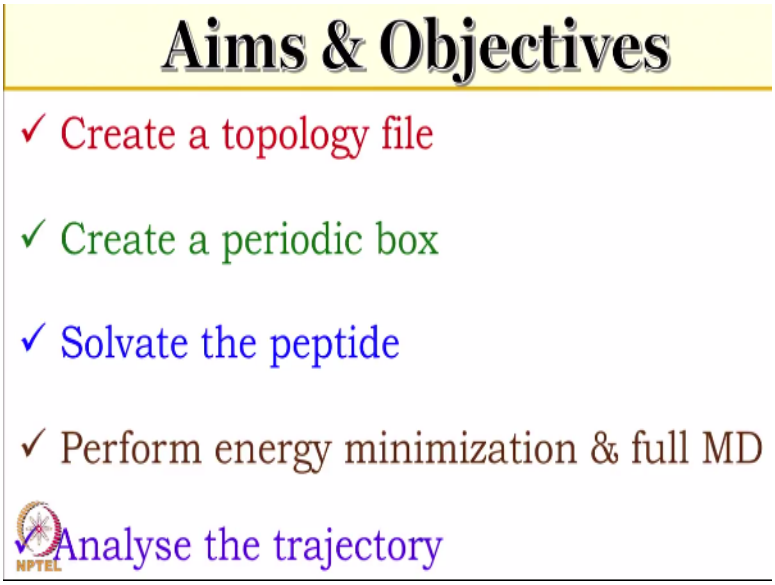
Computational Chemistry & Classical Molecular Dynamics
Prof. B. L. Tembe
Department of Chemistry
Indian Institute of Technology – Bombay

Lecture – 40
Molecular Dynamics Simulations Using GROMACS-8 Simulation of s-peptide

We will resume our discussions on MD simulations. So far what we have done is to describe the MD simulation of a liquid Argon, then we considered simulation on water molecules, then we also considered mixtures of water and alcohol. Today, we will resume our discussion on MD simulations of an s-peptide, s-peptide is a collection of amino acids and Sonanki will describe how to go about this simulation. Can you start? Yeah.

So today we will do the MD simulations on s-peptide. So what we will do here is first we will create a topology file from a pdb file.

(Refer Slide Time: 00:56)



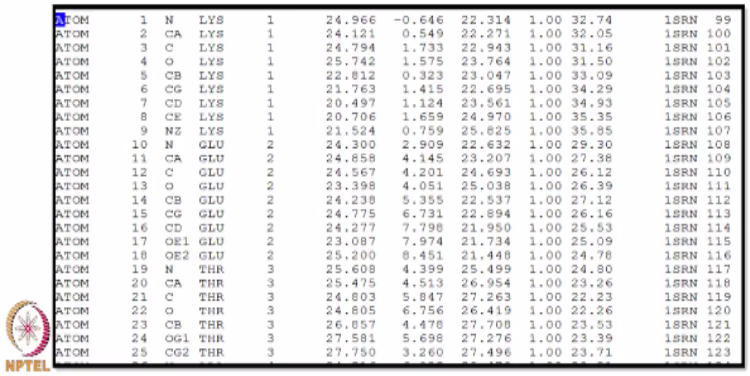
And then we will create a periodic box around the s-peptide, we will also solvate it and then we will perform energy minimization and full molecular dynamics and after performing full molecular dynamics we will try to analyse some of the properties of the s-peptide using the trajectory.

(Refer Slide Time: 01:16)

pdb file of s-peptide

To view the pdb file, open the file using the command vi,

vi speptide.pdb



```
ATOM      1  N   LYS     1      24.966  -0.646  22.314  1.00 32.74      1SRN  99
ATOM      2  CA  LYS     1      24.121   0.549  22.271  1.00 32.05      1SRN 100
ATOM      3  C   LYS     1      24.794   1.733  22.943  1.00 31.16      1SRN 101
ATOM      4  O   LYS     1      25.742   1.575  23.764  1.00 31.50      1SRN 102
ATOM      5  CB  LYS     1      22.812   0.323  23.047  1.00 33.09      1SRN 103
ATOM      6  CG  LYS     1      21.763   1.415  22.695  1.00 34.29      1SRN 104
ATOM      7  CD  LYS     1      20.497   1.124  23.561  1.00 34.93      1SRN 105
ATOM      8  CE  LYS     1      20.706   1.659  24.970  1.00 35.35      1SRN 106
ATOM      9  NZ  LYS     1      21.524   0.759  25.825  1.00 35.85      1SRN 107
ATOM     10  N   GLU     2      24.300   2.909  22.632  1.00 29.30      1SRN 108
ATOM     11  CA  GLU     2      24.858   4.145  23.207  1.00 27.38      1SRN 109
ATOM     12  C   GLU     2      24.567   4.201  24.693  1.00 26.12      1SRN 110
ATOM     13  O   GLU     2      23.398   4.051  25.038  1.00 26.39      1SRN 111
ATOM     14  CB  GLU     2      24.238   5.355  22.537  1.00 27.12      1SRN 112
ATOM     15  CG  GLU     2      24.775   6.731  22.894  1.00 26.16      1SRN 113
ATOM     16  CD  GLU     2      24.277   7.798  21.950  1.00 25.53      1SRN 114
ATOM     17  OE1 GLU     2      23.087   7.974  21.734  1.00 25.09      1SRN 115
ATOM     18  OE2 GLU     2      25.200   8.451  21.448  1.00 24.78      1SRN 116
ATOM     19  N   THR     3      25.608   4.399  25.499  1.00 24.80      1SRN 117
ATOM     20  CA  THR     3      25.475   4.513  26.954  1.00 23.26      1SRN 118
ATOM     21  C   THR     3      24.803   5.847  27.263  1.00 22.23      1SRN 119
ATOM     22  O   THR     3      24.805   6.756  26.419  1.00 22.26      1SRN 120
ATOM     23  CB  THR     3      26.857   4.478  27.708  1.00 23.53      1SRN 121
ATOM     24  OG1 THR     3      27.581   5.698  27.276  1.00 23.39      1SRN 122
ATOM     25  CG2 THR     3      27.750   3.260  27.496  1.00 23.71      1SRN 123
```

So pdb file of s-peptide we first need a pdb file which is the protein data bank file of the peptide. So to view the pdb file we will view it using the command vi. So we have to type vi speptide.pdb and after we type vi speptide.pdb something like this will appear on the screen. So here we will see that in the fourth column there is one about for lysozyme, then there is glutamine, then there is tryptophan, then there are different amino acids right.

So how many sites are there for lys, there are about 6? There are about 8 to 9 sites for lysozyme, then there will be another amino acid glutamine, then there will be alanine. So this s-peptide.pdb they have to download from? The racb site, from the protein data bank site. Into their directory? Yeah, into their directory, they have to download it. Okay. So this is the pdb file of the peptide right.

(Refer Slide Time: 02:30)

Create a topology file

A topology file can be created with the *pdb2gmx* program.

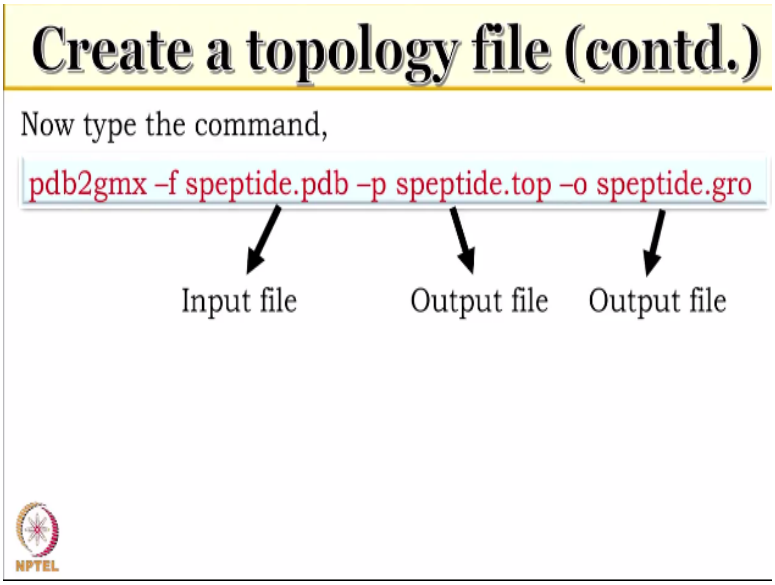
First type, *pdb2gmx –h* to see what it does,

```
Option     Filename  Type         Description
------------------------------------------------------------
  -f       eiwit.pdb  Input        Structure file: gro g96 pdb tpr etc.
  -o       conf.gro  Output       Structure file: gro g96 pdb etc.
  -p       topol.top  Output       Topology file
  -i       posre.itp  Output       Include file for topology
  -n       clean.ndx  Output, Opt. Index file
  -q       clean.pdb  Output, Opt. Structure file: gro g96 pdb etc.

Option       Type   Value   Description
------------------------------------------------------
-[no]h       bool   yes     Print help info and quit
-[no]version bool   no      Print version info and quit
-nice        int    0       Set the nicelevel
-chainsep    enum   id_or_ter  Condition in PDB files when a new chain and
                              molecule_type should be started: id_or_ter,
                              id_and_ter, ter, id or interactive
-ff          string select  Force field, interactive by default. Use -h for
                              information.
-water       enum   select  Water model to use: select, none, spc, spce,
                              tip3p, tip4p or tip5p
```

And after viewing the pdb file so we will create a topology file using the pdb file. So to create a topology file we will use the command pdb2gmx right. So first we will type pbd2gmx-h, -h is the help command. So first we will see what the command does. So after you type pbd2gmx-h it will display what are the input file that is required to run this program, what is the output file that we expect from this program.

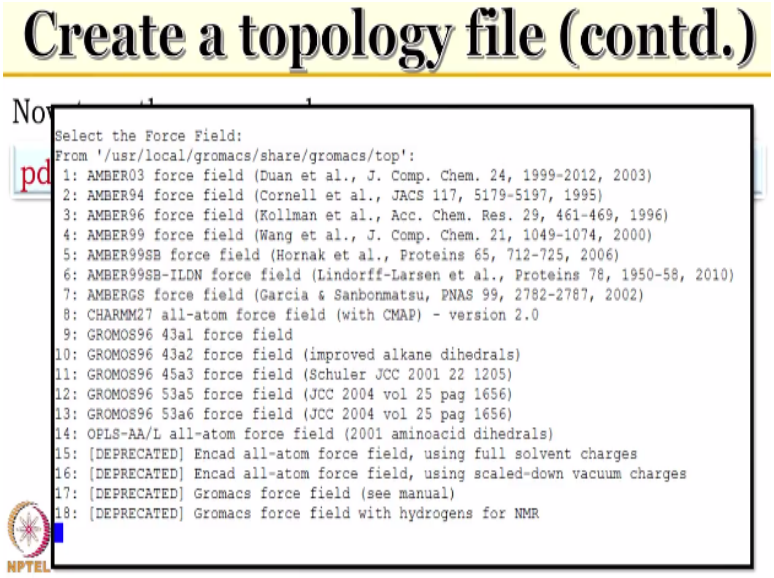
So we see that the input file required for this program is the pdb file which is the s-peptide.pdb which we already have and output file we can get a topology file, we can get a gro file and index file and itp file. So now, what is that clean.pdb? that is a pdb file format, okay, in a different format. Okay, -q that is the option know? Yeah that is the option. Okay.

(Refer Slide Time: 03:27)



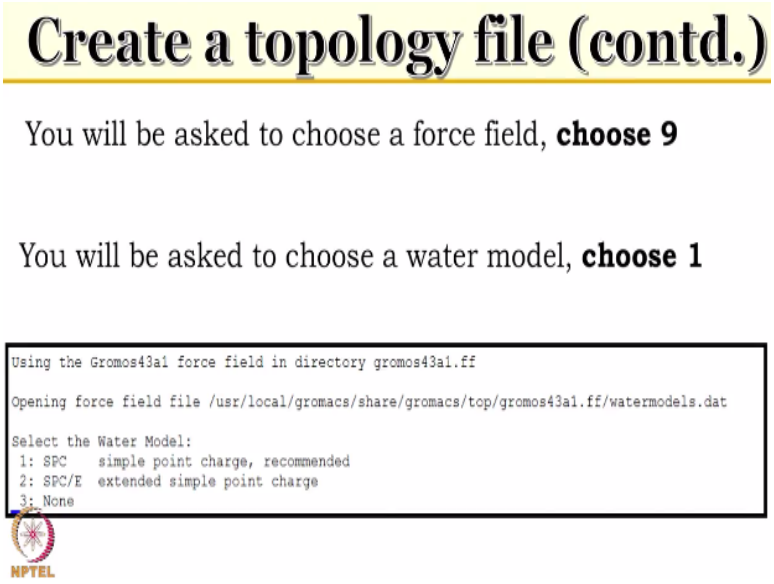
So now we will type the command pdb2gmx space –f speptide.pdb which is my input file then –p speptide.top, top is my topology file for the peptide and –o speptide.gro which is the GROMACS structure file. So your speptide.pdb is my input file and speptide.top and speptide.gro these 2 are my output file.

(Refer Slide Time: 03:55)



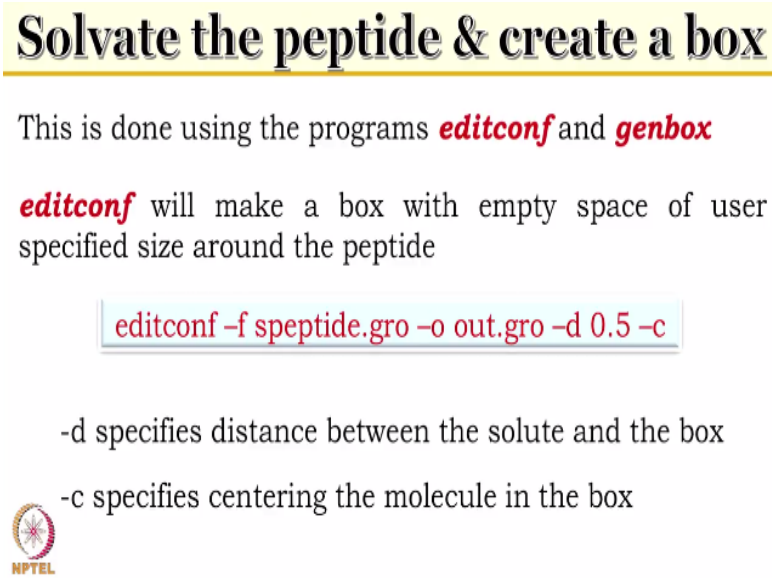
So after I type the command something like this will be displayed in the screen, which will ask me to select a force field from a number of force filed that are already given. There is AMBER, CHARMM, GROMOS, OPLS, there are all kind of force field that are available. So we have to select one force field. So, you just have to give the number of the force filed? Yeah, we just have to give the number of the force field.

(Refer Slide Time: 04:16)



So we choose 9 here so and 9 is the gromos force field, gromos 43a1 force field similarly one can choose different force field as well. So I have chosen the gromos force field and all the references are given there from where these are taken. Yeah, the references are given there, yeah. Okay. So after I choose force field then it will ask me to select a water model. So select the water model. So I choose 1 which is the SPC that is the simple point charge model.

(Refer Slide Time: 04:48)



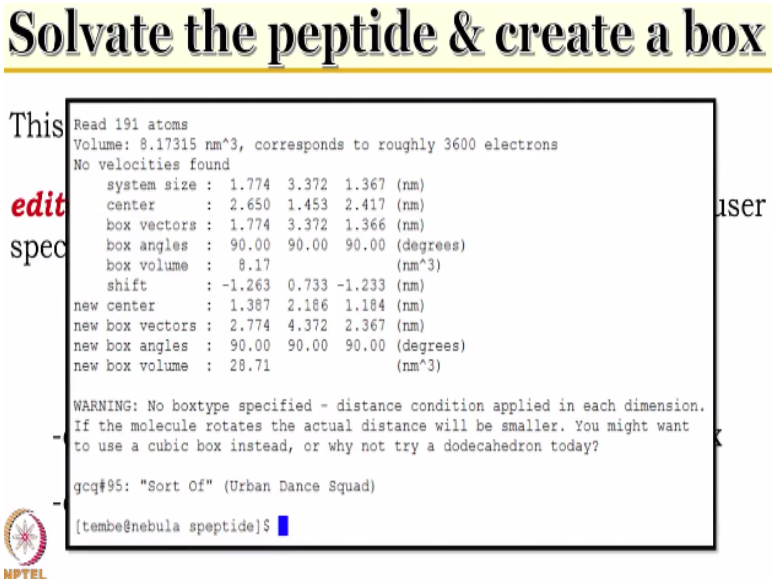
And now I will have my speptide.pdb and speptide.gro file. Now what I have to do I have the pdb file in a box, now I have to solvate the peptide. Now to solvate the peptide and to create a box we will use the command editconf and genbox. So by using editconf it will make a box with the user specified size around the peptide. Whatever the box size that we give, it will create a box around the peptide using that box length.

The coordinates of s-peptide are already assigned? The coordinates are given in the pdb file. So they are already taken in that other files which were the output files? In the gro file yeah, from the pdb file only the gro file is generated, so the coordinate will already be there in the gro file. And it usually puts it in the center of the box? Yeah, that you have to specify. Okay, that you will specify later now?

Yeah, right now I will specify it. So to generate a box around the peptide we will type the command editconf space –f speptide.gro which is my gromos structure file –o, o is the option for output file, out.gro is my output file –d 0.5 space –c. So what is –d? –d is the distance between the solute and the box and –c it specifies centering the solute molecule in the box. So by giving the command –c I am giving the command that my peptide has to be in the center of the box.

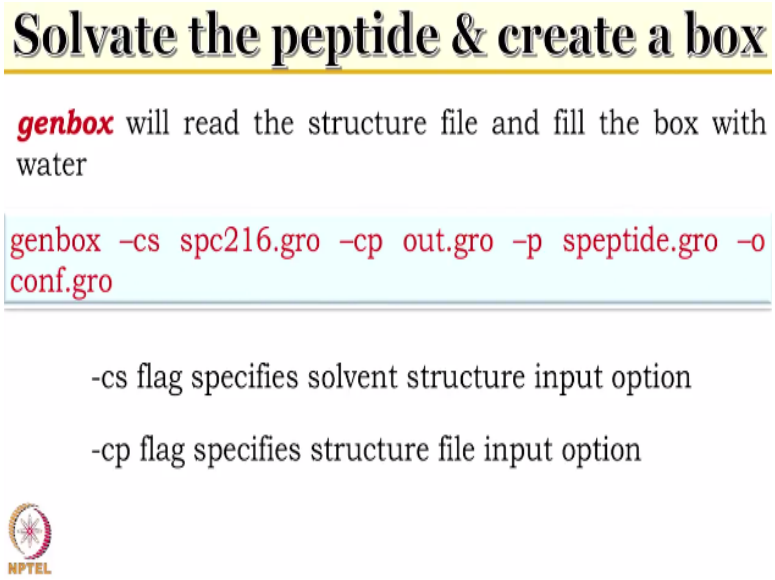
So that minus, instead of –c what are my options, can I give some? Then you have to specify the coordinate, then you have to get the coordinate. Xyz? Yeah xyz, then you have to specify why do you exactly want your peptide to be. Okay. So now that I have my gro file and I have my box around the peptide.

(Refer Slide Time: 06:46)



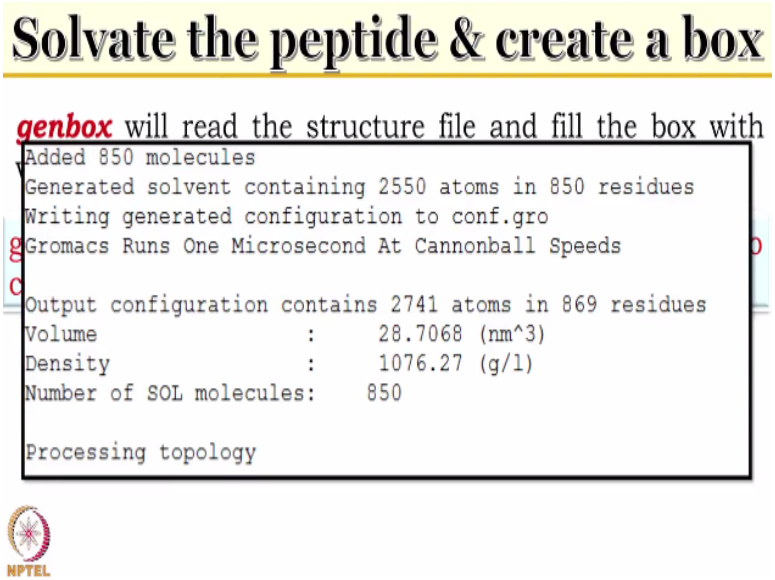
I will now solvate the peptide using the command gen box. So after editconf command you will get something like this where you will have your system size, center of the box, box vector, box angle and all the stuff.

(Refer Slide Time: 06:59)



So now genbox, it will read the structure file that is the out.gro file and it will fill the box with water. So the command is –genbox –cs spc216.gro which is the water model that I will be using to solvate my peptide –cp out.gro, out.gro is the peptide and within the simulation box –p speptide.gro and –o conf.gro. So –cs it specifies the solvent structure input option, –cp is the structure file input option for the solute molecule right.

(Refer Slide Time: 07:40)



So after you give the command something like this will be displayed in the screen where it says that added 850 molecules. So it will add 850 water molecules in the simulation box and it will also modify the topology file because initially when we have generated the topology file using the pdb file only the peptide was there, but now I also have water in my simulation system. So it will add the topology file for water also in the top file.

What was the box length that was given for this calculation? I did not give any box length so it has taken the box length according to the size of the peptide. Okay. I have just specified that the distance between the solute and the box has to be 0.5 and that the solute has to be in the center of the box. Okay. So it will take the box length according to the size of the peptide. Okay, but it has given the volume now. Yeah.

So from that volume the cube root of that volume should give me the length of the box? Yeah. These are all cubic boxes? It may or may not be a cubic boxes. Okay. Because to specify a box type then you have to give –dodecahedron or –cubic. But now it says box angles are 90, 90, 90, so I expect that to be a cubic box? Yeah. Okay. So there is a slight error in this –p speptide instead of gro there will be top.

Because –p is the flag for topol.top option. Okay, so instead of speptide.gro it should be speptide.top. Yeah. On the right, extreme right. Yeah. Okay. So now it will also modify my topology file right.

(Refer Slide Time: 09:19)

genbox also changes the topology file **speptide.top** to include the water molecules in the topology

```
tail speptide.top
```

Something like this will be displayed on the screen

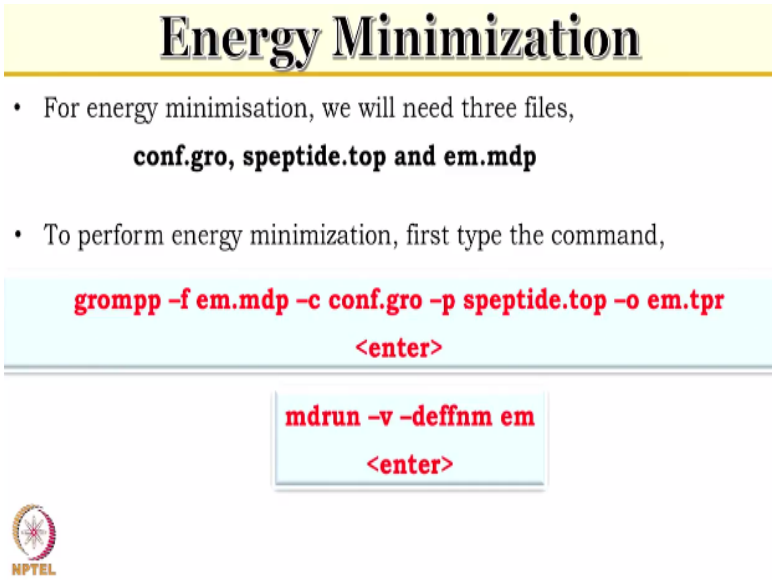
```
[ system ]
; Name
Protein in water

[ molecules ]
; Compound        #mols
Protein             1
SOL               850
```

So if I just type tail speptide.top it will take me to the last few lines of the topology file right and something like this then will be displayed in the screen that there is protein there is one that is my speptide and SOL that is my solvent molecule. There are total 850 solvent. Initially when we look, when we have a look in the speptide.top when we generate it using the command, using only the pdb file of s-peptide.

We will just have protein in the pdb file, in the topology file. We would not have solvent. So this 850 is the number of molecules? Number of solvent molecules that has been added in the simulation box. Okay, then the number of atoms for what will be 3 times 850. Yeah, 3 times 850 and total number of protein. And whatever number of atoms associated with that protein. Yeah, that will be mngro file. Does it display that also, number of atoms in a protein?

Yeah, but then you have to open the gro file to have it. Okay right. So now that we have protein and solvated protein, we have the topology file and so now we will proceed towards the energy minimization of the overall system.

(Refer Slide Time: 10:30)

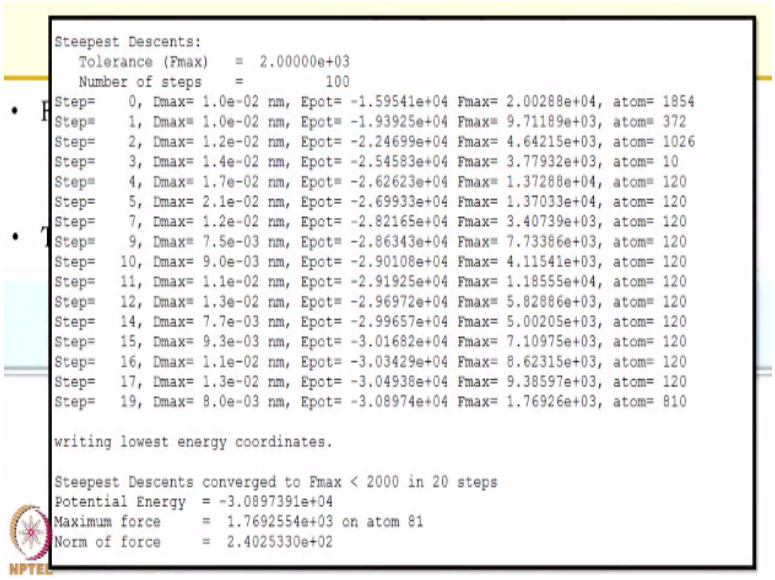


So for energy minimization we will need 3 files, that is the gro file, we need the top file and we need the energy minimization, molecular dynamics parameter file that is the em.mdp file. Now to perform energy minimization we have to first generate a tpr file. So first we will type the command, grompp space –f em.mdp which is my molecular dynamics parameter file for energy minimization then space –c conf.gro which is my gro file including my solvent and the peptide.

Then space –p speptide.top which is the topology file and space –o em.tpr that is my output file. So after generating em.tpr now we will use this tpr file for the mdrun for energy minimization. So when it does the energy minimization, does it tell you how many steps it did or it just? Yeah it will display on the screen that how many steps did it take to perform the energy minimization.

And whatever constraints you have put for the minimization that also it will display? Yeah, it will display the value of force at what step it stopped energy minimization, it will display everything. So next command is mdrun space –v space –deffnm space em. So what –deffnm does, it will generate all my output file that is I will have a different gro file, I will have a tpr file and I will have an xtc file, it will generate all those file using initial as em. Okay.

(Refer Slide Time: 12:09)



So something like this will be displayed in my screen. So as we can see it took 19 steps to perform energy minimization and at each step it has written the potential energy, total force on which atom. So my final potential energy is -3.08 * 10 raise to the power 4 kilo joule per mole. So this is the value. So the maximum force is also written to the right. So initially the maximum force was very large. Yes. Now finally it has become small and it is writing the atom number on which the force is the maximum. Yeah. Okay.

So now since my energy minimization is done so in this case when you have a large molecule like peptide and protein so you have to do a position restrained MD, what it does is a part of the system is not allowed to move, right. So we what we do, we restrain the position of the peptide and we allow the water molecule or the solvent molecule to equilibrate around the protein. So you have to modify the speptide.top that is the topology file accordingly.

(Refer Slide Time: 13:15)

Position Restrained MD

Position restrained MD means Molecular Dynamics in which a part of the system is not allowed to move far off their starting positions. To be able to run with position restraints we must add a section to the **speptide.top** file, describing which atoms are to be restrained. Such a section is already generated by the ***pdb2gmx*** program. In the topology file (**speptide.top**) it looks like

```
; Include Position restraint file
#ifdef POSRES
#include "posre.itp"
#endif
```

So we just have to add 3 lines there that is position restraint.itp file we just have to add this position restrained itp file in the speptide.top file. So in this file all the atom which has to be kept fixed are given there? Yeah, that is the peptide has to be kept fixed and we will allow the water molecules. So are you fixing all the atoms of the peptide or some of them? All the atoms of the peptide. Okay.

But is it possible to do with some atoms fixed, some moving or that is difficult. Yeah, that is also possible then you have to specify those particular atoms in you itp file. Okay. So now to perform position restraint MD we need the gro file from energy minimization.

(Refer Slide Time: 13:58)

Position Restrained MD (contd.)

- For position restrained MD, we will need three files,

  em.gro, speptide.top and pr.mdp

- To perform position restrained MD, first type the command,

```
grompp -f pr.mdp -c em.gro -p speptide.top -o pr.tpr
<enter>
```

```
mdrun -v -deffnm pr
<enter>
```

We need the topology file that is speptide.top and we need position restraint mdp file that is pr.mdp file. So how is that created pr.mdp? You can, it is just same as full.mdp with some

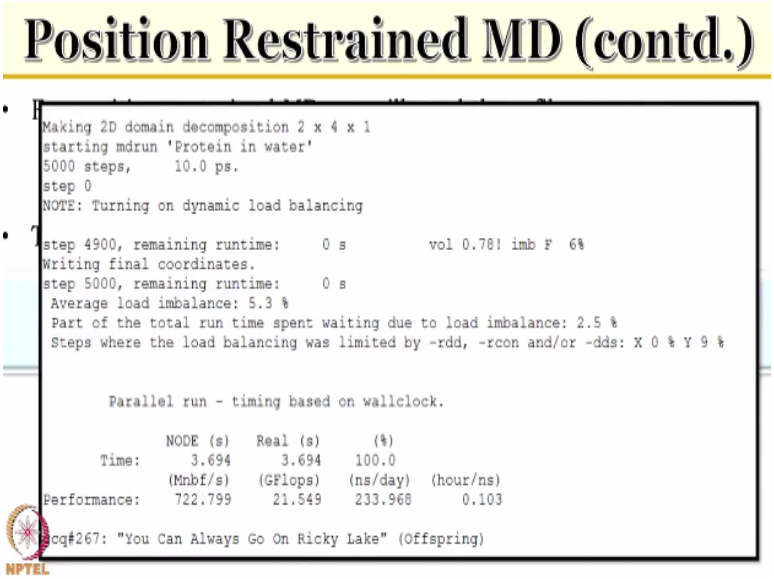
extra constraint added. Okay, but that you have to add manually or is there? Yeah you can add manually even one can download it from GROMACS tutorial sites also. So they have given that pr.mdp files?

Yeah, they have given the format what is the required format for that, okay. So now to do it here also we have to first generate the tpr file. So we will use the command grompp space –f space pr.mdp, pr.mdp is the position restrained molecular dynamics parameter file then space –c, space em.gro, em.gro is my energy minimized configuration and space –p speptide.top which is the topology file.

Then space –o pr.tpr and this tpr file will contain all the information about mdp file, gro file and the top file. Is the order important? You have this –f pr.mdp? No the order is not important. You can give in any order? Yeah the order can anything. So first one can be –o pr.tpr and the last one could be that? Yeah it can be, yeah, the order is not important here. Okay.

Just the flag is important at what flag you are using for what file. Okay. And after you generate the tpr file you have to run the tpr file using the command mdrun. So mdrun space –v space –deffnm space pr. So the only output file that we required from position restraint MD is the gro file that is the pr.gro file. But many others will be created with pr as the starting? Yeah, pr as the starting.

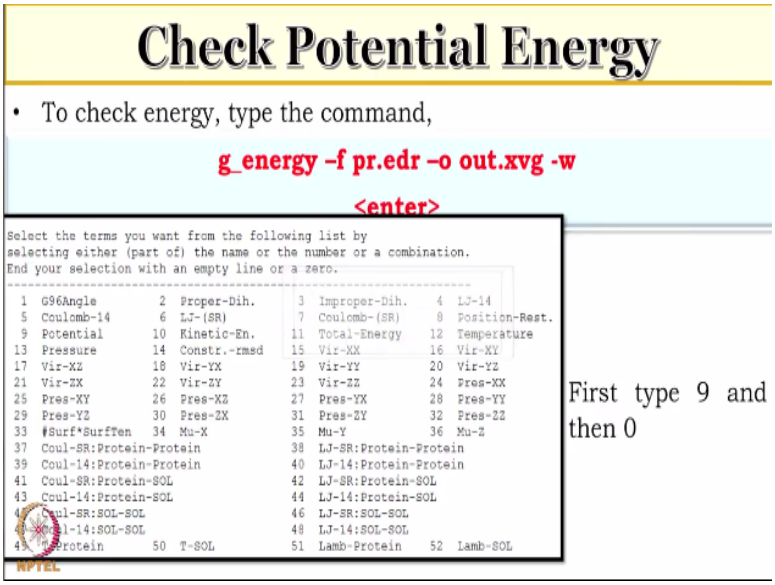
(Refer Slide Time: 15:50)



So something like this will be displayed that I have given total of 5000 steps in my pr.mdp file and my time step is 2 femtoseconds, so total amounting for 10 picosecond simulation. So it will display in the screen that how much time did it take to complete the simulation and all stuff. So once we do it in the terminal then one will know that how it works. Okay, but why is it showing 0 seconds everywhere, step 4900 remaining time?

That is the remaining run time is 0 second. That means it is finished? It is almost finished yeah. Okay, and what is that volume 0.781 on the 6% something? That is how much space will it take in the disc to occupy all the files, output files. So once we have done with position restraint MD, so now we will check the potential energy of the system. So to check the potential energy of the system whether it has minimized or not.

(Refer Slide Time: 16:55)



So we will type the command g_energy space –f space pr.edr, pr.edr is the energy file which we will get by position restraint MD then space –o out.xvg, this out.xvg file will contain the potential energy as a function of time. And what is that last one –w, what is that for? That will display the graph. Okay. And then enter, so after you do this you will be displayed with the number of parameters given here.

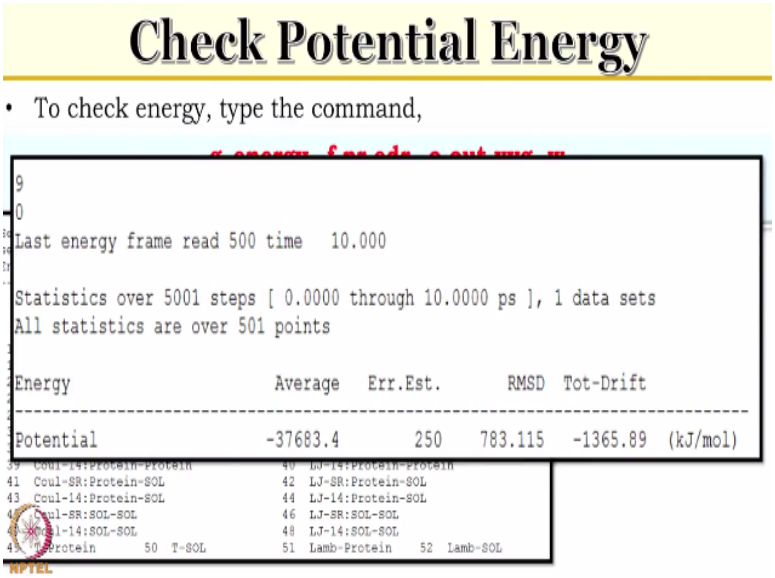
So there is dihedral, there is improper dihedral, then Lennard-Jones, short range, long range, total energy, temperature, pressure, all this energy values you can calculate as the function of time. So what we need here is potential energy. So we see that potential energy is in number 9 right. Okay. So we will first type 9 and then we will type 0. So it will give me the value of potential energy. But where is that, go back, where is that 9 displayed there? Third one.

In the third line, okay. So 9th is the potential energy, 10th is the kinetic energy, 11th is the total energy. One can check this energy values as a function of time. And what are those numbers on the third column? Which numbers? See first one says first one G9 angle. Yeah. The next one is proper dihedral. That is the second one is proper dihedral. What is the 3? Third one is improper dihedral, fourth one is.

So if you type 3 and then 0 it will give you the value of improper dihedral as the function of time. Okay. Suppose if you type 12 and then 0 it will give you the value of temperature as a function of time. Okay. And 4 and 0? 4 and 0 it will give the Lennard-Jones 14 interaction as the function of time. Okay. So this is what it means. But does it also plot those things? You can plot if you want?

Yeah, you have the file out.xvg right. Okay. So if you have out.xvg you just have to type xmgrace out.xvg it will give you the plot. Okay, first you do this type 9 and then 0 then give that xvg. Yeah. Okay.

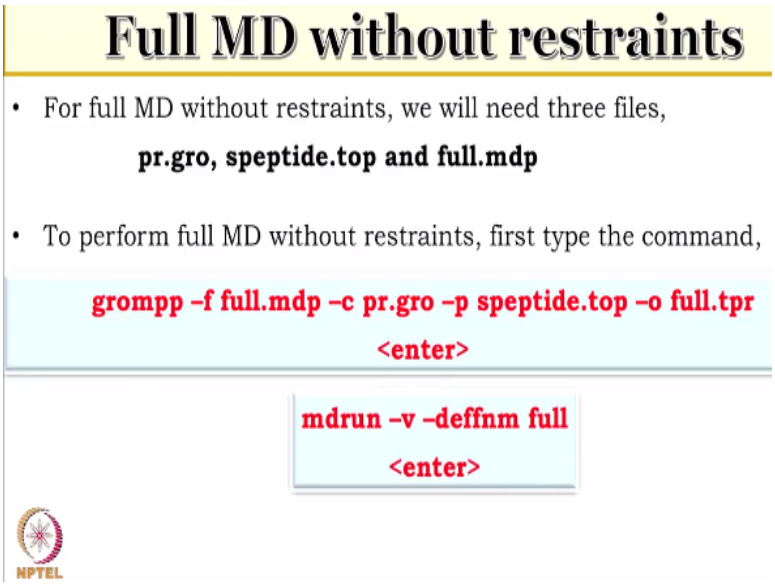
(Refer Slide Time: 18:54)



So after you type 9 and then 0 it will give the value of potential energy. So here you can see that the average value is given -37683.4 kilo joule per mole you will also have the error estimate and the RMSD value as well and what is that total drift? Yeah. RMSD is the total root mean square deviation in the average energy. And since we are just doing it for 10 picosecond so my potential energy is very large -37000.

So the total drift may give you the difference between the maximum and the minimum value. Yeah. Around the average. Yeah, and if we perform simulation for sufficiently larger amount of time we will get a smaller and smaller RMSD. Yeah. Okay.

(Refer Slide Time: 19:39)



So after we check we have done with checking the potential energy now we will perform MD without restraint where we are allowing the peptide to move as well along with the solvent molecules. So what we do here, we need the gro file from the position restraint MD, then we need the topology file and we need the full mdp file. So here also first we will generate the tpr file using the command grompp space –f space full.mdp which is the molecular dynamics parameter file for full molecular dynamics simulation.

Then –c space pr.gro which is my position restraint gro file then –p speptide.top which is the topology file and –o which is my output file full.tpr, this is my output file. So what decides that there are no constraint in this command or it is still with constraint, the present one? That is without restraint, that is in the previous one we have restraint the position of the peptide right and we are just allowing the solvent molecule to equilibrate around the peptide.

But now in the mdp file we will have a change, we will change in the mdp file and then we will allow the peptide to move as well. We would not fix it. So what is it that is changed so that it? In the full.mdp file we have to give a command there. We have to add an extra line there that which are the atom that you want to fix or which are the atom that you do not want to fix.

So there is a difference in the pr.mdp and full.mdp file. So this pr is different from the earlier pr. This is a full.mdp file. Okay. So after my full.tpr file is generated I will use it to run my MD simulation. So I will type the command mdrun space –v space –deffnm space full and then it will give the result of full molecular dynamic simulation.

So I will have trajectory file, I will have tpr trl file and with which I can calculate potential energy, the number of hydrogen bond that the peptide is forming and we can also plot Ramachandran plot which is very important in peptide, which we will show when we perform it in the terminal. So do you have examples of that some of these files here or we will show while demonstrating?

We will show while demonstrating. Okay, and what are some of the difficulties we come across when we do these things? See, this is a large peptide so while you generate a topology file then you have to specify n number of bond length in the topology file to create it. You may or may not have a topology file for a protein or a peptide. So you have to create it manually.

You have to go add the bond length and bond angle in the bonded or non-bonded file and from there you will generate a topology file. So it does give a lot of error while creating a topology file basically and since it is a peptide you have to do a long simulation, 10 picosecond is for just showing purpose that it is performed like, it will give me result easily, but to equilibrate it sufficiently since it is a long, large molecule so you will require long simulation.

So at least say 10 nanoseconds or more than that, yeah at least 10 nanoseconds or more than that and for the MD run how much time. For MD run maybe 15 to 20 nanosecond to get a good result so that you know we can compare it with some experimental value if available, okay. Is there a separate file where the last configurations are saved? Suppose I do not want to do 50 nanosecond in one stretch, I want to do it 25 nanoseconds and 25 nanoseconds, how do I go about?

So what you can do is first 25 nanosecond you have done with the simulation right and then you will have a trajectory file, so you convert your trajectory file into a gro file and the moment you convert it, you will have configuration at each time step in that gro file. So you

can take the last configuration and then you can start simulation with that configuration, the remaining 25 nanosecond.

So the last configurations has to be put in some place which is the first configuration of the previous run. Yeah. So what is the procedure for that? See you will have a trajectory file from your 25 nanoseconds simulation right. So you convert your trajectory file into a gro file that is xtc to gro. One command is there? Yeah, there is a command trjconv, so you convert it into a gro file.

In that gro file you will have configuration at each time step. Okay. Right suppose you have given 2 femtosecond as a time step. So it will write configuration at 0 time step then after 2 femtosecond then after 2 femtosecond till your last time step. So you take the configuration of the last time step and then you start from there, in that file you will have configuration at each and every time step.

Correct, but for the reading do not I need just the last configuration rather than this whole big file? Yeah so what you can do you can create the configuration, you can give a beginning time and you can give an end time. Suppose your beginning trajectory time is 10,000 and your ending time is 10,001 so it will give the one configuration that you need. Okay, and is there a command to find out how much disc space is used?

I have to check it whether there is a command or not. No there is some df or something it tell you how much space is used. That is du space – sh. Right, so I think it is a good idea to keep on finding how much space is remaining because when you keep doing simulations lots and lots of files are created. Yeah, that is true. So one option is to keep separate directories for all the simulations.

And once the simulations are done you just take all the files, remove the file from the directory, or you can save them on a separate hard disk, yeah, and then when you analysing it just take the files which are required for analyzing. Okay, is there a way to compress all the files in a directory? because when you do tar, you do tar know, tar and untar is there a way to, one option is to copy all this on a separate hard disk.

Yeah, even if you tar it or untar it, the file size will be large itself right. So you have to anyway copy and then you have to remove the folder from that directory. So it is best to keep it in a separate directory. Yeah it is best to keep it in a separate directory. So, so far what we have done is that argon simulation then water simulation, mixture simulation and now we have done the s-peptide simulation.

So the next time we will do a free energy of solvation of a molecule is that correct? Yeah. We will do free energy of solvation we will conclude this lecture at this point. So what I would suggest all the users whatever list of commands we have given it is best you write it in a particular file because it is very difficult to remember all these things if you have not practised many times.

So best is to keep a list of all the commands and maybe they should make a directory of some of the errors they have got in the past because that also helps you quite a lot. So once you have these list of commands then you can execute one at a time and then proceed till the end of the thing and what are the commands you have set for plotting xmgrace and then the xvg file.

Right, so using the xm grace and all the xvg files you can plot various distribution functions and analyze your results. We will conclude this lecture at this point. Thank you.