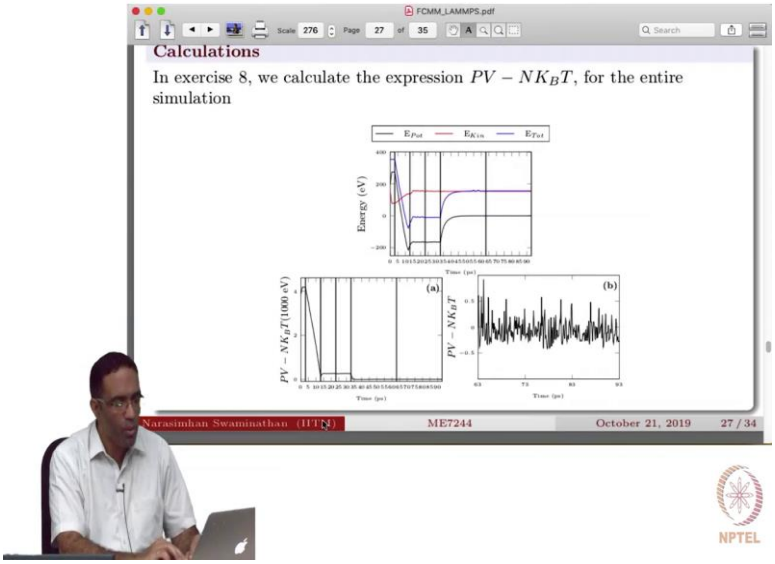
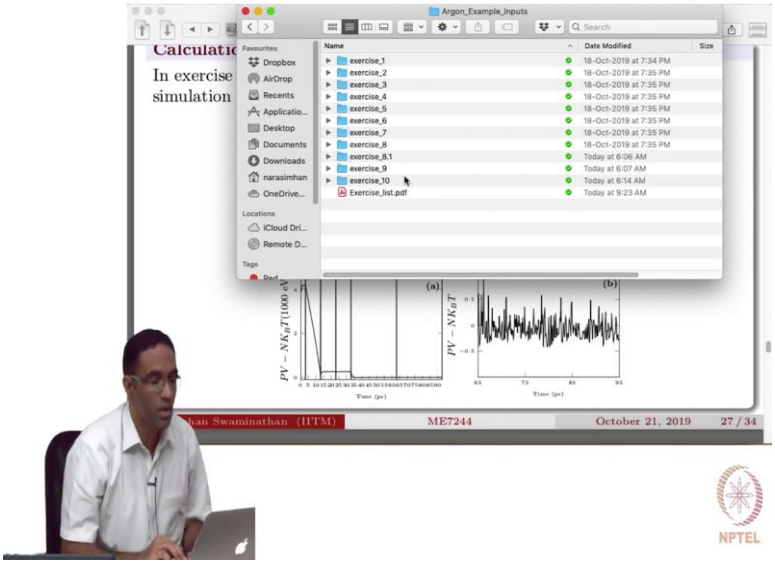
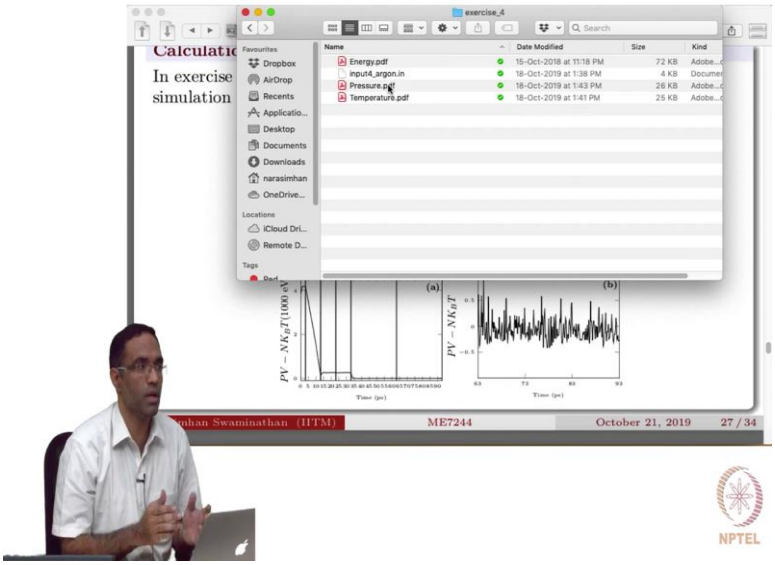
Foundation of Computational Materials Modelling
Professor. Narasimhan Swaminathan
Department of Mechanical Engineering
Indian Institute of Technology Madras
LAMMPS Exercises - 1

(Refer Slide Time: 0:17)

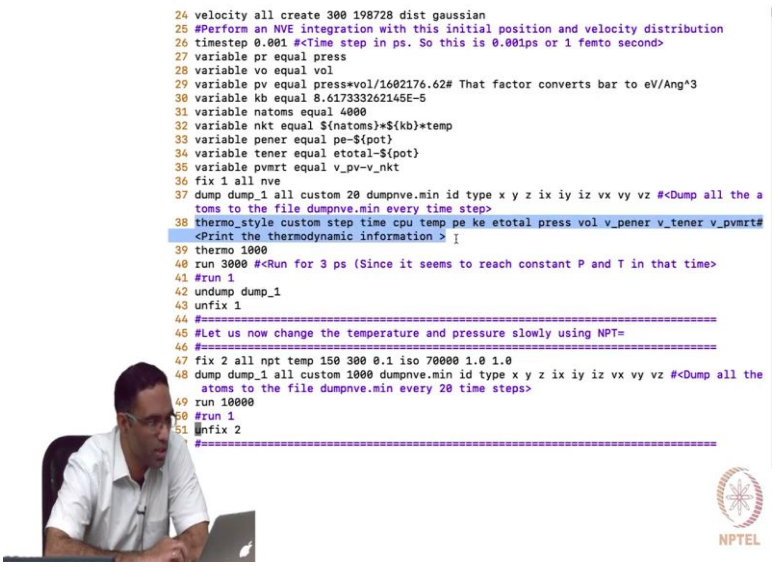






Good afternoon, let us continue from where we left off we were talking about exercise number 8 where we finally were able to equilibrate the gas, argon gas to 1 atmospheric pressure and 300 Kelvin. I have actually, if you have taken a look at your moodle, I have actually put up the entire set of about 10 exercises, a couple of, 2 of which we will be doing now or I will be talking about now. Each of these exercise folders will contain an input script and generally contain additional figures describing the variation of the energy and the temperature and the pressure.

(Refer Slide Time: 1:07)



So one of the things that I wanted to highlight in exercise number 8. I do not know if you have actually seen these exercises if you have gone through these input scripts a lot of useful methods of calculating different quantities are done. So one of the things that I wanted to

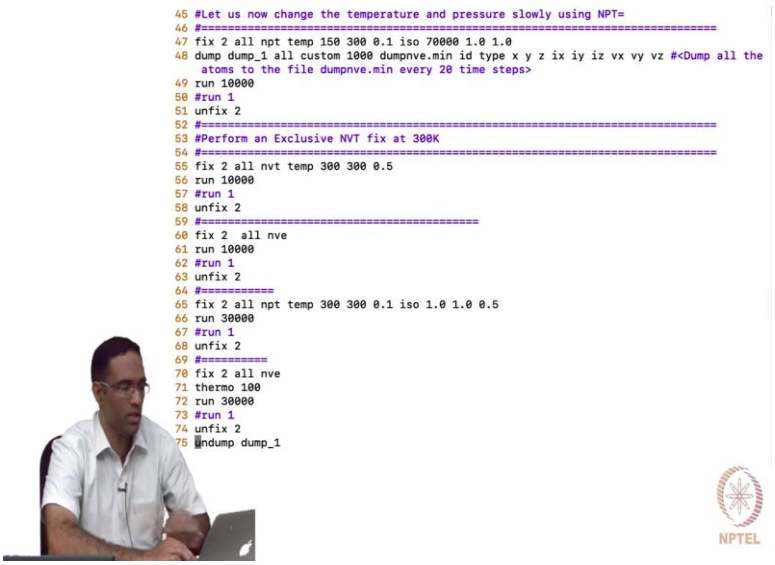
show you is once it becomes ideal gas we want to be able to verify whether the pressure times volume is in fact is equal to n times kb times T. So what I have done here, I should take a look at exercise 8, the current version of it and that is actually right there on the screen right now. I have calculated various quantities right here.

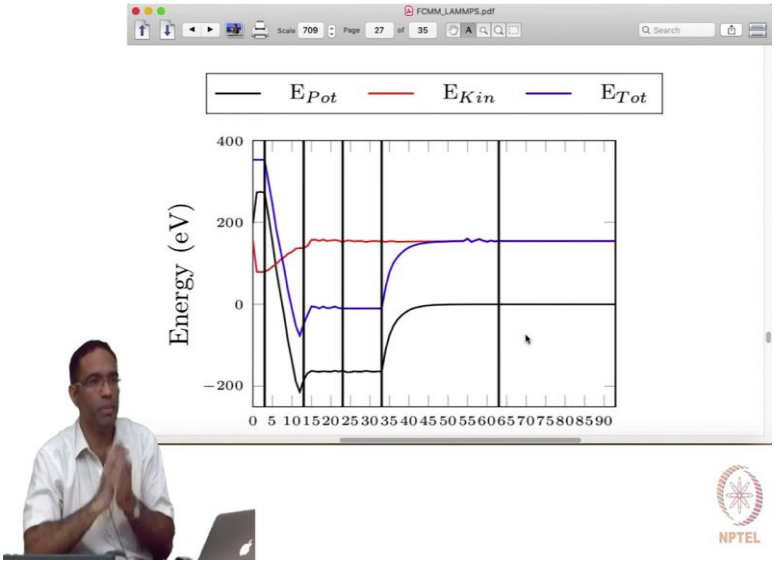
So I have defined a pressure called, I have define a variable called pressure and then I have calculated the value pv and set a pv is equal to pressure times volume divided by some number here which is a factor which converts the pressure from bar to electron volts per angstrom cube. kb is assigned, the number of atoms is assigned and then I am calculating the value of nt which is number of atoms times, the Boltzmann constant times, the temperature. So you see the temperature is actually a variable that is already available in LAMMPS so I do not need to put the dollar sign.

Then I also calculate out other quantities like the potential energy and the kinetic energy here. Potential energy and the total energy both of which are measured with respect to the base potential energy of the system. You know this these details I did not talk about in the class but if you take a look at the document that is there in the folder it should be extremely clear. Then looking at the quantity pv minus mrt. So I am just calculating the variable pv minus variable nkt and I am as the simulation is running here, in the nve ensemble I am actually also printing out that variable here.

In the thermo style, I am printing in addition to the usual step time temperature etcetera. I am also printing variables which are shown as v underscore pener and this pener is nothing but the variable defined here which basically computes the current potential energy minus a base. The potential energy of the system after just after minimization. I just told the potential energy just after the minimization and I am subtracting the current potential energy from that base potential energy. So that I can see how the energy is actually distributed amongst the, energy actually distribute as a function of time of the entire system. And these this quantity is what I want to look at now which is basically the quantity pv minus nrt.

(Refer Slide Time: 4:01)

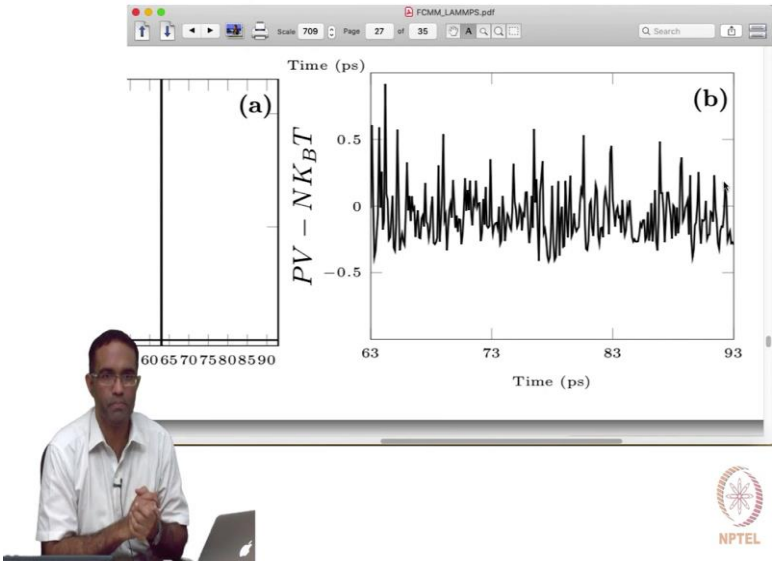




And I have run the entire simulation just like how I did previously. And if you take a look at it, this is how it is going to look. So what is happening here is exercise 8, basically consists of about a total simulation time of about 93 picoseconds, out of which the first 3 picoseconds is simply some nve simulation and then the next 10 picoseconds is my npt simulation making an attempt to change the temperature and the pressure of the system which did not succeed. And then you have another 10 picoseconds where I did some nvt to bring the temperature up to 300 Kelvin, then another set of nve simulations to see if it had indeed reached 300 Kelvin following that I did another npt simulations for a very long time for about 30 picoseconds and during this time something interesting started to happen.

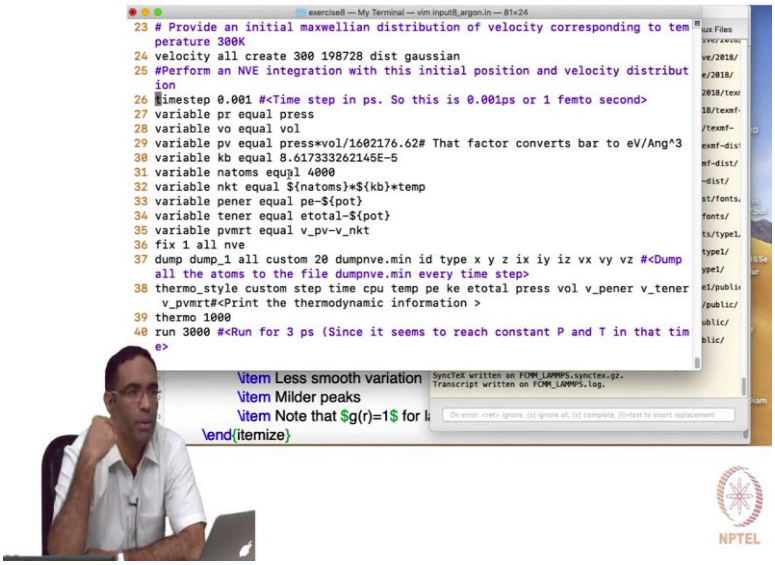
You see that the total energy, which is the blue curve increased, the kinetic energy became equal to the total energy and the potential energy of the system became 0, which means it is like an ideal gas rate, there is absolutely no potential energy in the system and then when I did the remaining nve, the kinetic energy, all the energy is just kinetic energy which is what is expected in the ideal gas and then the potential energy of the system is 0. So it is very clear that starting from a very very configuration there is extremely far away from what we wanted, like, we wanted 300 Kelvin and 1 atmosphere but we started off at some extremely high pressure and we have done several steps to actually bring the system to the state that we required to be in.

(Refer Slide Time: 6:01)



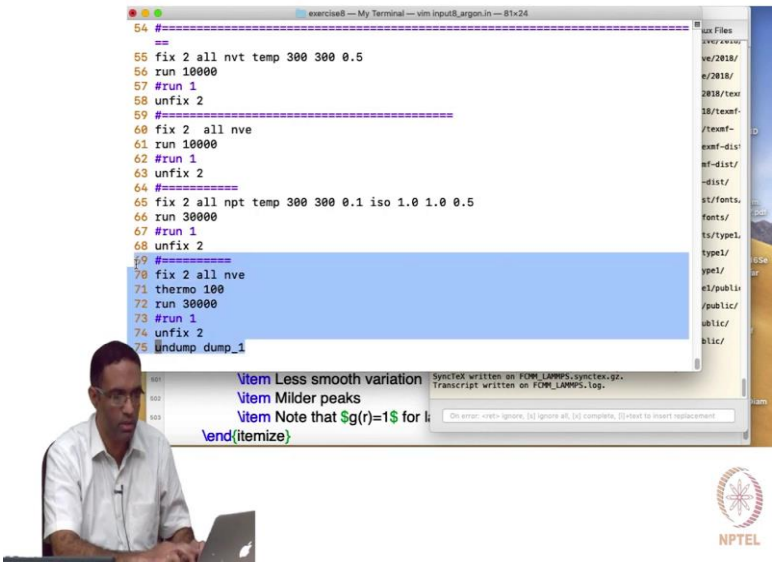
Now I have plotted the quantity PV minus NKBT which is in 1000s of electron volts, again as a function of time over the various chunks of the simulation. So initially it is having really weird values because it is not a gas PV is equal to NRT need not be satisfied, but as you keep changing these things finally I have a value PV minus NKBT oscillating about 0 the zoomed version of this NVE period is actually shown here. You can see that it is oscillating about 0. So PV minus NKBT is calculated and it is satisfied as expected.

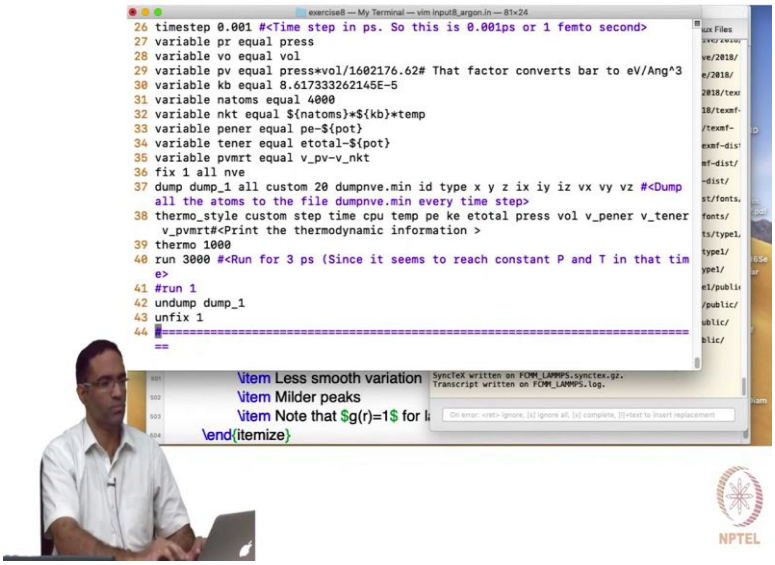
(Refer Slide Time: 6:57)

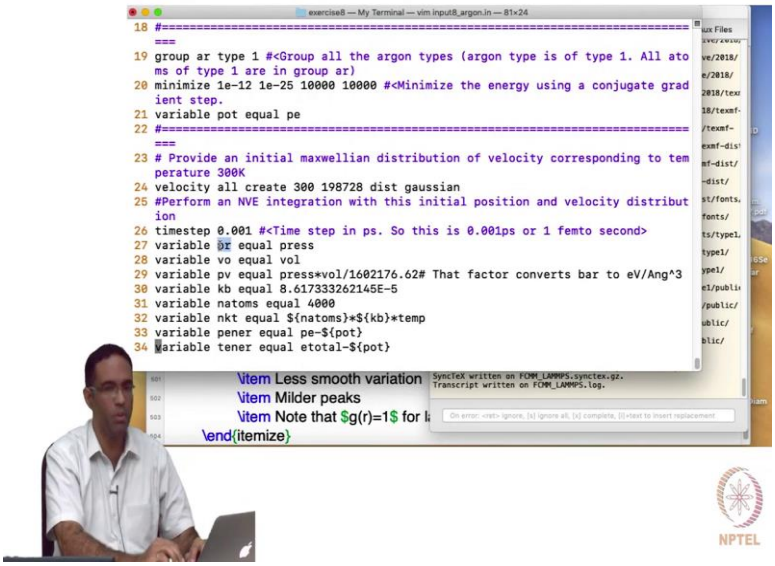


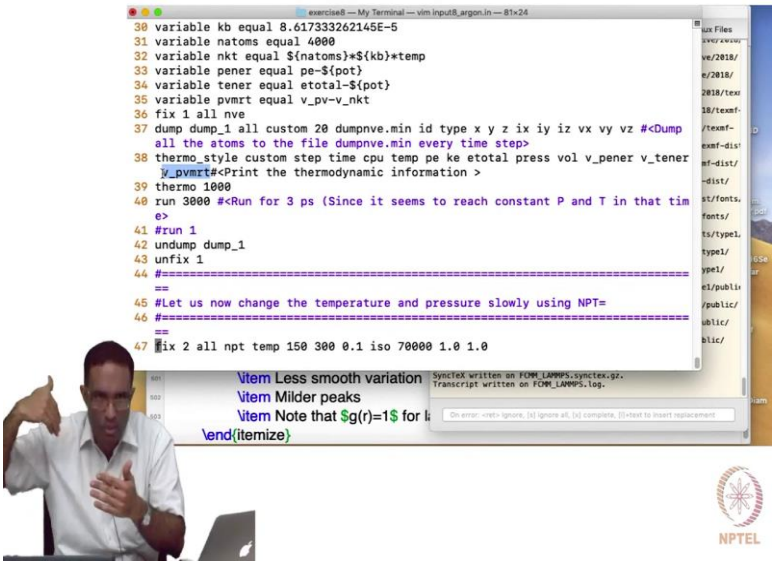
So in this manner, the use of the variables is I have just used it to illustrate PV minus NKBT, but you can imagine using it for various other things. For example, sometimes you might want to calculate every time what you are interested in statistical mechanics are the average quantities. What is the average pressure? What is the average temperature? So you could use these variable commands to actually calculate the average pressure, calculate the average volume and all these things, right here in the script instead of post-processing after you run the simulation. Of course you should be careful about the time because every time you have such variable commands it is going to consume some time to actually do.

(Refer Slide Time: 7:41)









Another thing that I have done here is…

Student: Sir?

Professor: Yes

Student: The NVE simulations is starting after the variables have been…

Professor: Yes, correct, the NVE simulation here, the final one right?

Student: The very first simulation what is that? Is it NVE or something else?

Professor: Yes, yes NVE.

Student: It is starting after the variables have been defined? But lammps execute step by step.

Professor: So there will be, but it is after the minimize and all that, so it can still have the bunch have atoms and it knows how to calculate the pressure from the current values of the positions on the velocities. So it will calculate whatever pressure it wants to calculate and give it to the variable PR. And secondly you should understand that although I have defined a variable here that variable changed. So once I flagged that calculations to be taking place it will continue to take place throughout the simulation. That is why even though pv minus nkt is defined on top.

If I am saying thermo style and ask it print out this information we underscore pv minus mrt it is doing this for the entire simulation. It is just it is flag, does that variable that variable you can imagine, it as that particular calculation has been ticked as on. So it should continue to do it throughout the entire simulation that is the way you look at it.

Student: So if you have to look at flow of control in this particular script then it goes down and then once it sees run, does it do something else?

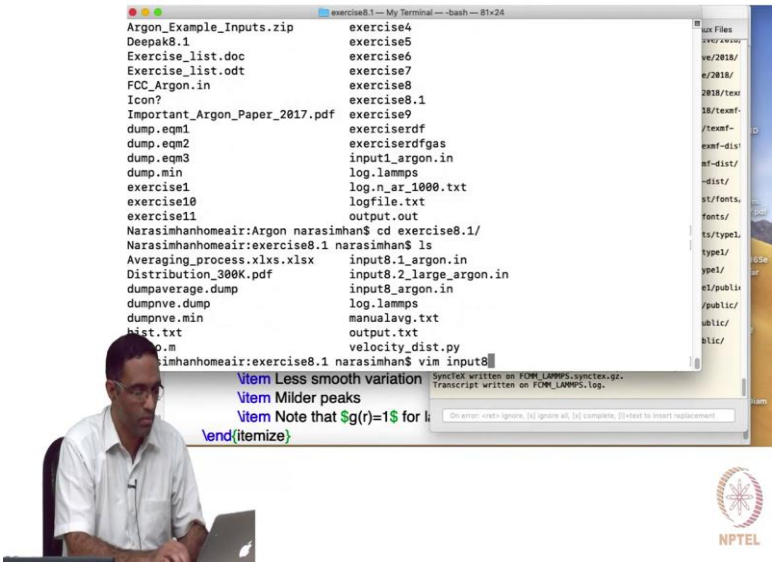
Professor: Whenever it sees run it will run whatever is there above it.

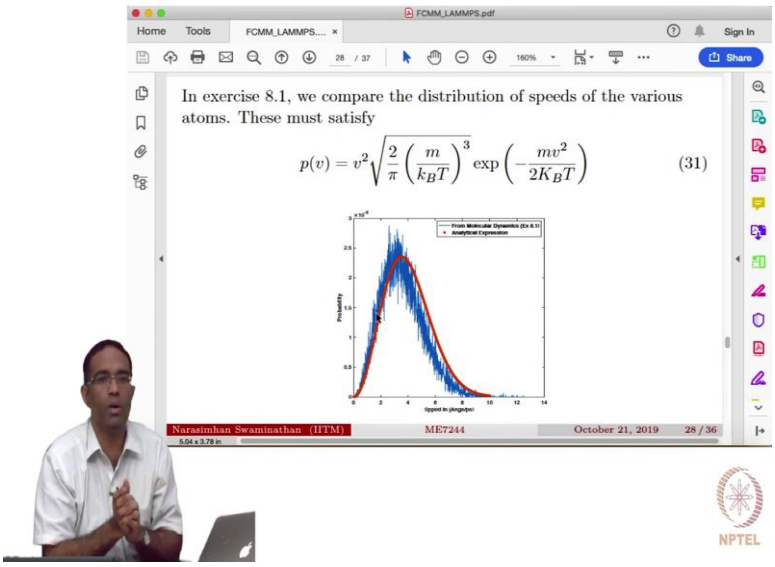
Student: Starting from where?

Professor: No whatever flags have been switched on, are the details that it is going to keep in mind while it is performing that run unless you switch it off explicitly. That is why I always unfix it, stop doing it. So that it kind of switches it off before you before it goes on to the next part of the program. So that is the way it works. Is this okay, definitions? See these definition of variables and all you have to practice and try a little bit. You know, unless you have tried with a simple script, thought about a problem, try to see if I am able to get something that

makes sense, you are not going to understand how this variable actually works. So this is just a simple demonstration.

(Refer Slide Time: 10:28)

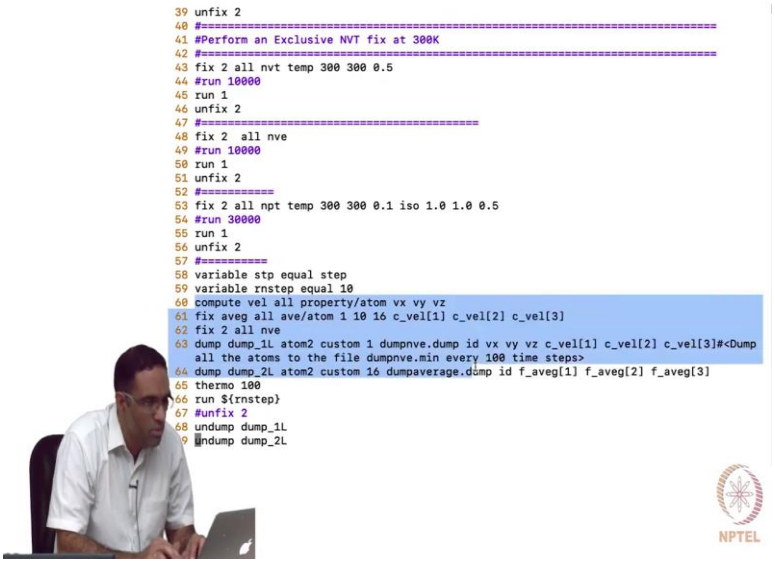




We will see something else now. We will see what is referred to as, another thing that I have done in exercise 8.1 at equilibrium. Suppose we are talking about the system at equilibrium. So we know that at equilibrium the velocities or the speeds of all the atoms in the system must actually have a certain profile, basically it must have a Boltzmann distribution. So this is actually the probability, this expression is actually an analytical expression, probability that your velocity is this and you can show using kinetic theory of gases or using statistical mechanics that this is in fact the expression that turns out.

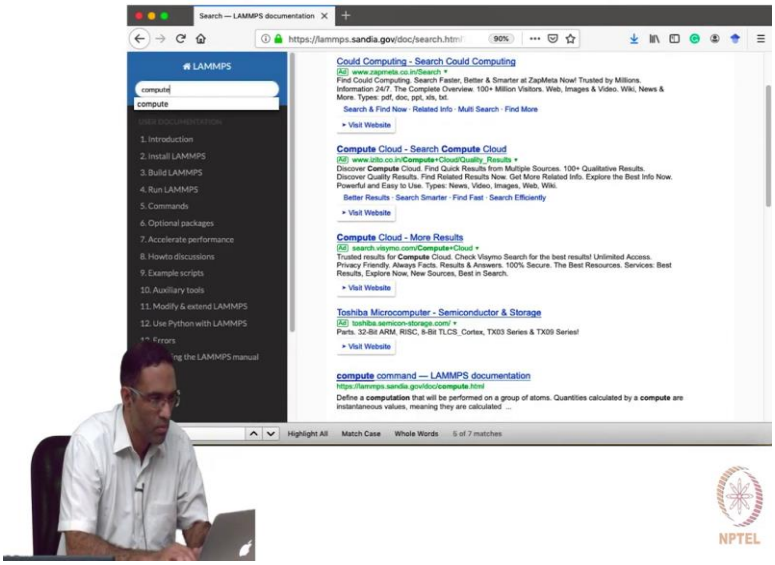
And what I have done is I have now obtained these, calculated these velocities and averaged them over time. So basically in the NVE run, I am calculating the velocities of each and every atom and summing them all up over the entire duration of the run and then plotting a histogram to see whether the distribution is similar to what I get at equilibrium. So that exercise basically is used to demonstrate what is referred to as our compute command and fix average per atom command.

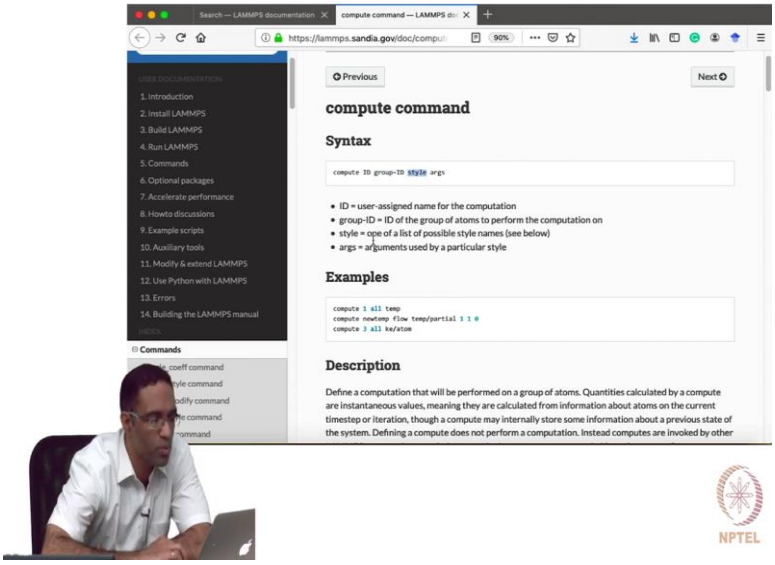
(Refer Slide Time: 12:02)

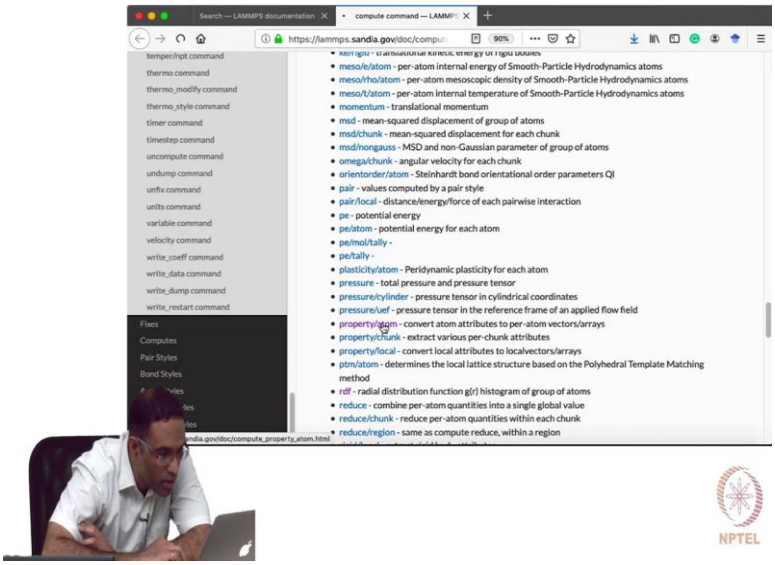


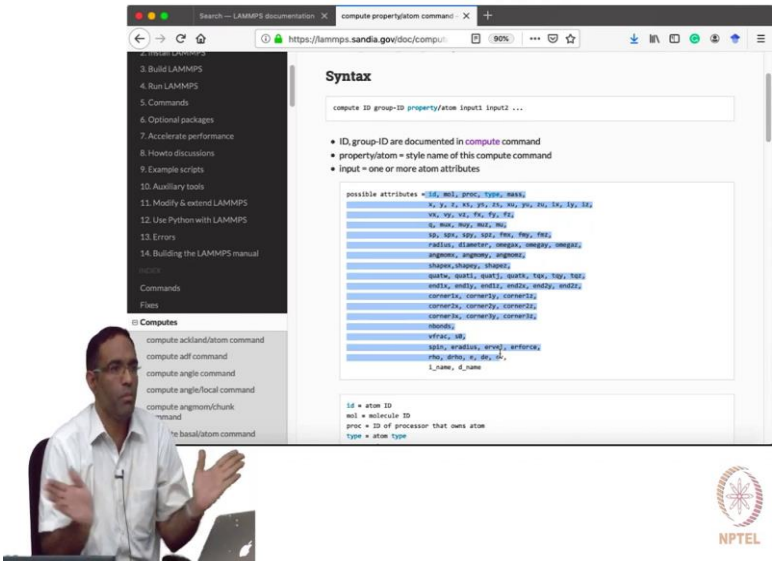
So, let us spend a little bit time on this. Everything is exactly the same as an exercise 8 here. So, nothing on the top is actually varying, everything is exactly the same. I just copied the same input file except that now I have not done all those calculations that I needed to show that a pv is equal to mrt, other than that everything is the same. But here just before I start running the last final leg of the NVE, I am defining a compute. The compute command, compute is basically the keyword VEL is basically the ID of the compute which I am going to use to refer to it later on the script. All is basically the group of atoms over which I want to calculate it and then I say that the computed property is a property that is defined per atom. So compute has various different kinds. You can do a lot of computations here.

(Refer Slide Time: 12:54)





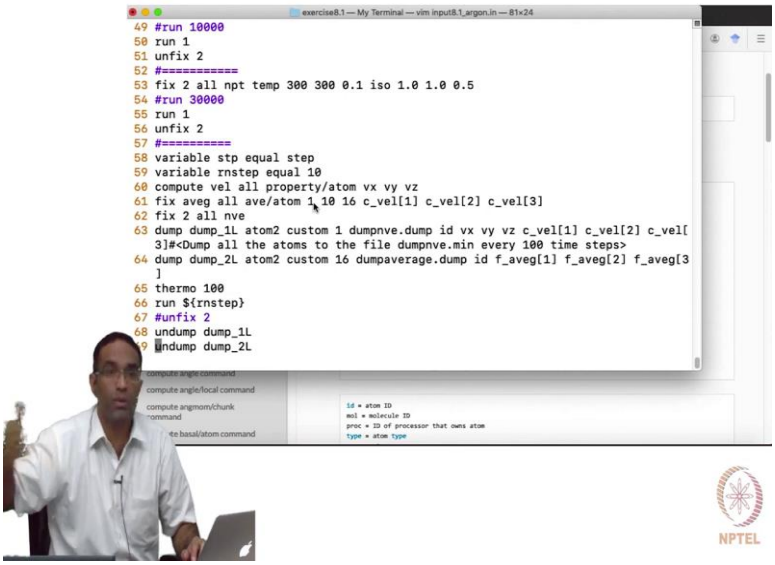




So compute ID is ID name of the compute, group ID is basically the group of atoms over which we want to carry out this compute and then the style. Style is one of a list of possible style names see below. So this is the huge number of computations that you can actually perform here. What I want to perform is, I want to calculate the velocities of all the atoms. So compute, these arguments can be anything, so I am looking at compute per atom, I am calculating the, I am interested in a property per atom.

So I go here click property per atom and then these are the various attributes that I can actually perform the calculation or get the calculation. I can calculate VX, VY and VZ, which is basically the velocities, X component, Y component and Z component of the velocity, but it has so many other things that also that you can do. For various reasons I cannot imagine the entire spectrum of instances where you would need all of these, but it depends on the problem that you are interested in. So for example, right now I wanted to plot the velocity distribution. So I am going to first calculate the velocities, VX, VY, VZ.

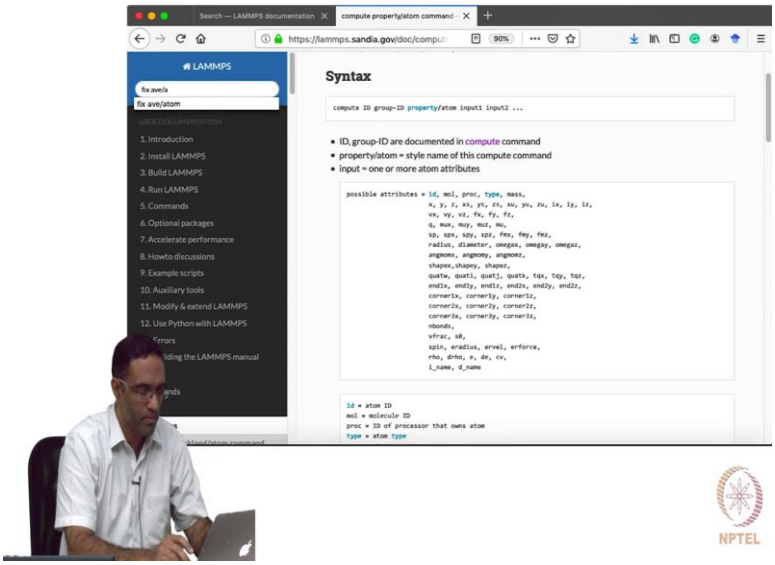
(Refer Slide Time: 14:33)

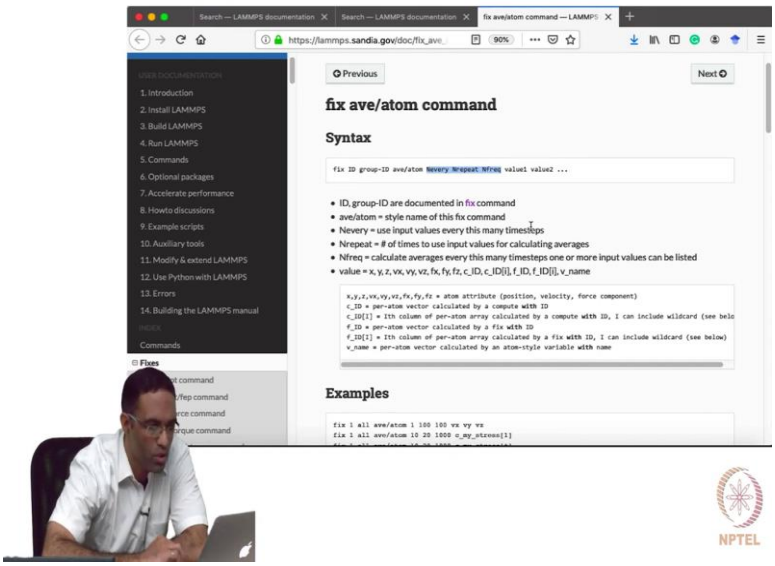


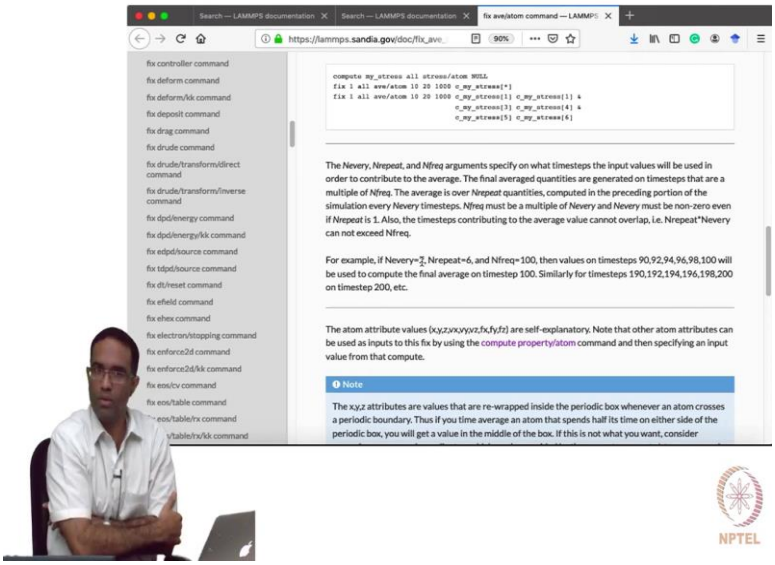
So what does that do that, simply calculates the velocities VX, VY, VZ and it that is available in the compute VEL. Now what I need to do now is, not interval if I just calculated it for one time, one snapshot, I know all the velocities. What I want to do is, to get good statistics, I want to average the velocity of say atom I over the entire NVE run, velocity of atom 2 over the entire run and then print out in a file the average VX, VY and VZ of every atom after averaging it over the entire run. So that involves a little bit of a trick here. So first I did some checks in order to make sure that I was understanding, whether this compute, whether this average is actually working.

So we use what is referred to as fix average per atoms. So this is basically the style name, average per atom is a style name, always the second thing is the ID that I would use in order to refer to it later on the script, all this is basically a group of atoms. And then this is something that you have to think carefully and give. So this basically specifies, how frequently do you want to calculate the averages, how many inputs do you want to take from the run in order to calculate the averages and how frequently do you want to repeat these calculations. These are the kind of information that will go in fix average atom command.

(Refer Slide Time: 16:15)







So you can go here, I will explain how this works.

Student: So the average velocity for a certain atom is it calculated like the sum of all the velocities at each time step divided by the total time.

Professor: Correct, yes total sum of all the velocities every time divided by total number of time steps. Yes.

Student: So, what do you means a how frequently do you want to calculate?

Professor: So, you are running you are running this entire step for 10000, 10000 steps doing, I can, I am LAMMPS gives me additional control to decide whether I want to use the velocity component, velocity X, let us say VX, VX component of atom 1 at instance at time step 1, time step 2, time step 3 time step 4, time step 5, times step 10000, add them all up and divide by 10000 or I want to use the information at time step 1, time step 3, time step 5, times step 7 times step 9 so on, divided by 5000 or do I want to do it in a different frequency.

So that frequency and all those things number of times how you want to do it is specified both by those 3 numbers. So those 3 numbers are here. It is N, every N repeat and N frequent. So N every is use the input values every this many time steps. It can be little hard to understand this, I will give you an example and then it will become clear. N repeat, number of times to use input values to calculate the averages and calculate averages every this many time step and you can list 1 or more input values can be listed here. So what I want to do, so a good, if you want to understand this clearly just take a look at the example that is right here.

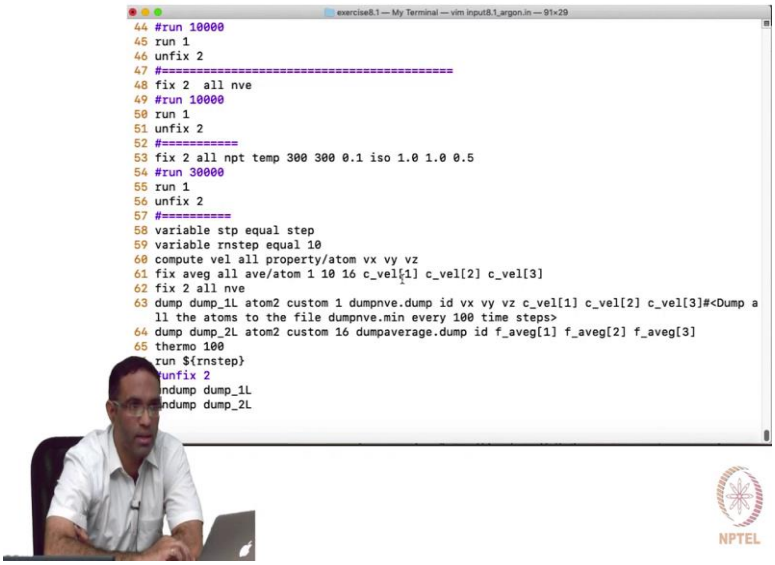
So if N every is 2, N repeat is 6 and N frequency is 100 then from 100 the last 6 values with 2 skipped will be taken into account in order to perform the average and it will print out the average after the 100th time step or in multiples of 100.

Student: What is the N repeat?

Professor: N repeat is 6, so 100, so how many things are taking 1 2 3, sorry 1 2 3 4, 1 2 3 4 5 6. 6, I am taking 6 snapshots and they are all taken every 2 times, you have a gap of 2 and then N frequency is 100. So if you are running your simulation for 200, you will have 1 set of average being printed out at the end of 100 steps but for that 100 it would only take 100, 98, 96, 94, 92 and 90 steps to calculate the average and then from 200 it would again take 200, 198, 196, 194, 192 and 190 to calculate the second average

So you have in your output file, once you output this you will have 2 averages being printed of. What I wanted to do is, I wanted to average it over the entire spectrum of runs. So obviously N frequency would essentially be the total number of steps for which I am running. N repeat would be as many data points as there are, would again be 10000 or 10 or whatever I want it to be and this would be 1, because I want everything to be counted.

(Refer Slide Time: 20:05)



Now, this is a little complicated. So we need to check whether, what LAMMPS is doing and what we think it is doing is actually correct. Sometimes the manual can be a little bit difficult to understand, so it is a good idea to make some quick checks to see if that is what? it is doing.

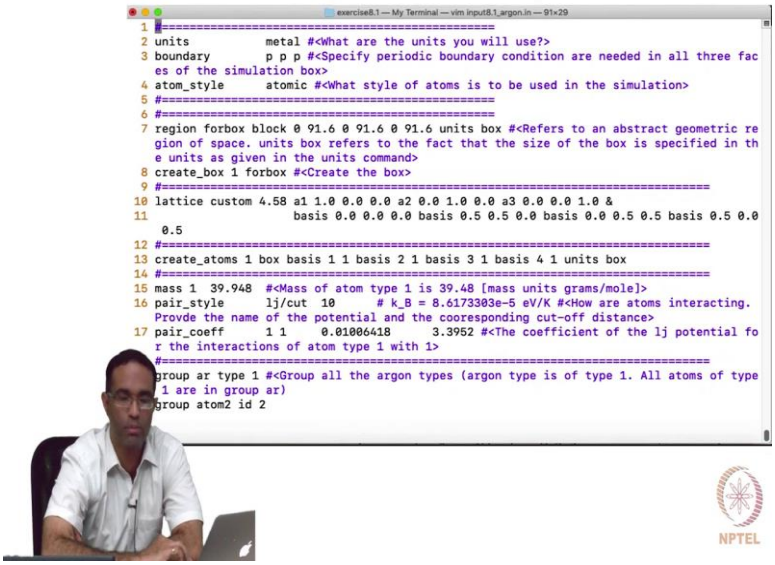
Student: So, N frequency is telling us after how many steps we want it, or it is same because frequency usually say, if you say 100 then we want to repeat 100 times?

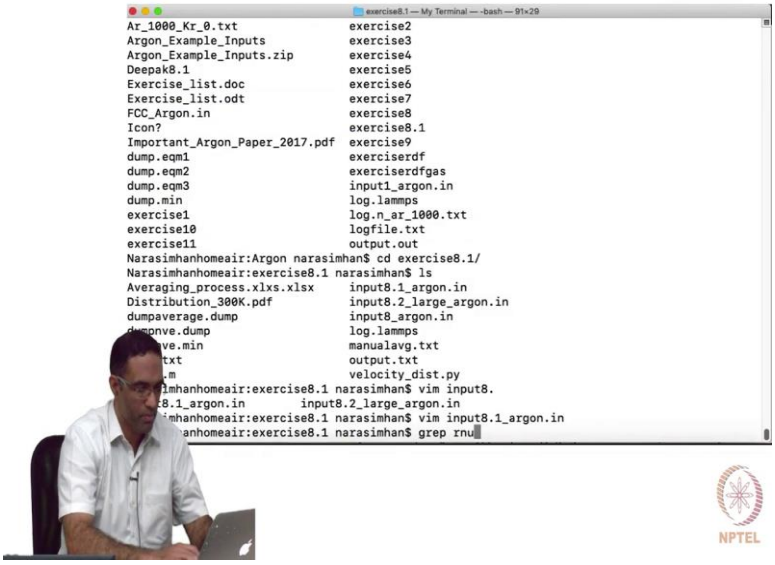
Professor: No, that is not what it means. So here is if your total number of steps is…

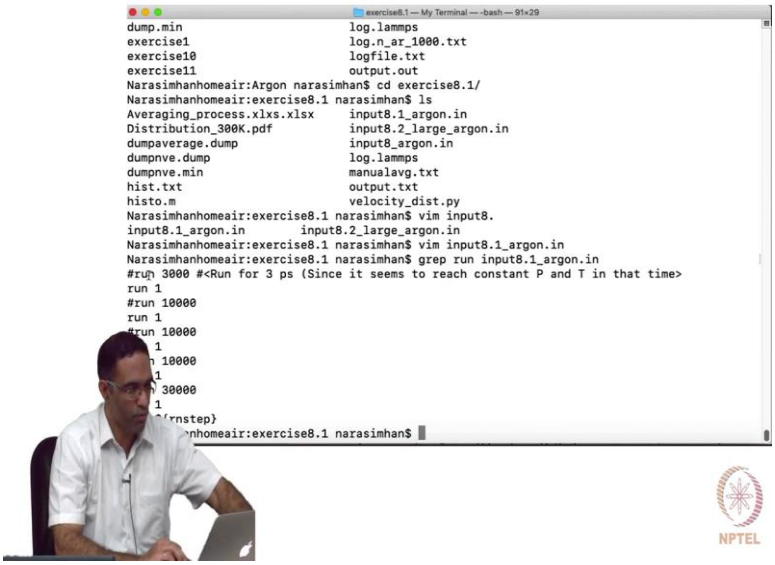
Student: N frequency was I thought the total number of steps by whatever we give, every that step it will print but here we are specifying the step itself.

Professor: Yes, you are specifying the step itself, exactly, that is how it is working.

(Refer Slide Time: 21:00)







exercise8.1 — My Terminal — vim input8.1_argon.in — 91×29

```
#=============================================
units           metal #<What are the units you will use?>
boundary        p p p #<Specify periodic boundary condition are needed in all three fac
es of the simulation box>
atom_style      atomic #<What style of atoms is to be used in the simulation>
#=============================================
#=============================================
region forbox block 0 91.6 0 91.6 0 91.6 units box #<Refers to an abstract geometric re
gion of space. units box refers to the fact that the size of the box is specified in th
e units as given in the units command>
create_box 1 forbox #<Create the box>
#=============================================================================
lattice custom 4.58 a1 1.0 0.0 0.0 a2 0.0 1.0 0.0 a3 0.0 0.0 1.0 &
                    basis 0.0 0.0 0.0 basis 0.5 0.5 0.0 basis 0.0 0.5 0.5 basis 0.5 0.0
 0.5
#=============================================================================
create_atoms 1 box basis 1 1 basis 2 1 basis 3 1 basis 4 1 units box
#=============================================================================
mass 1  39.948  #<Mass of atom type 1 is 39.48 [mass units grams/mole]>
pair_style      lj/cut  10      # k_B = 8.6173303e-5 eV/K #<How are atoms interacting.
Provde the name of the potential and the cooresponding cut-off distance>
pair_coeff      1 1     0.01006418      3.3952 #<The coefficient of the lj potential fo
r the interactions of atom type 1 with 1>
#=============================================================================
group ar type 1 #<Group all the argon types (argon type is of type 1. All atoms of type
 1 are in group ar)
group atom2 id 2

".1_argon.in" [dos] 69L, 3582C
```

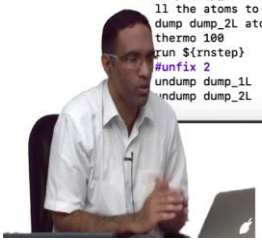
exercise8.1 — My Terminal — vim input8.1_argon.in — 91×29

```
#=============================================================================
mass 1  39.948  #<Mass of atom type 1 is 39.48 [mass units grams/mole]>
pair_style      lj/cut  10      # k_B = 8.6173303e-5 eV/K #<How are atoms interacting.
Provde the name of the potential and the cooresponding cut-off distance>
pair_coeff      1 1     0.01006418      3.3952 #<The coefficient of the lj potential fo
r the interactions of atom type 1 with 1>
#=============================================================================
group ar type 1 #<Group all the argon types (argon type is of type 1. All atoms of type
 1 are in group ar)
group atom2 id 2
minimize 1e-12 1e-25 10000 10000 #<Minimize the energy using a conjugate gradient step.

#=============================================================================
# Provide an initial maxwellian distribution of velocity corresponding to temperature 3
00K
velocity all create 300 198728 dist gaussian
#Perform an NVE integration with this initial position and velocity distribution
timestep 0.001 #<Time step in ps. So this is 0.001ps or 1 femto second>
fix 1 all nve
thermo_style custom step time cpu temp pe ke etotal press vol #<Print the thermodynamic
 information >
thermo 1000
#run 3000 #<Run for 3 ps (Since it seems to reach constant P and T in that time>
run 1
fix 1
#=============================================================================
#s now change the temperature and pressure slowly using NPT=
#=============================================================================
```

exercise8.1 — My Terminal — vim input8.1_argon.in — 91×29

```
fix 2 all nvt temp 300 300 0.5
#run 10000
run 1
unfix 2
#=========================================
fix 2  all nve
#run 10000
run 1
unfix 2
#===========
fix 2 all npt temp 300 300 0.1 iso 1.0 1.0 0.5
#run 30000
run 1
unfix 2
#==========
variable stp equal step
variable rnstep equal 10
compute vel all property/atom vx vy vz
fix aveg all ave/atom 1 10 16 c_vel[1] c_vel[2] c_vel[3]
fix 2 all nve
dump dump_1L atom2 custom 1 dumpnve.dump id vx vy vz c_vel[1] c_vel[2] c_vel[3]#<Dump a
ll the atoms to the file dumpnve.min every 100 time steps>
dump dump_2L atom2 custom 16 dumpaverage.dump id f_aveg[1] f_aveg[2] f_aveg[3]
thermo 100
run ${rnstep}
#unfix 2
undump dump_1L
undump dump_2L
```



So let us take a look at the example here. A simple example like I said if I want to check something in LAMMPS I run everything for 1 step to make sure that everything is, all the commands are specified properly. I am just going to say that, how many steps have actually passed since I started simulation. So let us do a grep run. So,1, 2, 3, 4, 5 steps are there. So, and then there is, there are actually 6 steps including the step that we used for our minimization. I do not know why this is not turning up. So 1, 2, 3, 4. So what you have to give here. So I do not know why this is 16, it should be 15.

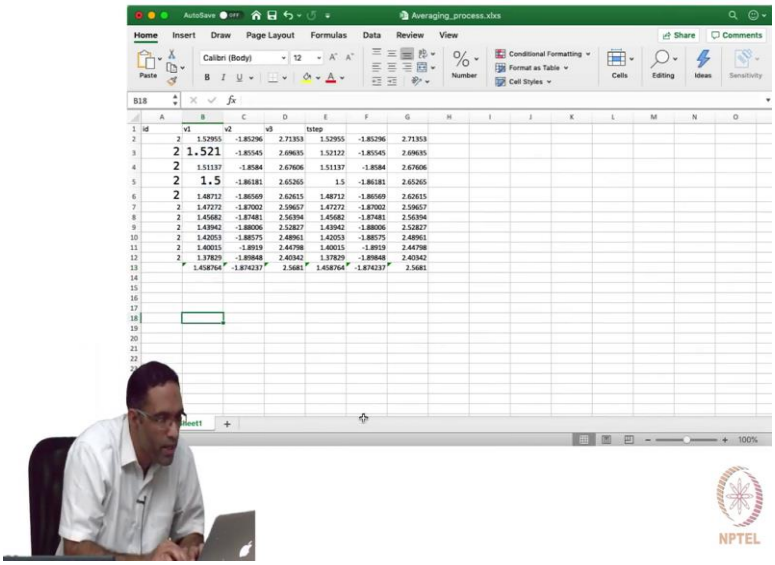
But anyways, if there are say 5 steps before this, the step that it starts printing. So it starts printing from 6, 7, 8, 9, 10; 6 to 10 steps. At the end of this it will be 5 and then it will start printing 6 to 16 steps which are actually 10 steps. I am going to run it only for 10 steps, so it is correct. So until here it has finished 5 steps run 1 is occurring 5 times and then it is going to start printing this dump for about for only 1 atom. I am just doing it for 1 atom because I just want to see if I am able to add up the velocities and it averages out correctly. So I have in the previously I defined a new group called atom, atom 2 which contains only the atom 2 and I am going to perform that calculation only for that particular atom here.

So I am just going to print the velocities VX, VY, VZ and then, for every step, and then I am going to say that you calculate my average starting from 16th step for all the 10 steps at a frequency of 1. So it is running for rn step is basically equal to 10 here. So what is going to do is, it is going to take 16, 15, 14, 13, 12, 11, 10 and add them up and store them in the file dump average dot dump. And this is the manner you actually if I what you want to print out. So this is a fix ID which performs this calculation and this is the manner you print out that information. So this is the average X velocity this will contain the average Y velocity this will contains the average Z velocity over these last 10 steps for this particular atom 2. So I did that in an excel file just to convince myself that was the way it worked.

Student: Is it necessary to put F underscore?

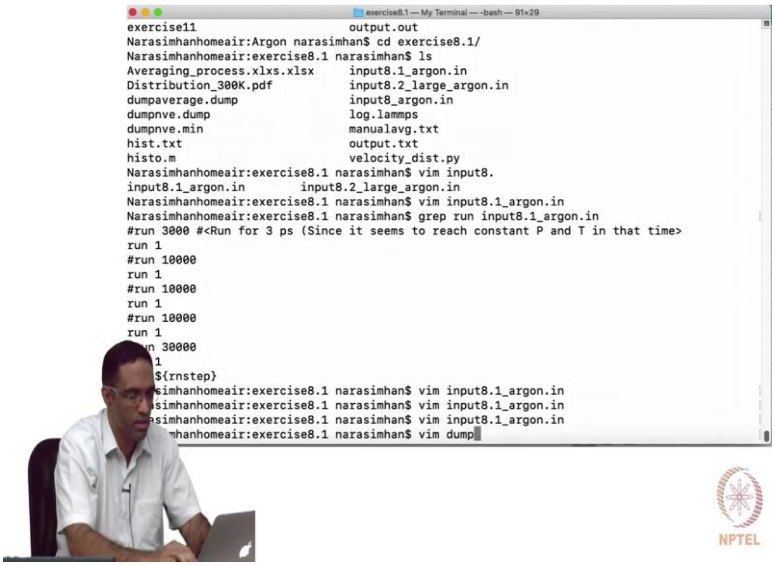
Professor: Yes, because a compute is referred to by C underscore, a fix is referred to by F underscore, variables are referred to by V underscore or dollar within flower brackets.

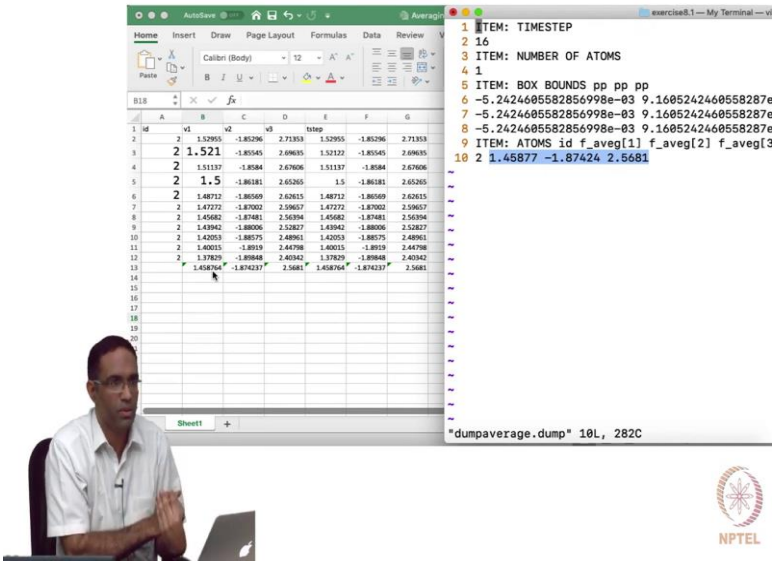
(Refer Slide Time: 24:43)



So once you do this, so what I am trying to show you here is Argon, exercise 8.1, averaging process. So including the first 1 which I do not want to average the velocities of atom 2 are in the, this is the 16th, 15th, 14th, 13th, 12, 11, 10, 9, 8, 7 from here it should start summing or averaging. And these are the values that I obtained from the dump file for the velocities of this atom 2 and this average is therefore calculated over these 10 samples and it turns out to be 1.458764

(Refer Slide Time: 25:31)





Now, we can open our, I do not know whether I have that sure. For just that atom if you look at the average that LAMMPS is calculated, the Z, velocity Z is 2.5681 and V2 is 1.87424 and VX is exactly the value that we basically wanted, 1.45877 and 1.45877 and what we think it is doing by giving those numbers and what it is actually doing match. So essentially we have understood what this command does well. So this test is extremely important to perform otherwise you will not understand what this N repeat, N every and N frequency actually does.

Student: Sir, there should be some constraint on this is N repeat, N every and N frequency?

Professor: Yes

Student: Like, product of N repeat, N every should be less than …

Professor: N frequency. Absolutely. This is how you can average the properties. Now what you have done is you have averaged the properties of every atom. This is fixed average per atom, per atom quantities are being averaged. In this manner you can average any per atom quantity. This is just an example of velocity but say for example you wanted to average the pressure you can calculate the pressure per atom and average it properly.

Student: Sir when we put 16 here, we included the fact that we had done some small runs in the beginning and then calculate them after that and put it there. But suppose we are running this for like another 1000 steps still input 32, 48 and so on like this N freq. N frequency would mean that, so we saw the example that it was 100 and it would be there at 200. So here again so it will go to 32...

Professor: No, because my, that particular part of simulation stopped at 10 but if did not, yes it would.

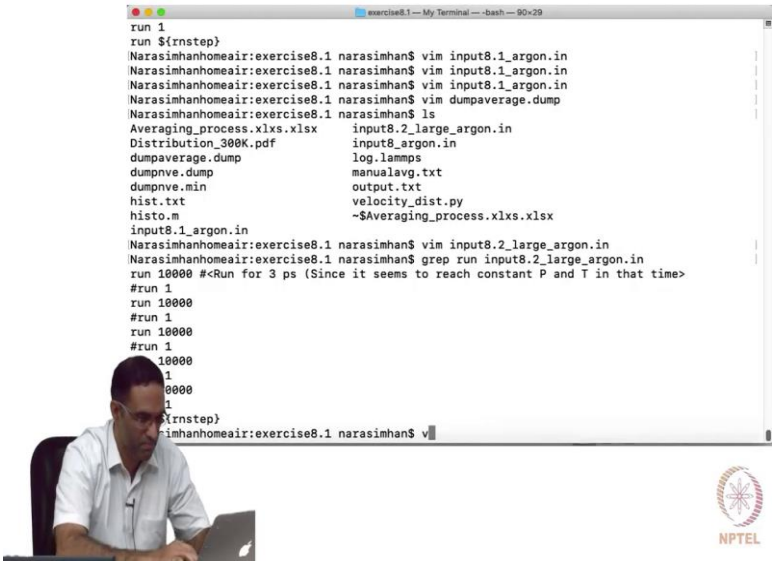
Student: It would repeat every 16 steps?

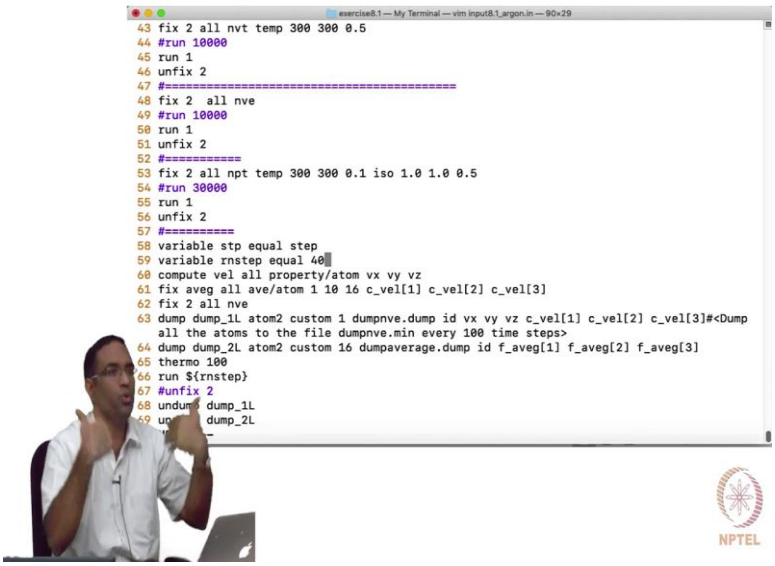
Professor: Yes, it would repeat every 16 steps.

Student: But here automatic

Professor: That was my aim that is why I did it like that.

(Refer Slide Time: 27:55)





If you did this, if you gave here naught 8, or if this was actually 32, if this was 20, then this can be every, for a 40 maybe, say 40.

Student: Then you get two averages?

Professor: You will get two averages.

Student: Sir, whenever we start at fix, we unfix it at the end. I do not understand why is that necessary because we are specifying the fix number of time steps using the run command, so if we specifying 1000, it should stop after 1000.

Professor: That is not the way it works.

Student: It does not stop at 1000?

Professor: No.

Student: Then why are you specifying 1000?

Professor: So that I wanted to run that fix for about 1000.

Student: No, if it is not stopping even at 1000.

Professor: No, no, it is not stopping at 1000. It will run this at 1000. It will run something for 1000 steps but these fixes are still switched on. So if you continue to run another 1000, it will also run that if you did not unfix it.

Student: I did not get your point, the switch. Let us say first fix is fix NVE then we unfix it and start another fix NPT. So before fix NPT, if I do not write NPT what will happen?

Professor: It is running that NVE for some more time, because you have 1 more run here. It is equal to saying this like, fix NVE, run 10000, run 10000. That is what it means.

Student: So it will run NVE or NVT?

Professor: Whatever, anything, anything if you run.

Student: If you run it without unfixing it will run 10 first and then 10 again and then 20.

Professor: 20, yes. So do not do it. You do not want that.

Student: 20 again 20…

Student: Then 20 20, right? 10 20 20?

Professor: You can experiment with these things. I have not done that. All these combinations of how many runs to give, to me it makes logical sense to actually unfix something because there is a physical meaning to that. So I do not want that to be, I do not want that fix to be applied to those set of atoms anymore. So I just stop doing it. So let us look at it from that perspective. If you, why do not you try. You just give 10, 20, 20, 30 and see what happens. I do not know what happens actually.