go to purchase them is much higher than the cost of the GPUs. So, that can be a fair comparison. So, for example this would cost like dollar 500 and this would cost maybe dollar 5000 or even more. So, we are getting a higher efficiency that means we can perform more time steps per second while spending lesser money from our pocket and this is what is an ideal limit that is why we should use the GPUs. It turns out just like in this case the Ewald Summation could not be used for the GPUs there are other limitations as well whenever memory becomes an important factor we have to go to CPU codes. But over time molecular dynamics have been increasingly performed on GPUs because of their large efficiency in comparison to the CPU based codes and it is only getting better over time. This particular study is 10 years back in time. So, it is clearly much better now than that time.

There are alternate strategies where we can combine both CPUs and GPUs. So, some of the part of the code we can assign to the GPU that can effectively and that part has to be requiring less memory. On the other hand the heavier part that requires more memory can be done in CPUs and therefore there are efforts on trying to combine the CPU codes with GPU codes as well and almost every software nowadays has features to give some of the tasks to the GPU.

The only problem with GPU programming is that it does not follow the standard programming we are used to. So, in this particular case we programmed the GPU part and the CUDA language. But the CPU part was using a software that was coded in C or FORTRAN language. So, it is not that the same code will run on the GPU and we can work with that. We have to make I would say rethinking of the code, actually redoing the code because we are working with an altogether different architecture of the hardware that is GPUs as compared to the CPUs.

So, with this particular idea I want to conclude today's lecture. In the next lecture I will carry on with the same example and discuss what results we can infer from this particular simulation, thank you.