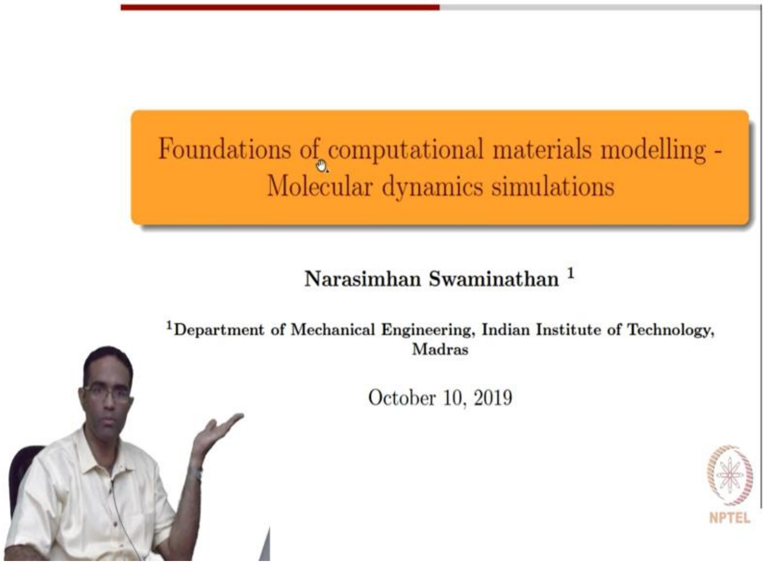
Foundation of Computational Materials Modelling
Professor Narasimhan Swaminathan
Department of Mechanical Engineering
Indian Institute of Technology Madras
Lecture 25
Basic introduction to MD

(Refer Slide Time: 00:16)



Foundations of computational materials modelling - Molecular dynamics simulations

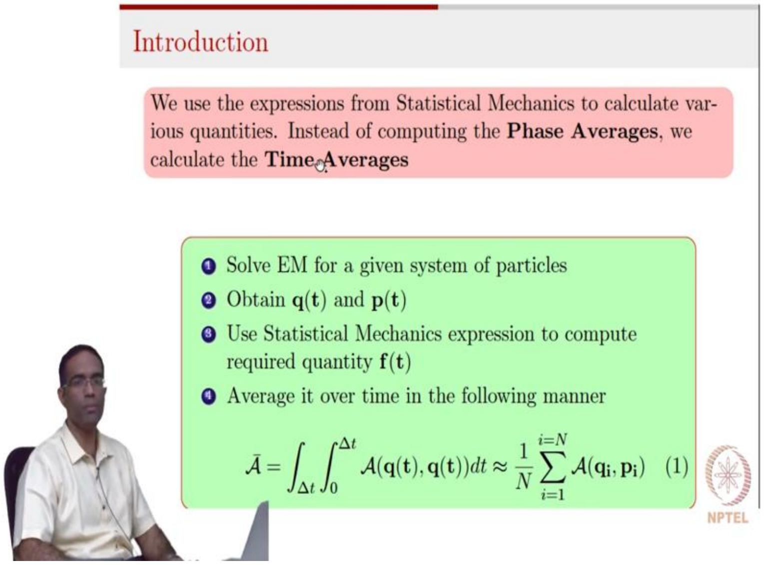
Narasimhan Swaminathan [1]

[1]Department of Mechanical Engineering, Indian Institute of Technology, Madras

October 10, 2019

So good afternoon. So let us continue with our course. So since we have completed statistical mechanics I think it is a, it is the appropriate time for us to start discussing Molecular dynamics simulations and to see how we can do some simple simulations, understand the foundations and some fundamental concepts that are involved while running molecular dynamics simulation. We are not going to write any code in order to perform these simulations. We are going to use an existing code called LAMMPS in order to do this. So the basic purpose of this module is to teach you how to use LAMMPS to some extend and show you some simple inputs scripts with which you can, you will be doing several exercises and that is one way we would be learning how to use LAMMPS.

(Refer Slide Time: 01:02)



Introduction

We use the expressions from Statistical Mechanics to calculate various quantities. Instead of computing the **Phase Averages**, we calculate the **Time Averages**

1. Solve EM for a given system of particles
2. Obtain $\mathbf{q(t)}$ and $\mathbf{p(t)}$
3. Use Statistical Mechanics expression to compute required quantity $\mathbf{f(t)}$
4. Average it over time in the following manner

$$\bar{A} = \int_{\Delta t}\int_0^{\Delta t} \mathcal{A}(\mathbf{q(t)}, \mathbf{q(t)})dt \approx \frac{1}{N}\sum_{i=1}^{i=N}\mathcal{A}(\mathbf{q_i}, \mathbf{p_i}) \quad (1)$$

But before doing that I think we need to we need a little bit more background on what this molecular dynamics stimulation software does not only LAMMPS for that matter anything. So firstly like we have been talking so far Statistical mechanics expressions where various thermodynamics quantities are given in terms of the position and momenta are known. And instead of computing the phase averages we actually compute time averages in molecular dynamic simulations.

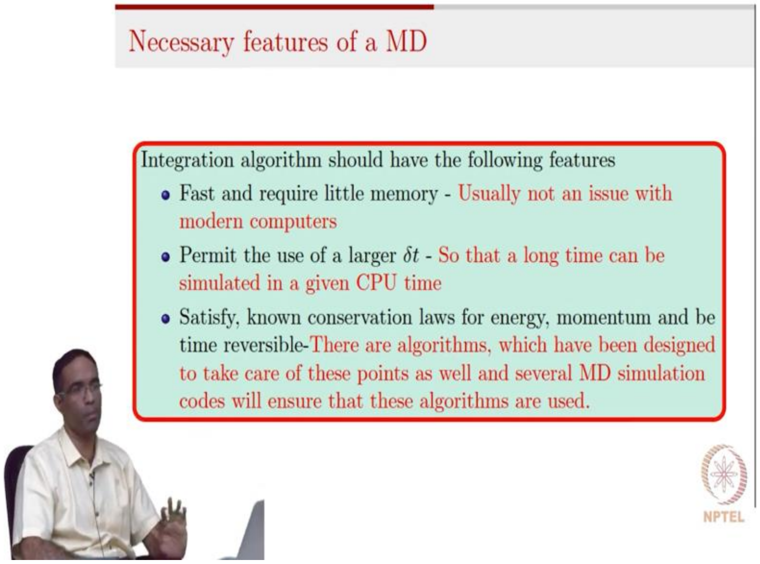
So essential steps are, solve the equation of motions for a given system of particles, obtain the variation of the positions and the momenta as function of time. And then use statistical mechanics to actually compute the required quantities say some function of t and then you basically average it over time. So the manner in which you do it is given in the following formula right here. So this first integral should not be there it is nothing but 1 by delta t, 1 by delta t limit delta t tends to infinity is what we use to write that is what I intended to write here but I think I made a mistake. So that is not right. But instead of averaging it in this particular manner.

Student: delta t tends to 0.

Professor: delta t tends to 0 no over a long period of time. So t equal to t naught plus delta t t equal to t naught to t equal to t naught plus delta t for very large delta t is what you are going to integrate. So delta t is not 0 delta t is a very large quantity and which is what is the process of time averaging so to speak.

However, in molecular dynamics simulations what happens is the software is capable of giving you the positions and the momenta of all the particles that are there in the system. So once you know A as a function of p and q or the positions of the momenta, then you just sum over all the, of all time which you have actually printed out the positions of the momenta, add them up and divide by the total number of sample that you have to give you the average value of this particular quantity. So this is how you essentially calculate the time averages of quantities in molecular dynamics simulation.

(Refer Slide Time: 03:25)



Necessary features of a MD

Integration algorithm should have the following features

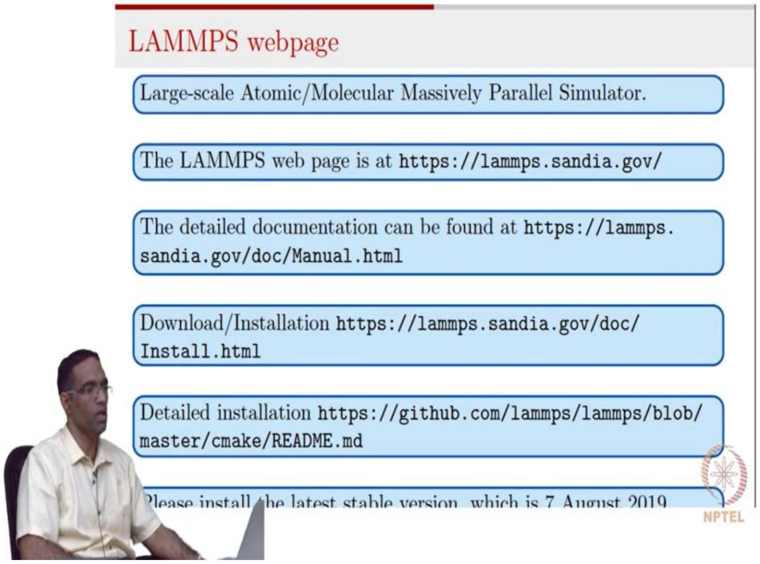
- Fast and require little memory - Usually not an issue with modern computers
- Permit the use of a larger δt - So that a long time can be simulated in a given CPU time
- Satisfy, known conservation laws for energy, momentum and be time reversible-There are algorithms, which have been designed to take care of these points as well and several MD simulation codes will ensure that these algorithms are used.

So if you really want to write a molecular dynamic simulations code, then simulations a code which actually does molecular dynamic simulations, there are lot of issues that you have to keep track of. Thankfully, all these things have already been taken care of in these softwares like LAMMPS because of which we do not need to do. But is a good idea to know some of these things before we start doing some simulations. Firstly, it needs to be reasonably fast it should require reasonable amount of memory. And all these things have to be kept track in the codes that you write in various ways. It must permit the use of somewhat larger time steps. So when you are trying to integrate the equations of motion you are doing so through a numerical means.

And therefore, you have to specify some time step in order to capture the position and the momenta of the particles for various times. So you should be able to use somewhat reasonably the numerical scheme must allow you to use reasonably large delta t. At the same time give you a reasonable amount of accuracy. So if you have taken codes in numerical methods you will actually know how these things are actually important. Last but not the least it must actually satisfy the numerical method that you are using must actually satisfy the various conservation laws like conservation of energy, conservation of momentum.

And also be time reversible, all our equations of motions are actually time reversible in reality, the classical equation of motion. If you substitute t by minus t you should get back the same expression for the movement of the particles. Of course these things have mathematical bases and we can talk about them at length in probably in different course. But what I want to insist is that the algorithms that are actually used in softwares like LAMMPS actually make sure that all these things are happening properly. So we do not really want to worry about that. You just need to keep them in mind.

(Refer Slide Time: 05:27)



LAMMPS webpage

Large-scale Atomic/Molecular Massively Parallel Simulator.

The LAMMPS web page is at `https://lammps.sandia.gov/`

The detailed documentation can be found at `https://lammps.sandia.gov/doc/Manual.html`

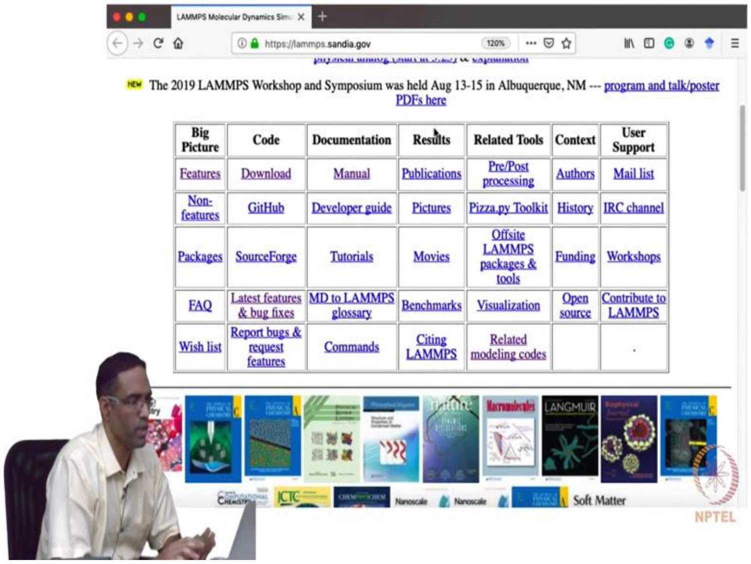
Download/Installation `https://lammps.sandia.gov/doc/Install.html`

Detailed installation `https://github.com/lammps/lammps/blob/master/cmake/README.md`

Please install the latest stable version, which is 7 August 2019

So LAMMPS is what we will be using. LAMMPS is a very popular molecular dynamic stimulation code. Large scale atomic molecular massively parallel stimulator is its expansion, that is what LAMMPS start for. And I am sure you all have visited the LAMMPS website so and there is a detailed documentation that LAMMPS has we will go over some elements of the documentation. And then I hope you all have had an opportunity to install some version of LAMMPS on your computers and we have a detail installation instructions given in this website write here. And I have checked that some of the codes that I am going to be using now are all going to be working with the most latest stable version of LAMMPS with the 7th August 2019. So should not have any serious problem there.

(Refer Slide Time: 06:22)

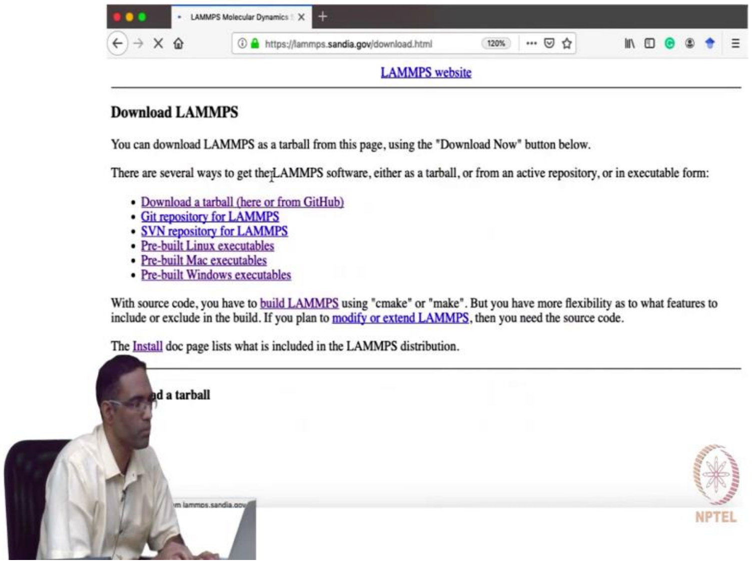


The 2019 LAMMPS Workshop and Symposium was held Aug 13-15 in Albuquerque, NM --- program and talk/poster PDFs here

Big Picture	Code	Documentation	Results	Related Tools	Context	User Support
Features	Download	Manual	Publications	Pre/Post processing	Authors	Mail list
Non-features	GitHub	Developer guide	Pictures	Pizza.py Toolkit	History	IRC channel
Packages	SourceForge	Tutorials	Movies	Offsite LAMMPS packages & tools	Funding	Workshops
FAQ	Latest features & bug fixes	MD to LAMMPS glossary	Benchmarks	Visualization	Open source	Contribute to LAMMPS
Wish list	Report bugs & request features	Commands	Citing LAMMPS	Related modeling codes		.

So this is the LAMMPS website, a lots of stuff one can get overwhelmed with the number of things you can actually find in this website.

(Refer Slide Time: 6:36)



LAMMPS website

Download LAMMPS

You can download LAMMPS as a tarball from this page, using the "Download Now" button below.

There are several ways to get the LAMMPS software, either as a tarball, or from an active repository, or in executable form:

- Download a tarball (here or from GitHub)
- Git repository for LAMMPS
- SVN repository for LAMMPS
- Pre-built Linux executables
- Pre-built Mac executables
- Pre-built Windows executables

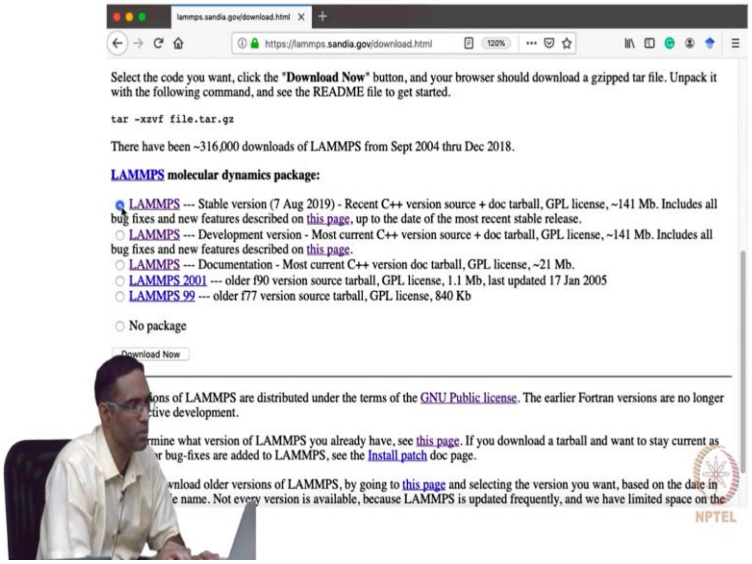
With source code, you have to build LAMMPS using "cmake" or "make". But you have more flexibility as to what features to include or exclude in the build. If you plan to modify or extend LAMMPS, then you need the source code.

The Install doc page lists what is included in the LAMMPS distribution.

d a tarball

But the download page is something that is important for us.

(Refer Slide Time: 6:39)



Select the code you want, click the **"Download Now"** button, and your browser should download a gzipped tar file. Unpack it with the following command, and see the README file to get started.

```
tar -xzvf file.tar.gz
```

There have been ~316,000 downloads of LAMMPS from Sept 2004 thru Dec 2018.

LAMMPS molecular dynamics package:

● LAMMPS --- Stable version (7 Aug 2019) - Recent C++ version source + doc tarball, GPL license, ~141 Mb. Includes all bug fixes and new features described on this page, up to the date of the most recent stable release.
○ LAMMPS --- Development version - Most current C++ version source + doc tarball, GPL license, ~141 Mb. Includes all bug fixes and new features described on this page.
○ LAMMPS --- Documentation - Most current C++ version doc tarball, GPL license, ~21 Mb.
○ LAMMPS 2001 --- older f90 version source tarball, GPL license, 1.1 Mb, last updated 17 Jan 2005
○ LAMMPS 99 --- older f77 version source tarball, GPL license, 840 Kb

○ No package

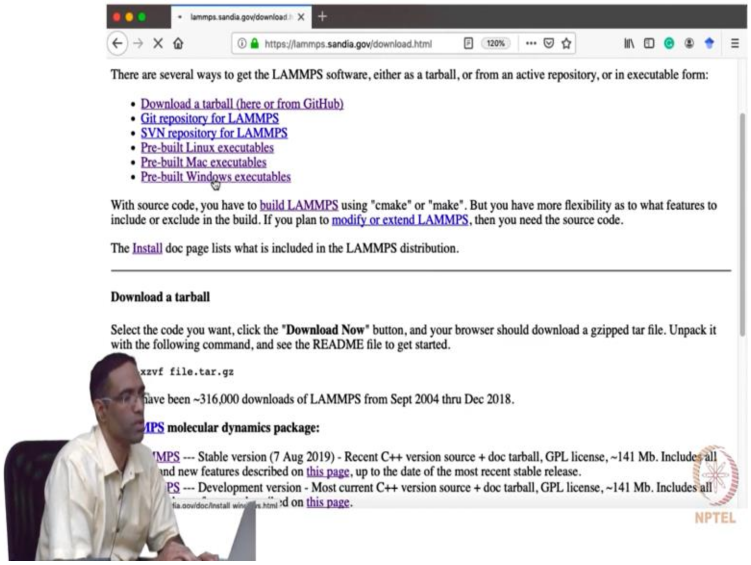
ons of LAMMPS are distributed under the terms of the GNU Public license. The earlier Fortran versions are no longer tive development.

rmine what version of LAMMPS you already have, see this page. If you download a tarball and want to stay current as r bug-fixes are added to LAMMPS, see the Install patch doc page.

wnload older versions of LAMMPS, by going to this page and selecting the version you want, based on the date in e name. Not every version is available, because LAMMPS is updated frequently, and we have limited space on the

What I did was you click the LAMMPS stable version and click download now and it will give you most stable version.

(Refer Slide Time: 6:52)



There are several ways to get the LAMMPS software, either as a tarball, or from an active repository, or in executable form:

- Download a tarball (here or from GitHub)
- Git repository for LAMMPS
- SVN repository for LAMMPS
- Pre-built Linux executables
- Pre-built Mac executables
- Pre-built Windows executables

With source code, you have to build LAMMPS using "cmake" or "make". But you have more flexibility as to what features to include or exclude in the build. If you plan to modify or extend LAMMPS, then you need the source code.

The Install doc page lists what is included in the LAMMPS distribution.

Download a tarball

Select the code you want, click the **"Download Now"** button, and your browser should download a gzipped tar file. Unpack it with the following command, and see the README file to get started.

```
xzvf file.tar.gz
```

have been ~316,000 downloads of LAMMPS from Sept 2004 thru Dec 2018.

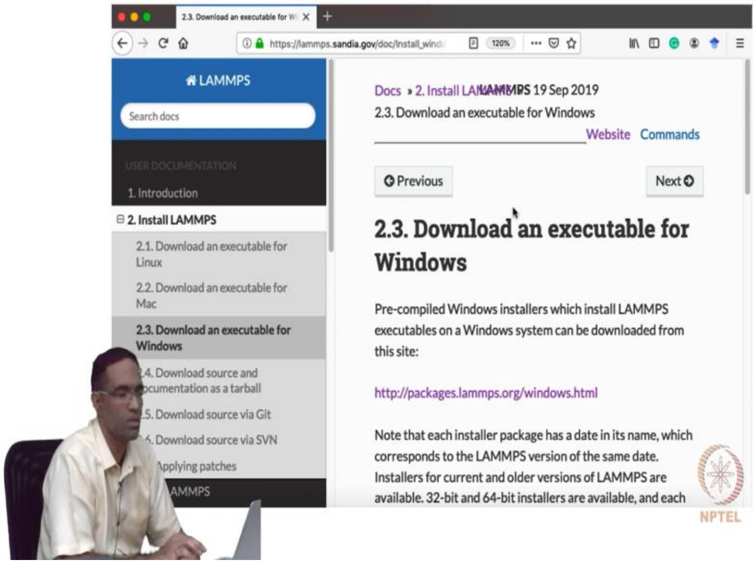
MPS molecular dynamics package:

MPS --- Stable version (7 Aug 2019) - Recent C++ version source + doc tarball, GPL license, ~141 Mb. Includes all and new features described on this page, up to the date of the most recent stable release.

PS --- Development version - Most current C++ version source + doc tarball, GPL license, ~141 Mb. Includes all d on this page.

And there are is you if you have if you want prebuilt executables for your operating system than you can just click on to this and go to the corresponding prebuilt executables.

(Refer Slide Time: 7:01)



2.3. Download an executable for Windows

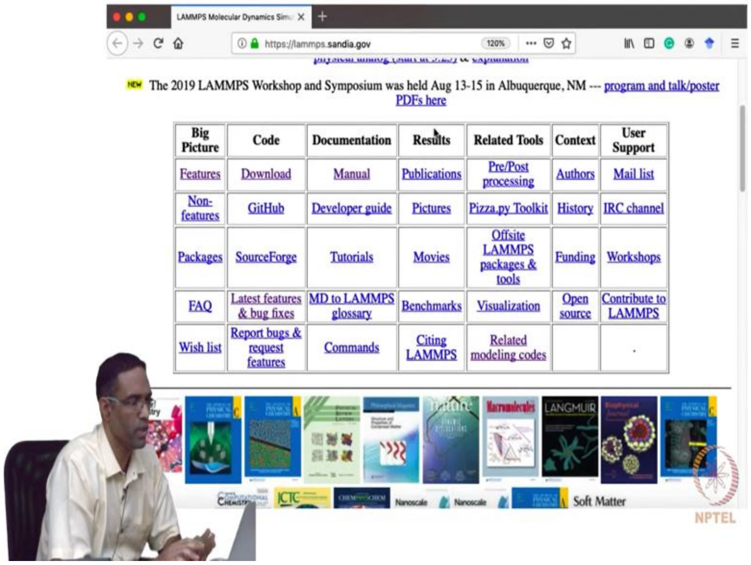
Pre-compiled Windows installers which install LAMMPS executables on a Windows system can be downloaded from this site:

http://packages.lammps.org/windows.html

Note that each installer package has a date in its name, which corresponds to the LAMMPS version of the same date. Installers for current and older versions of LAMMPS are available. 32-bit and 64-bit installers are available, and each

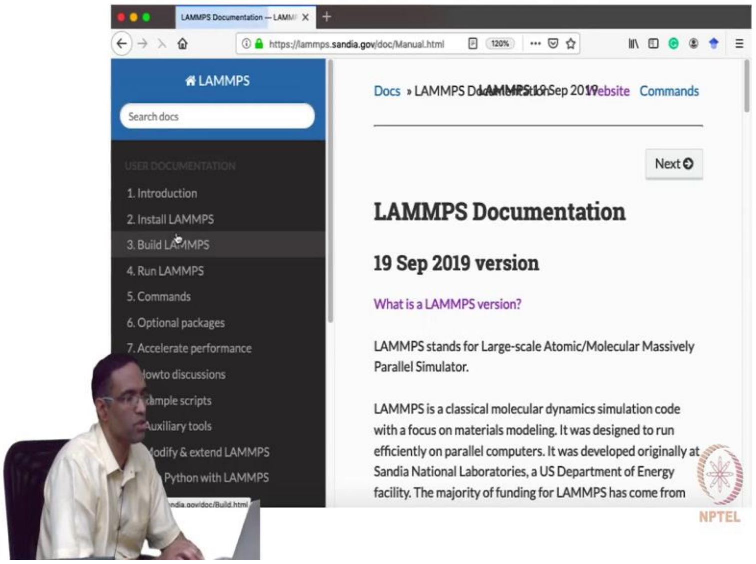
And (compile) and have it for your system and then use it for your specific system windows or mac or Linux. I generally preferred to use the downloaded from the source and then compile it on my computer and generate an executable file for my system.

(Refer Slide Time: 7:26)



NEW The 2019 LAMMPS Workshop and Symposium was held Aug 13-15 in Albuquerque, NM --- program and talk/poster PDFs here

Big Picture	Code	Documentation	Results	Related Tools	Context	User Support
Features	Download	Manual	Publications	Pre/Post processing	Authors	Mail list
Non-features	GitHub	Developer guide	Pictures	Pizza.py Toolkit	History	IRC channel
Packages	SourceForge	Tutorials	Movies	Offsite LAMMPS packages & tools	Funding	Workshops
FAQ	Latest features & bug fixes	MD to LAMMPS glossary	Benchmarks	Visualization	Open source	Contribute to LAMMPS
Wish list	Report bugs & request features	Commands	Citing LAMMPS	Related modeling codes		.



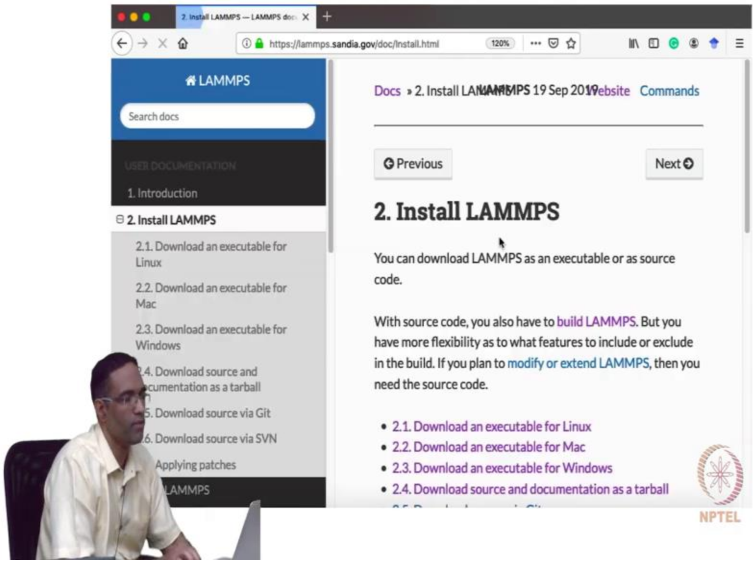
LAMMPS Documentation

19 Sep 2019 version

What is a LAMMPS version?

LAMMPS stands for Large-scale Atomic/Molecular Massively Parallel Simulator.

LAMMPS is a classical molecular dynamics simulation code with a focus on materials modeling. It was designed to run efficiently on parallel computers. It was developed originally at Sandia National Laboratories, a US Department of Energy facility. The majority of funding for LAMMPS has come from

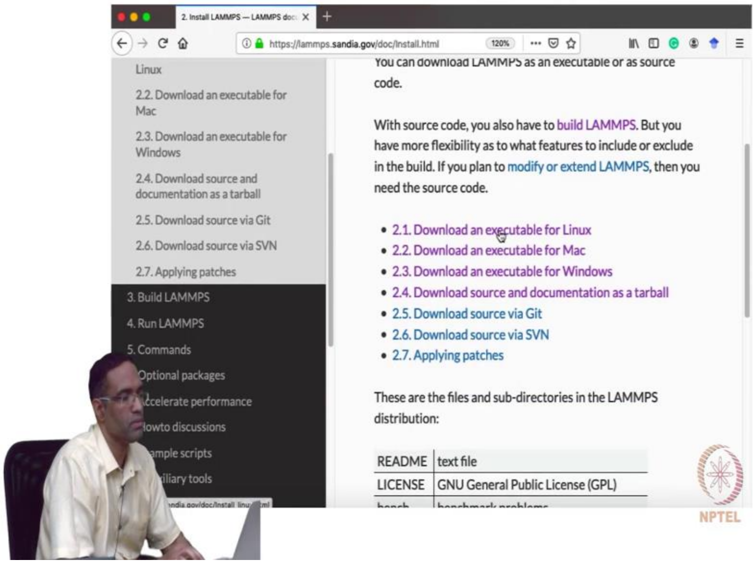


2. Install LAMMPS

You can download LAMMPS as an executable or as source code.

With source code, you also have to build LAMMPS. But you have more flexibility as to what features to include or exclude in the build. If you plan to modify or extend LAMMPS, then you need the source code.

- 2.1. Download an executable for Linux
- 2.2. Download an executable for Mac
- 2.3. Download an executable for Windows
- 2.4. Download source and documentation as a tarball



You can download LAMMPS as an executable or as source code.

With source code, you also have to build LAMMPS. But you have more flexibility as to what features to include or exclude in the build. If you plan to modify or extend LAMMPS, then you need the source code.

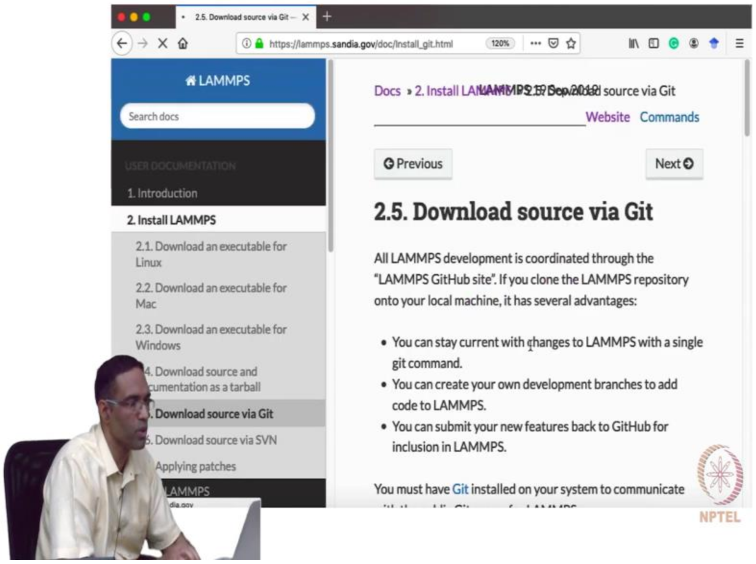
- 2.1. Download an executable for Linux
- 2.2. Download an executable for Mac
- 2.3. Download an executable for Windows
- 2.4. Download source and documentation as a tarball
- 2.5. Download source via Git
- 2.6. Download source via SVN
- 2.7. Applying patches

These are the files and sub-directories in the LAMMPS distribution:

README	text file
LICENSE	GNU General Public License (GPL)

If you want to know how to do that also you can actually go to the manual and in the manual you will find install LAMMPS. After downloading, just download an executable for Linux but that is not we want.

(Refer Slide Time: 7:45)

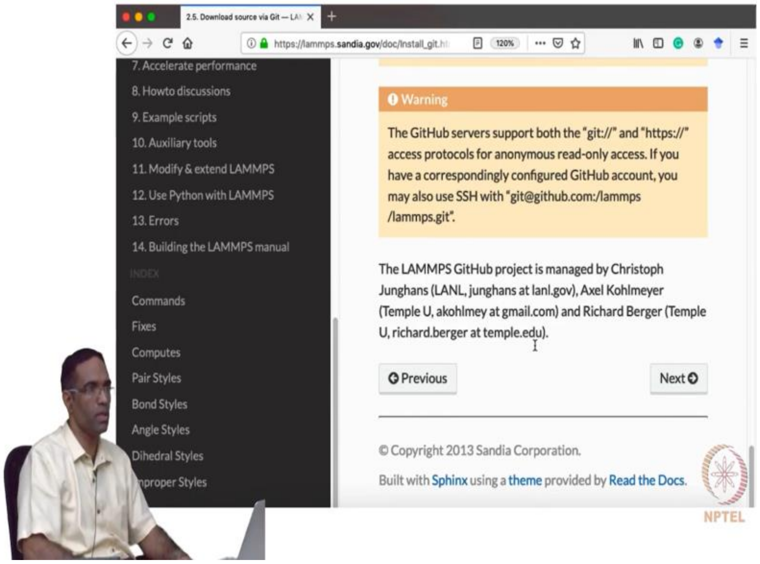


2.5. Download source via Git

All LAMMPS development is coordinated through the "LAMMPS GitHub site". If you clone the LAMMPS repository onto your local machine, it has several advantages:

- You can stay current with changes to LAMMPS with a single git command.
- You can create your own development branches to add code to LAMMPS.
- You can submit your new features back to GitHub for inclusion in LAMMPS.

You must have Git installed on your system to communicate



Warning

The GitHub servers support both the "git://" and "https://" access protocols for anonymous read-only access. If you have a correspondingly configured GitHub account, you may also use SSH with "git@github.com:/lammps/lammps.git".

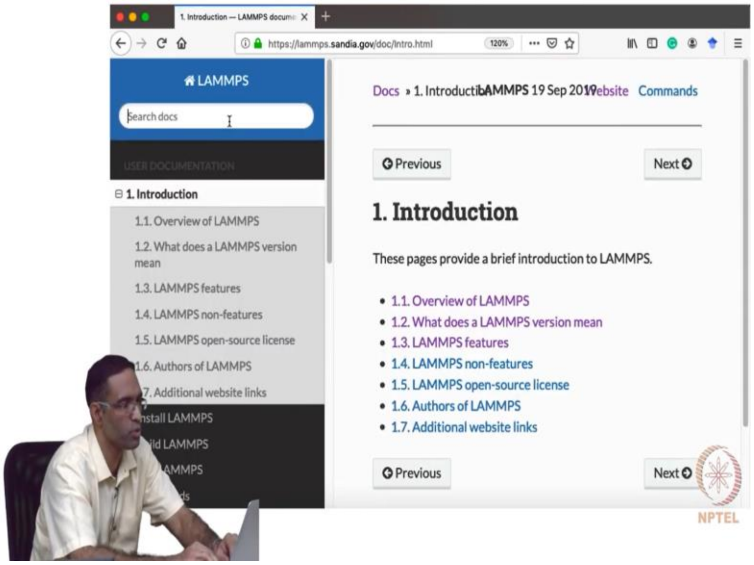
The LAMMPS GitHub project is managed by Christoph Junghans (LANL, junghans at lanl.gov), Axel Kohlmeyer (Temple U, akohlmey at gmail.com) and Richard Berger (Temple U, richard.berger at temple.edu).

© Copyright 2013 Sandia Corporation.

Built with Sphinx using a theme provided by Read the Docs.

I want to source code, so once you download the source code you can use what you refer to as make, a c make, a combination of c make and make to basically to get the executable that is required for your system. It is not required to know all these details for this course but of course if you are doing research and you want to install it on a cluster so that it functions over many processors then you might need all that. For this current purposes that is not necessary.

(Refer Slide Time: 8:16)



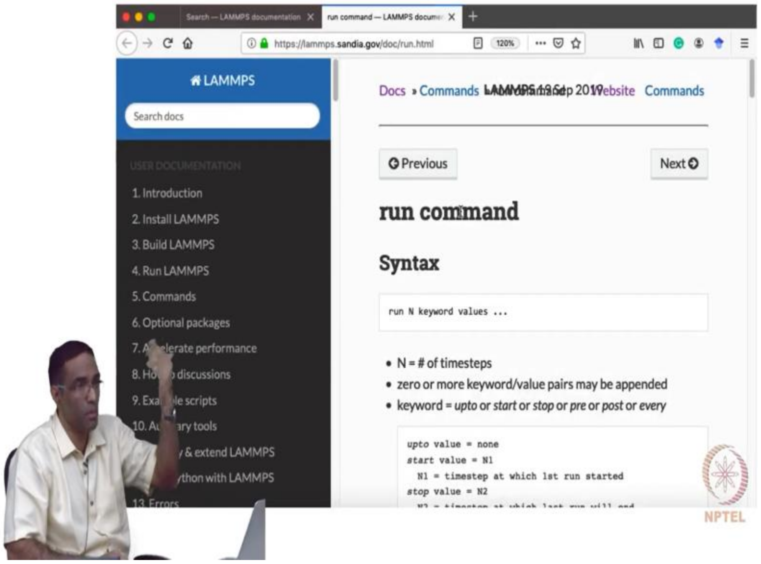
1. Introduction

These pages provide a brief introduction to LAMMPS.

- 1.1. Overview of LAMMPS
- 1.2. What does a LAMMPS version mean
- 1.3. LAMMPS features
- 1.4. LAMMPS non-features
- 1.5. LAMMPS open-source license
- 1.6. Authors of LAMMPS
- 1.7. Additional website links

So the documentation so LAMMPS is pretty useful so you can type an specific commands to see how that functions.

(Refer Slide Time: 8:27)



run command

Syntax

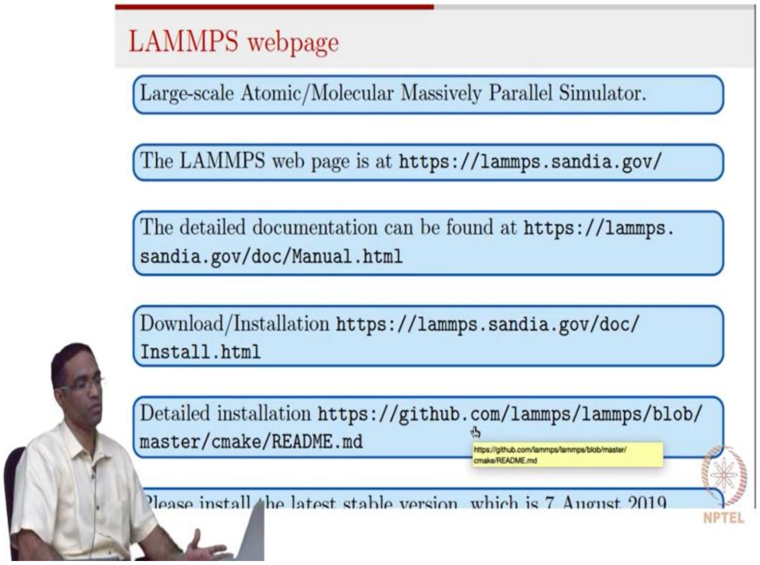
```
run N keyword values ...
```

- N = # of timesteps
- zero or more keyword/value pairs may be appended
- keyword = *upto* or *start* or *stop* or *pre* or *post* or *every*

```
upto value = none
start value = N1
  N1 = timestep at which 1st run started
stop value = N2
```

So for example if I say run, it is going to tell me what the run, it is going to open the page which tells me everything about that particular command. So the whole problem is that there are so many different commands and unless you know what commands to use, you do not know what to type in here. So the basic purpose of this course is to tell you what options are basically available in the software and what each of these does.

(Refer Slide Time: 08:52)



LAMMPS webpage

Large-scale Atomic/Molecular Massively Parallel Simulator.

The LAMMPS web page is at `https://lammps.sandia.gov/`

The detailed documentation can be found at `https://lammps.sandia.gov/doc/Manual.html`

Download/Installation `https://lammps.sandia.gov/doc/Install.html`

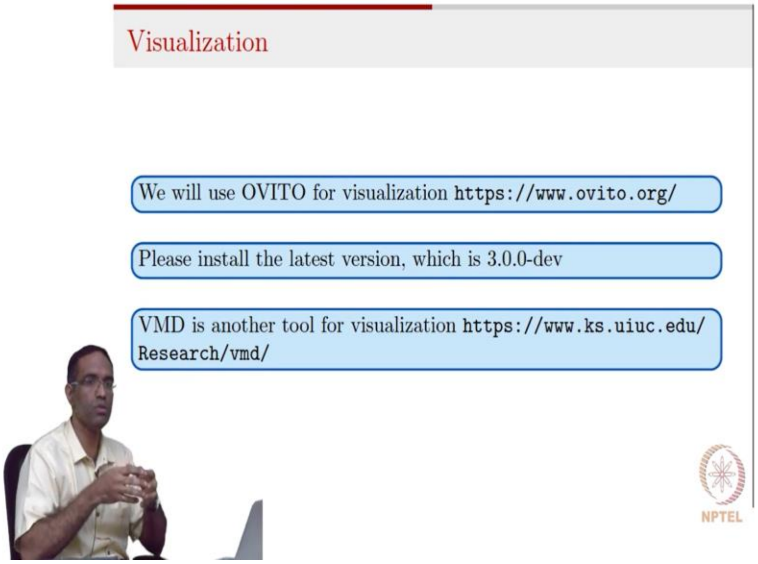
Detailed installation `https://github.com/lammps/lammps/blob/master/cmake/README.md`

https://github.com/lammps/lammps/blob/master/cmake/README.md

Please install the latest stable version, which is 7 August 2019

So let us go back to our power point presentation. This is a very detail insulation that you can get once you have downloaded the it is not opening here so let us not worry about it. In that particular website you can get a fairly detailed instruction as to how you can use c make and make combination to actually install LAMMPS this one right here the one that I am pointing.

(Refer Slide Time: 09:14)



Visualization

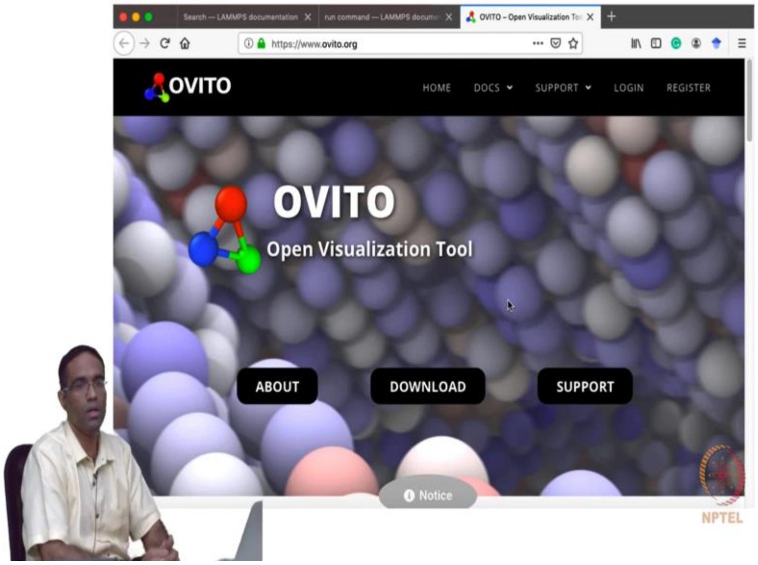
We will use OVITO for visualization `https://www.ovito.org/`

Please install the latest version, which is 3.0.0-dev

VMD is another tool for visualization `https://www.ks.uiuc.edu/Research/vmd/`

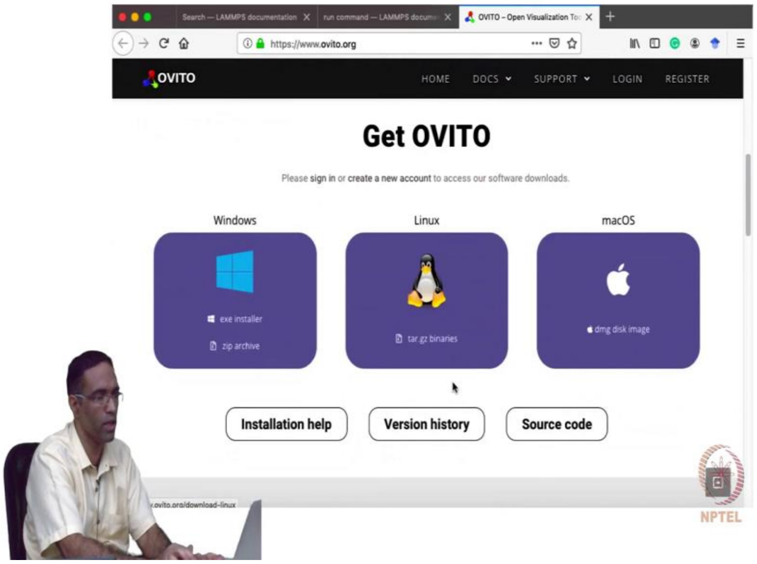
After, one of the most important things after you run the software is to basically visualize your stimulation. For visualization purposes also there are several different softwares available.

(Refer Slide Time: 9:43)



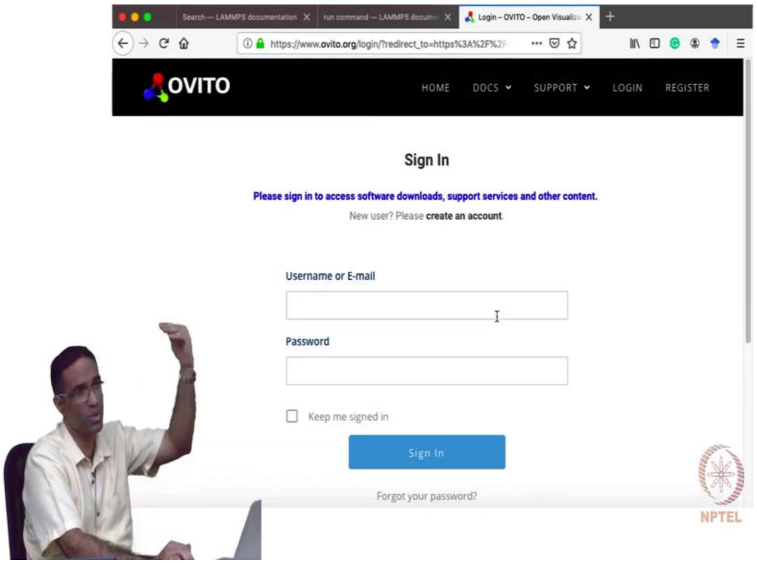
The one that we are going to use is called as OVITO and it has a lot of different functionalities, it is very easy to use and you can also combine it with python scripts to write your own utilities.

(Refer Slide Time: 9:49)



Get OVITO

Please sign in or create a new account to access our software downloads.



Sign In

Please sign in to access software downloads, support services and other content.

New user? Please **create an account**.

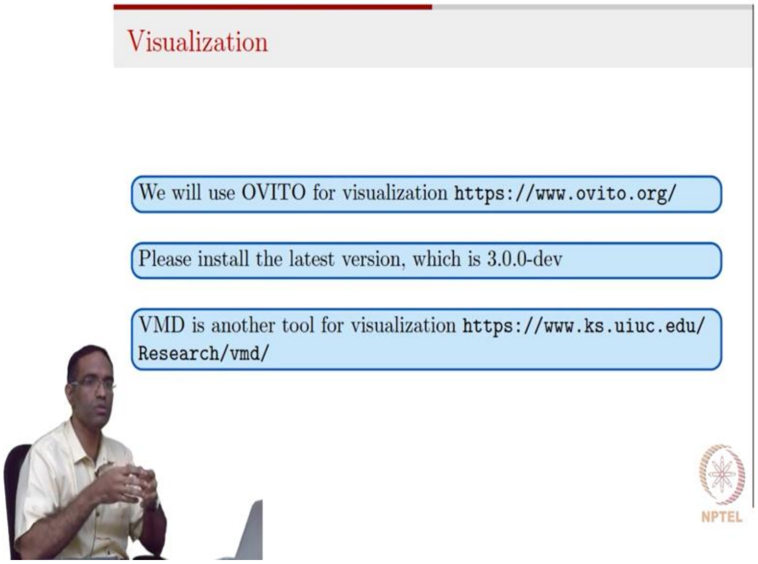
Username or E-mail

Password

☐ Keep me signed in

Sign In

Forgot your password?



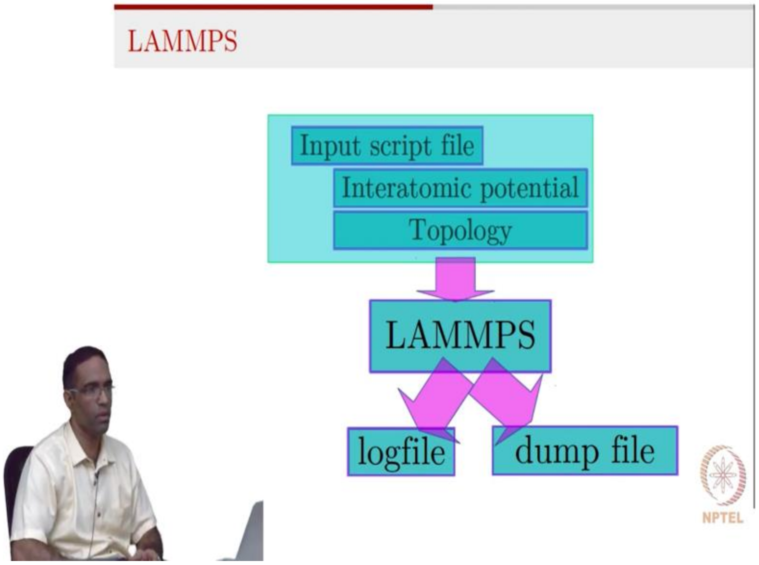
Visualization

- We will use OVITO for visualization `https://www.ovito.org/`
- Please install the latest version, which is 3.0.0-dev
- VMD is another tool for visualization `https://www.ks.uiuc.edu/Research/vmd/`

For example, so I would go to the website click download and choose whatever I want and it so happens that there is a older version, I had a newer and I have actually there is 1 2, there is version which was released in 2017 and one version 3.00 hyphen dev, a developer version which is released in 2019. The one that is on my computer right here is 3.00 the latest version.

So please install that so that we have some compatibility between what you are seeing and what I see, what I am seeing I am going to see. VMD, visual molecular dynamic is another tool that is used for visualization of these atoms and molecules. However, in this course I am going to stick to OVITO but if you want you can actually try that also, that is very pretty good.

(Refer Slide Time: 10:35)


LAMMPS

Input script file

Interatomic potential

Topology

LAMMPS

logfile

dump file

So LAMMPS, what is LAMMPS need? So LAMMPS is a c++ code, a c++ code and it is going to require some inputs and it is going to do something with the input file that you have given to it. And then it is going to spit out some files depending upon what you have asked it to do. There are 2 different kind of files that can go as input to LAMMPS, one is basically of course the sorry there is one input script file and in addition to the input script file you can have a file which is describing the kind of interactions the various atoms in the system are having. Which we will call as the interatomic potential file. And then there is going to be another file which can consist of the positions, velocities of all the atoms that you want to simulate.

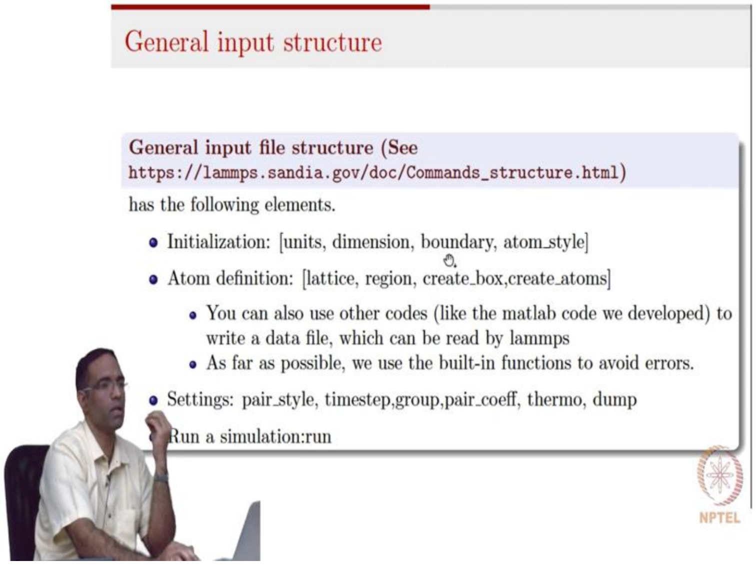
Now the input script can either, what it can do is these two files is interatomic potential file and topology file are generally optional. It depends on the problem that you are trying to solve. It is possible to specify the parameters for the interatomic potential. And also generate the crystal structure from within the input script itself. You do not need always these interatomic potential files and the topology files what I am calling topology file is basically the positions of all the atoms in the system for a crystalline solid and I am your MATLAB code can actually generate that. You can either write the MATLAB code and have the input script read the coordinates from their file or you can have commands in the input script which will the generate the crystal structure itself, we will see that also.

So once you give these two, this information to the input to LAMMPS it reads it understand what to do. And at the end it is going to print out several files and one file is called as the log file, it can be named differently but we will call it the log file which contains thermodynamic information for the entire system, for the entire system, for the, for example the pressure of the system or the temperature of the system, the stress the components of the entire simulation box that you are having all that will be there in the log file. And you can choose from the input script what you want to printout and what you do not want to printout in the log file.

Dump file is something that will contain per atom information like for example atom coordinates as a function of time or atom velocities as a function of time or per atom stresses as a function of time. What it means is something we will discuss later but per atom stresses or volume of an atom or something like that. All these quantities you can calculate and print it out into a file called as the dump file. And you can choose for in the input script the format in which you want these things to be printed out. So we will actually look at all of these things one by one slowly but overall LAMMPS will take in general it can take a basic input script a file which contains information on the interatomic potential and a file that contains information on the topology.

So this the word topology is generally used when we are talking about system like a polymer system where in addition to the atom coordinates you also need to specify how these atom are actually bonded. For example, there might be a chain of atoms and all these atoms are bonded together and there might be another chain. This atom or the second chain and the atom on the first chain are not actually bonded. So you can specify all this information in the topology file. For the kind of stuff that we are going to do in class, we are not going to be deal with polymers we will generally deal with metals and ceramics and so forth. So for such cases it is sufficient if we have the file containing only the positions of all the atoms in the system.

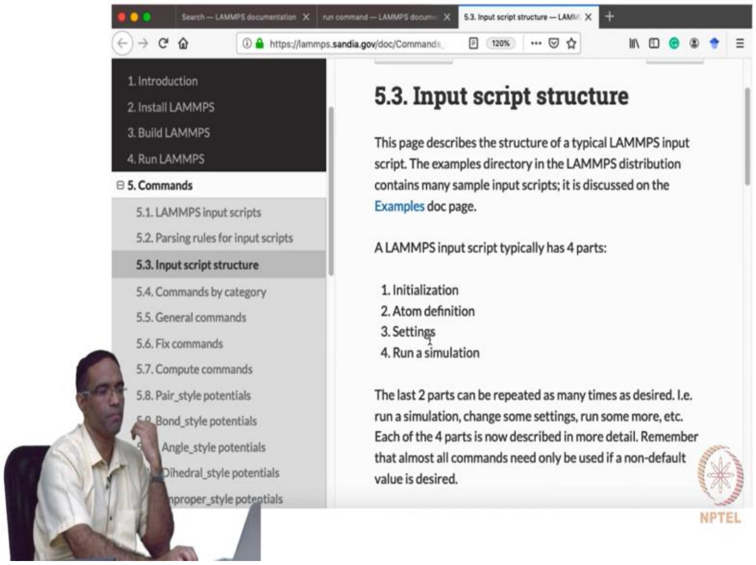
(Refer Slide Time: 14:48)



General input structure

General input file structure (See `https://lammps.sandia.gov/doc/Commands_structure.html`) has the following elements.

- Initialization: [units, dimension, boundary, atom_style]
- Atom definition: [lattice, region, create_box,create_atoms]
    - You can also use other codes (like the matlab code we developed) to write a data file, which can be read by lammps
    - As far as possible, we use the built-in functions to avoid errors.
- Settings: pair_style, timestep,group,pair_coeff, thermo, dump
- Run a simulation:run



5.3. Input script structure

This page describes the structure of a typical LAMMPS input script. The examples directory in the LAMMPS distribution contains many sample input scripts; it is discussed on the Examples doc page.

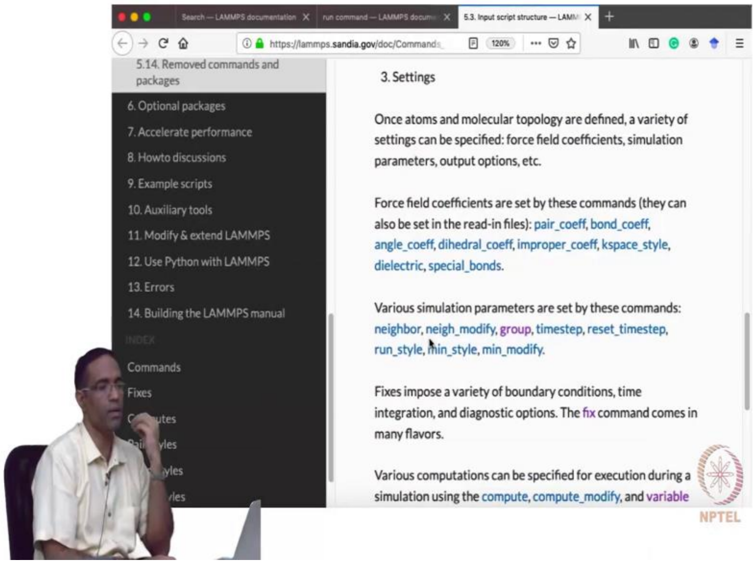
A LAMMPS input script typically has 4 parts:

1. Initialization
2. Atom definition
3. Settings
4. Run a simulation

The last 2 parts can be repeated as many times as desired. I.e. run a simulation, change some settings, run some more, etc. Each of the 4 parts is now described in more detail. Remember that almost all commands need only be used if a non-default value is desired.

So the general input file structure this is a good place to start. So how what does the input script by now you must have understood that the input script plays a very important role. That is where you are actually telling LAMMPS what to do. So the second you know how to write a good input script, you are basically using LAMMPS there. So the input script structure consists of 4 aspects one is you initialize the system that means you can basically construct your crystal structure for example define the atom for example what is the atom’s mass, what is the charge on the atom so on and so forth.

(Refer Slide Time: 15:40)



3. Settings

Once atoms and molecular topology are defined, a variety of settings can be specified: force field coefficients, simulation parameters, output options, etc.

Force field coefficients are set by these commands (they can also be set in the read-in files): pair_coeff, bond_coeff, angle_coeff, dihedral_coeff, improper_coeff, kspace_style, dielectric, special_bonds.

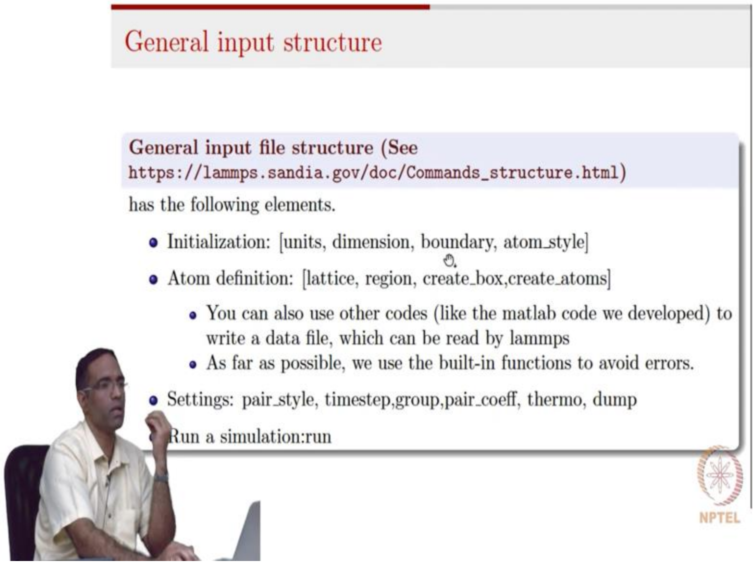
Various simulation parameters are set by these commands: neighbor, neigh_modify, group, timestep, reset_timestep, run_style, min_style, min_modify.

Fixes impose a variety of boundary conditions, time integration, and diagnostic options. The fix command comes in many flavors.

Various computations can be specified for execution during a simulation using the compute, compute_modify, and variable

Then perform a group of settings perform a group of setting like if you have any interatomic potential information that you want to give you can give that. And then you basically decide what you want to do with the system and run a simulation. So there are several different commands that are there which are associated with each of these categories.

(Refer Slide Time: 16:00)



General input structure

General input file structure (See https://lammps.sandia.gov/doc/Commands_structure.html)

has the following elements.

- Initialization: [units, dimension, boundary, atom_style]
- Atom definition: [lattice, region, create_box,create_atoms]
  - You can also use other codes (like the matlab code we developed) to write a data file, which can be read by lammps
  - As far as possible, we use the built-in functions to avoid errors.
- Settings: pair_style, timestep,group,pair_coeff, thermo, dump
- Run a simulation:run

Initialization, atom definition, setting and running a simulation and some of these commands are listed here, units, dimension, boundary, atom cells are all key words that are used in LAMMPS to define certain things. Similarly, atom definition, lattice, region, create box, create atoms all these commands or key words are used to do certain things that is associated with the definition of the structure that you are dealing with.

You can also use the other codes that I gave and have input script read from these. It is not necessary that you always need to use these commands. But in my opinion as far as possible you should try to use the commands that are already there build in LAMMPS so that you do not have to worry about making errors in your input script that you basically give to LAMMPS as far as possible. Pair style, time step, group, pair coefficients, the coefficients of the interatomic potential all these things will basically perform certain settings and help you around a specific kind of simulations.

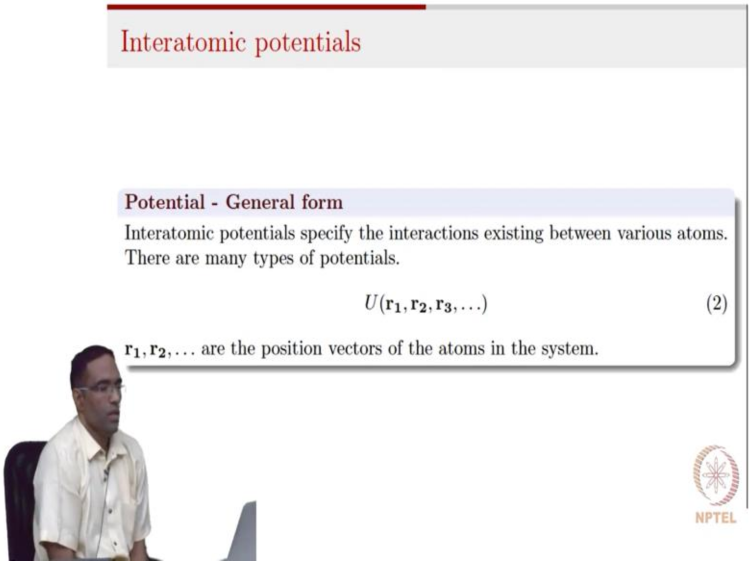
So dump for example is a command that will help you, dump out throw the information like what information, you want the information of all the atoms and the velocities of the system then the command is dump. These all, all these things may seem abstract to somebody who has never done any, written any input script for LAMMPS. But this is just an introduction we are going to see many of these in good detail.

So before we begin looking at an input script I think it important to understand a few of the details containing concerning the interatomic potential. So the interatomic potential is a very important input that actually goes into any molecular dynamic simulations, simulation software. So what interatomic potential defines is how these atoms are interacting that is it. The total energy of the system is always equal to the kinetic energy plus the potential energy.

The kinetic energy is easily calculable, can calculated as just half mv square summed over all the atoms in the system. However, the potential energy is very complicated, it involves analyzing how the energy of the system changes when the atoms are coming close together and going away from each other. And these have significant electronic contributions to that because when atoms are so nucleus is there and then it is surrounded by an electron cloud. So when the atoms are coming close together there is going to be a significant repulsion because of the electron, electron-electron cloud repulsion is going to take place.

This force has to be modelled in properly so actually capture the interatomic potential energy accurately. The other form of energies that is existing between the ions such as say columbic interaction. You already have a model for them, q1, q2 by r square is basically a model that captures the electronic interaction long range interactions that is existing between atoms in a system just because of their charges. However, for this nuclear interactions we have to develop proper models that captures the systems properly capture the behavior of the systems properly. So this interatomic potential is extremely crucial for any molecular dynamics software.

(Refer Slide Time: 19:49)



Interatomic potentials

Potential - General form

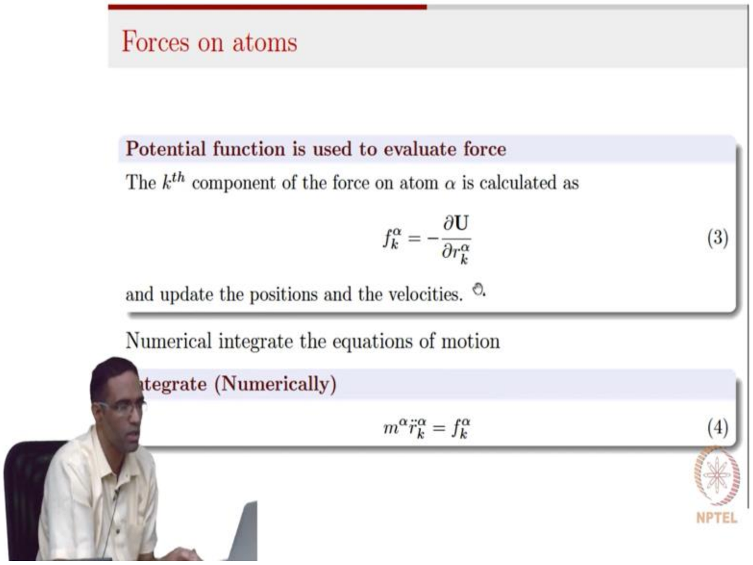
Interatomic potentials specify the interactions existing between various atoms. There are many types of potentials.

$$U(\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3}, \ldots) \tag{2}$$

$\mathbf{r_1}, \mathbf{r_2}, \ldots$ are the position vectors of the atoms in the system.

So basically, there are many different types of potentials but let us look at one general way of writing interatomic potential. The interatomic potential is obviously a function of the positions of all the atoms in the system. Let us look at some of the features of this interatomic potential before we actually start running in the simulation. So U is a function of r1, r2, r3 and so on where r1, r2 and r3 are all the Cartesian positions of the atoms that is there in the system.

(Refer Slide Time: 20:26)



Forces on atoms

Potential function is used to evaluate force

The k^{th} component of the force on atom α is calculated as

$$f_k^\alpha = -\frac{\partial \mathbf{U}}{\partial r_k^\alpha} \quad (3)$$

and update the positions and the velocities.

Numerical integrate the equations of motion

tegrate (Numerically)

$$m^\alpha \ddot{r}_k^\alpha = f_k^\alpha \quad (4)$$

How is this interatomic potential useful? The derivative of the interatomic potential with respect to the position is actually the force on the atom. And once you know the force on the atom you can use mx double dot is equal to that force and solve this problem numerically to actually calculate the positions and the momenta of all the particles as a function of time. So this interatomic potential is essential to actually solve the numerical equations. You actually have to take a derivative of the potential with respect to the distance to calculate the force or the position with respect to calculate the force. And use that in this equation right here equation number 4 right here and solve this numerically in order to update your positions with time.

So this expression U may be a function of r1, r2, r3 so what this expression says is the k th component of the force on the alpha th atom is nothing but minus du U by du r alpha k where alpha is again a label for the atom and k is basically the component. So this is why this is how the interatomic potential is used in a molecular dynamic simulation software. Of course you will also compress of the columbic interaction in case you are talking about a charged system. Otherwise it is just the nuclear interaction that is existing between the atoms. When the atoms there are not ions, any doubts so far?

Student: U is modelled by..?

Professor: U is?

Student: We are calculating mostly the force.

Professor: Yes.

Student: So LAMMPS will calculate the force and…

Professor: If you have an explicit functional form for U in terms of r then you just have to take the derivative of U with respect to r to calculate the force. Correct but there are some, there are some complications there I will come to it in a little bit. Yes, you are right, so have you have to know explicit functional forms for this potentials. So what are those functional form and what can they be functions of, is something that we need to look at. The first thing is...

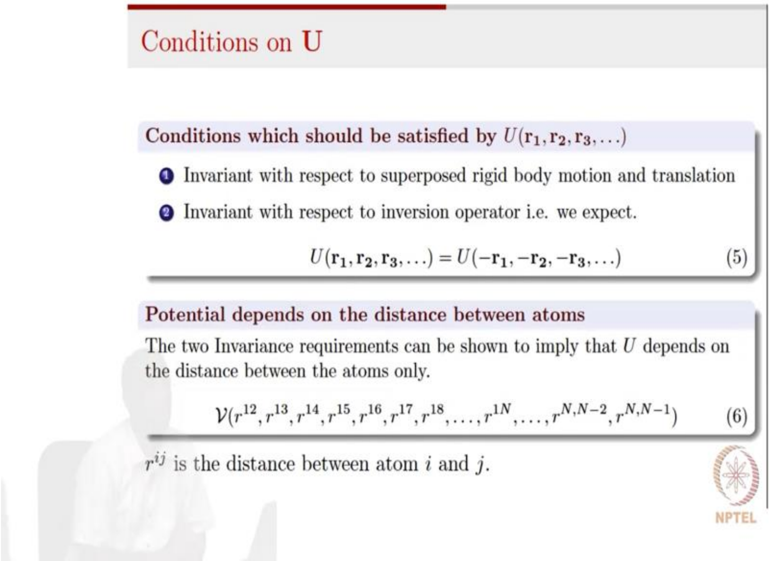
Student: Could you go back? So equation 3 according to equation 3 then fx will be simply minus delta u by delta x.

Professor: delta x?

Student: I am asking that, is it correct to say x component of force is minus del u by del x fy y component of force is minus del u by...

Professor: Of that atom, that means U is a function of, say U is a function of the positions of all the atoms then the force you need to calculate the force on the alpha th atom, force means x component of force y component of force and z component of force. So the force is nothing but negative of the derivative of the interatomic potential energy with respect to the position of the, if you want to calculate fx on the alpha th atom it is derivative of the, derivative of U with respect to the x coordinate of the alpha th atom.

(Refer Slide Time: 23:47)



Conditions on U

Conditions which should be satisfied by $U(\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3}, \ldots)$

1. Invariant with respect to superposed rigid body motion and translation
2. Invariant with respect to inversion operator i.e. we expect.

$$U(\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3}, \ldots) = U(-\mathbf{r_1}, -\mathbf{r_2}, -\mathbf{r_3}, \ldots) \quad (5)$$

Potential depends on the distance between atoms

The two Invariance requirements can be shown to imply that U depends on the distance between the atoms only.

$$\mathcal{V}(r^{12}, r^{13}, r^{14}, r^{15}, r^{16}, r^{17}, r^{18}, \ldots, r^{1N}, \ldots, r^{N,N-2}, r^{N,N-1}) \quad (6)$$

r^{ij} is the distance between atom i and j.

So let us look at this, the whole problem is, if you have U explicitly as a function of the actual positions of various atoms on the systems and it appears that way. That means the functional form is only a function of the actual positions of the atom. Then what does it mean? It means that if the body is actually moving rigidly like this or rotating rigidly like this then the energy changes. That is what it means.

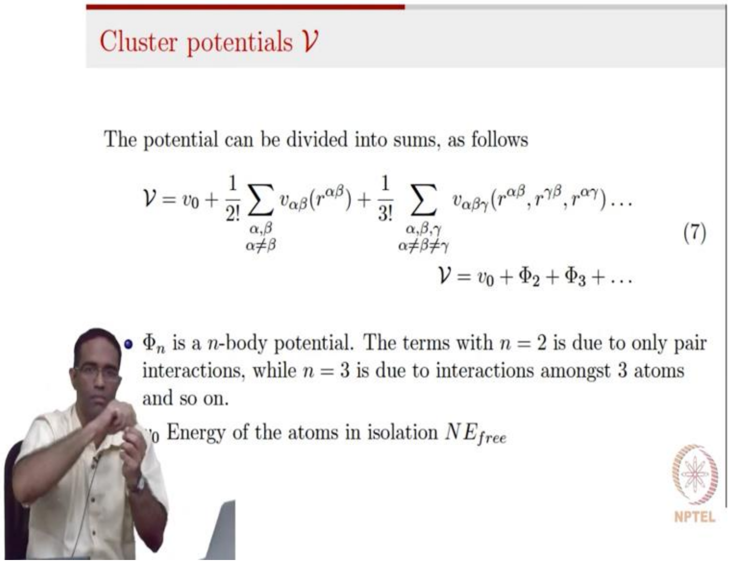
If U is a function of the actual position of all the atoms comprising this body then if this body moves rigidly either rotates rigidly or moves displaces rigidly. The value of U changes because r1 r2 r3 are all changing. But should it change? They should not change, right? Therefore, this U, the functional form of U must be such that it is invariant with respect to superposed rigid body motion or translation, it has to be.

The second issue is, it must also be invariant with respect to the inversion operator, in the sense we also expect that if you simply change your coordinate system from xyz to minus x minus y minus z, then which essentially amounts to saying replacing r1 r2 r3 with minus r1 minus r2 minus r3. The distances between these r, the system looks exactly the same and we should not see any change in the potential energy of the system.

So these two conditions have to be satisfied by this interatomic potential function U. So what happens is, mathematically if you try to show under what conditions these two conditions are satisfied, it so happens that the potential energy of the systems depends only on the distance existing between the atoms. Therefore, I have written down explicitly in equation 6 another potential form V, where V is a function of r12 r13 r14 r15 and so on till r1n, where r13 is basically the distance between atom 1 and atom 3.

But not just that, it has to be a function of also r21 r22 r23 and so on. And r and so on till rn to rn minus 1. It is only dependent upon that distance that is existing between the atoms and not necessarily and not exclusively on the positions of these atoms. Therefore, interatomic potentials that we will deal with will only have the scalar distance quantity r, which is between two different between two different atoms. And that is because of the requirement of the conditions in the two conditions that we just discussed.

(Refer Slide Time: 26:57)



Cluster potentials $\mathcal{V}$

The potential can be divided into sums, as follows

$$\mathcal{V} = v_0 + \frac{1}{2!}\sum_{\substack{\alpha,\beta \\ \alpha\neq\beta}} v_{\alpha\beta}(r^{\alpha\beta}) + \frac{1}{3!}\sum_{\substack{\alpha,\beta,\gamma \\ \alpha\neq\beta\neq\gamma}} v_{\alpha\beta\gamma}(r^{\alpha\beta}, r^{\gamma\beta}, r^{\alpha\gamma})\ldots \tag{7}$$

$$\mathcal{V} = v_0 + \Phi_2 + \Phi_3 + \ldots$$

- Φ_n is a n-body potential. The terms with $n = 2$ is due to only pair interactions, while $n = 3$ is due to interactions amongst 3 atoms and so on.
- v_0 Energy of the atoms in isolation NE_{free}

So now it is actually a still, it still remains a complicated problem, because you were talking about the distance between an atom here and atom there. And they are interacting is what you are saying, you are saying that in general you must actually depend on these distances. It so happens that the interatomic potential can actually be divided into sums that is shown as follows.

So, V is nothing but V naught plus 1 by 2 factorial times sum over only pairs of atoms. This (atom), the energy of this atom depends only on the distance between another atom, the another atom, another atom and so on. Basically, the second term does not account for the fact that the presence of a third atom can actually influence the interaction existing between two other atoms. Therefore, the second atom is actually called a two body potential.

The third or a pair potential, the third term incorporates the effect of the presence of a third atom on the influence between any two atoms. Similarly, you can actually have the entire interatomic potential written down as a sum of several n body potentials. The last one will be the how the presence of all the other, all the atoms is influencing the energy on one atom. So, V naught I did not talk about that, is basically the energy of the atoms in isolation which is nothing but, if you have n atoms in the system, it is n times energy of the free atoms.

So that is basically some sort of a base, we do not we usually let that go to 0 and we are only worried about the other terms that exist in this interatomic potential. Now this problem is still a little bit intractable, so how are we going to narrow it down. We do not use five body potentials and six body potentials very often. We generally stick with these two body potentials or pair potentials where we say that only interaction between an atom and another atom in terms of this distances actually is important. So…

Student: Sir?

Professor: Yeah?

Student: Sir what is V0? Could you explain what it is?

Professor: V0 is the energy of an atom in isolation. In vacuum is there is something called its energy, is that energy.

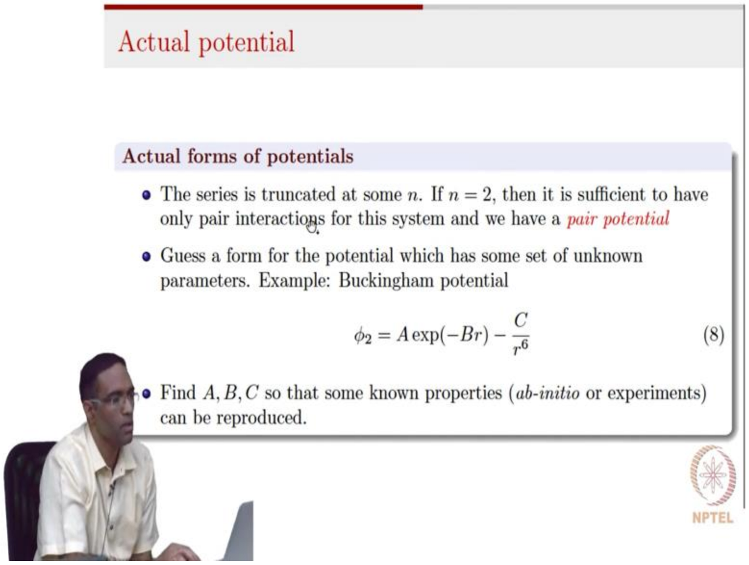
Student: Sir but, when an atom is existing in isolation, it does not have any potential energy like potential energy comes into the picture only when it starts interacting with…

Professor: So now see it may have some energy because of the presence of the see the nucleons or the electrons and all these things might give it some energy no, 0 energy may be there for…

Student: Okay, so due to the subatomic particles themselves?

Professor: Yes, they may be it may have some energy. So that is its energy, the energy of the free atom, is a put everything together in an Ab-initio software and calculate the energy of just one atom, what is it? But, yeah, but right now you, in MD simulations we really do not require that, because we are always take the derivative of this V with respect to r and that is a constant it goes away. So that can be treated as some sort of a base with respect to which we are calculating other energies.

(Refer Slide Time: 30:36)



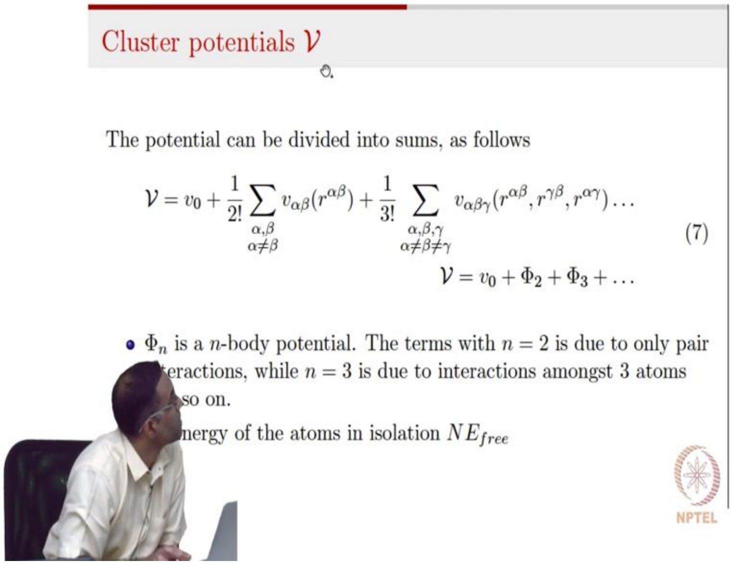
Actual potential

Actual forms of potentials

- The series is truncated at some n. If $n = 2$, then it is sufficient to have only pair interactions for this system and we have a *pair potential*
- Guess a form for the potential which has some set of unknown parameters. Example: Buckingham potential

$$\phi_2 = A\exp(-Br) - \frac{C}{r^6} \qquad (8)$$

- Find A, B, C so that some known properties (*ab-initio* or experiments) can be reproduced.



Cluster potentials $\mathcal{V}$

The potential can be divided into sums, as follows

$$\mathcal{V} = v_0 + \frac{1}{2!}\sum_{\substack{\alpha,\beta \\ \alpha\neq\beta}} v_{\alpha\beta}(r^{\alpha\beta}) + \frac{1}{3!}\sum_{\substack{\alpha,\beta,\gamma \\ \alpha\neq\beta\neq\gamma}} v_{\alpha\beta\gamma}(r^{\alpha\beta}, r^{\gamma\beta}, r^{\alpha\gamma})\ldots \qquad (7)$$
$$\mathcal{V} = v_0 + \Phi_2 + \Phi_3 + \ldots$$

- Φ_n is a n-body potential. The terms with $n = 2$ is due to only pair eractions, while $n = 3$ is due to interactions amongst 3 atoms so on.

nergy of the atoms in isolation NE_{free}

Actual form of potentials, so there are many different kinds of interatomic potentials that people have used in order to model various systems. Like I mentioned this second term that is existing here, where the interatomic potential only depends upon pair interactions. Only between the atom and another atom, and is not influenced, which is not influenced due to the presence of the third atom is not considered, is called as the pair potential. And so that is one form of interatomic potential. So generally what happens is you have to guess how do you build these potentials.

So, you have to guess some sort of a functional form for these interatomic potentials based on some physical intuition. And when you have some functional form for these interatomic potentials, you will come up with you will have certain unknown parameters for example, in this case the potential refer to us the Buckingham potential has unknown A, unknown B, unknown C and r is basically the distance between the atoms, pairs of atoms. So you try to fit the values of A, B and C, so that some property of the material that you are studying can be reproduced. So these properties are either generated through ab-initio calculations or through experiments, for example, the… Yeah?

Student: Sir when you are saying that potential, are you reducing the number of atoms down to two, or are you ignoring or truncating the extra like the p t…

Professor: I am truncating it, I am not looking at these other terms these higher terms. I am saying that this is sufficient for me, because any other thing does not happen.

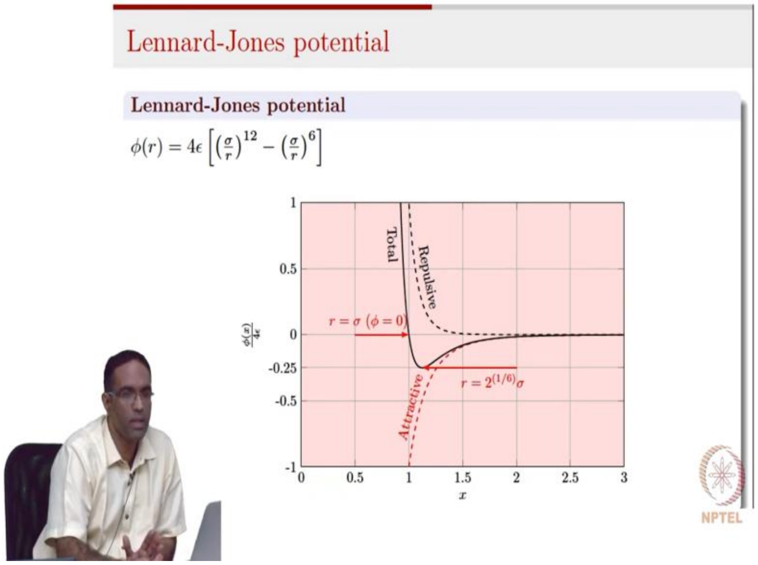
Student: Assuming it that does not happen?

Professor: The assuming that these are the terms have no influence they are very very small.

Student: Sir?

Professor: But that is not true for some systems. So the first potential that we can what I, that I want to talk about. So there are many different functional forms for these potentials, many different people come up with the large number of functional forms and that can be a, maybe we can talk about just that for about five six classes. But they are just different functions they are just different functions of r, they will have different parameters A B C D and so on, and all of these are basically given to you by other people which you can use for that particular material.

(Refer Slide Time: 33:15)



Lennard-Jones potential

Lennard-Jones potential

$\phi(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$

So, but some aspects of these potentials are important. So Lennard-Jones potential is a potential that you are all I think familiar with. So it is again a same potential like the Buckingham potential or is basically the distance between the atoms sigma and epsilon are the parameters that you need to fit, so that you are able to reproduce some of the material properties. Lennard-Jones has generally used for it, was used to actually predict the properties of noble gasses at extremely low temperatures.

So basically, solid argon, so on and so forth. It so happens, it so happened that the 1 by r power 6 can be shown that the attractive force is actually in the the order of 1 by r power 6. So they and that right here and in order to make it look good they just know that, you know the repulsive part of the potential must be extremely steeping, much more steeper than the attractive part. And therefore, just for aesthetic appearance they basically made this 12.

There is no more fundamental reason that that for this being a 12 6 potential. And the second use is the second thing is Lennard-Jones is not different people it is just one guy. So there are several different interesting things that will come out. So this is just these are just model systems, they may not be representing reality in any way. You have to evaluate your material very carefully before you actually make any conclusions.

But we will show in some of the exercises that for argon this is pretty good and we can see that PV is equal to rT when we use a potential that we know for argon. So let us go with this potential carefully, so there is a, the potential if it is plotted it looks like this the black one which I have marked total. The part which says attractive is the one that is going down which means as I take the atom far away, take two atoms far away from each other than the attractive forces go like that.

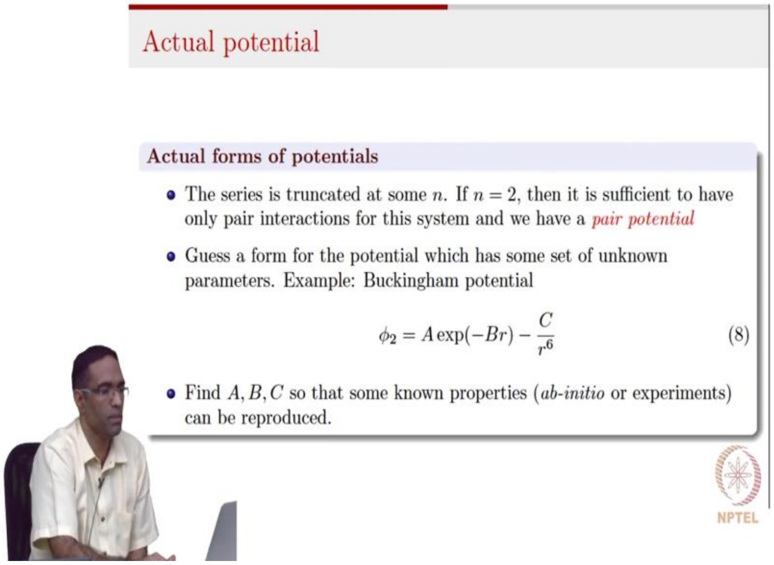
And they go as 1 over r to the power 6 and apparently there can be shown explicitly for noble gases for atoms which have closed shells you can show that. Then the other part is the one that is extremely steep which is sigma by r 1 by r to the power 12. So if you add these two you will get this total energy. This particular spot at which the energy crosses 0 is the value for which r is equal to sigma that is nothing but sigma r power 12 is equal to sigma by r power 6 for which sigma is equal to r.

And this value of r at which the energy is a minimum is having a value of r equal to 2 to the power 1 by 6 times sigma. In this case sigma is been chosen to be 1, but for argon or neon or krypton depending upon the material or the noble gas what you are talking about, the values of sigma and r vary and it has been determined, you can find it everywhere, we will do some exercises. Now the question is now how you can use this potential for other uses.

For example, when we are modelling polymer systems. So you have a chain of atoms and there is a bond that is existing between two atoms. And there is another chain of atoms and there are many monomers and atoms that are interacted are bonded here. In order to model the interaction between an atom here and atom here, which is not bounded where the only interaction is a weak Van der Waals forces. Also people use Lennard-Jones type of potential.

So in this manner these interatomic potentials while they have been concealed to be used for one specific purpose. They have also been used in very creative ways to actually model other systems. So that is with the respect to the Lennard-Jones potentials. I will not be talking about the form of the, I will be talking about the explicit the algebraic form of the potential, but we will not be looking at figures of this sort for other potentials unless and other it is really required. But generally the potential will consist of attractive part and repulsive part.

(Refer Slide Time: 38:01)



Actual potential

Actual forms of potentials

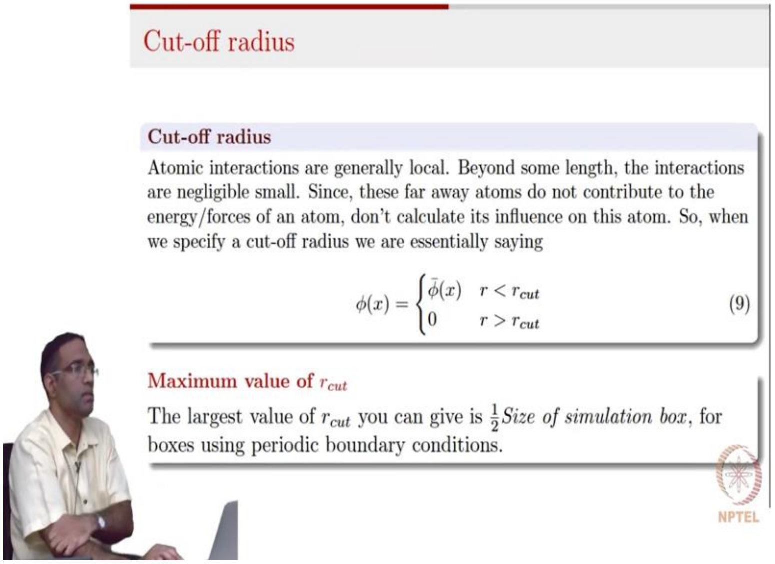
- The series is truncated at some n. If $n = 2$, then it is sufficient to have only pair interactions for this system and we have a *pair potential*
- Guess a form for the potential which has some set of unknown parameters. Example: Buckingham potential

$$\phi_2 = A\exp(-Br) - \frac{C}{r^6} \tag{8}$$

- Find A, B, C so that some known properties (*ab-initio* or experiments) can be reproduced.

This is also the same thing except that here you have C by r power 6 and A exp minus B by r is the repulsive part, it is the same thing. This is the first part is a repulsive part and the second part is basically the attractive part. How strong it is how its functional form is its all left to imagination as long as it satisfies these basic properties you are you can fit any form of the interatomic potential that you want for your system. But that is not very easy that being said but that is not very easy. Are there any questions? Very clear absolutely clear very good.

(Refer Slide Time: 38:43)



Cut-off radius

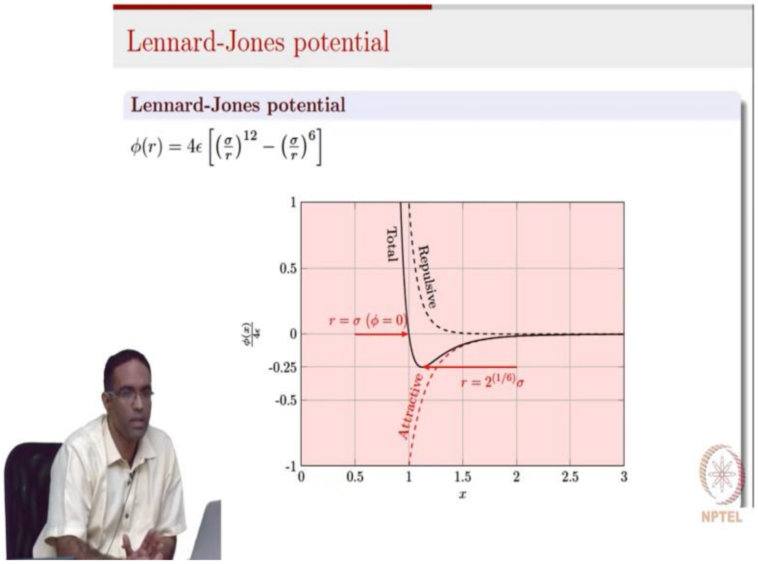
Cut-off radius

Atomic interactions are generally local. Beyond some length, the interactions are negligible small. Since, these far away atoms do not contribute to the energy/forces of an atom, don't calculate its influence on this atom. So, when we specify a cut-off radius we are essentially saying

$$\phi(x) = \begin{cases} \bar{\phi}(x) & r < r_{cut} \\ 0 & r > r_{cut} \end{cases} \tag{9}$$

Maximum value of r_{cut}

The largest value of r_{cut} you can give is $\frac{1}{2}$*Size of simulation box*, for boxes using periodic boundary conditions.



Lennard-Jones potential

Lennard-Jones potential

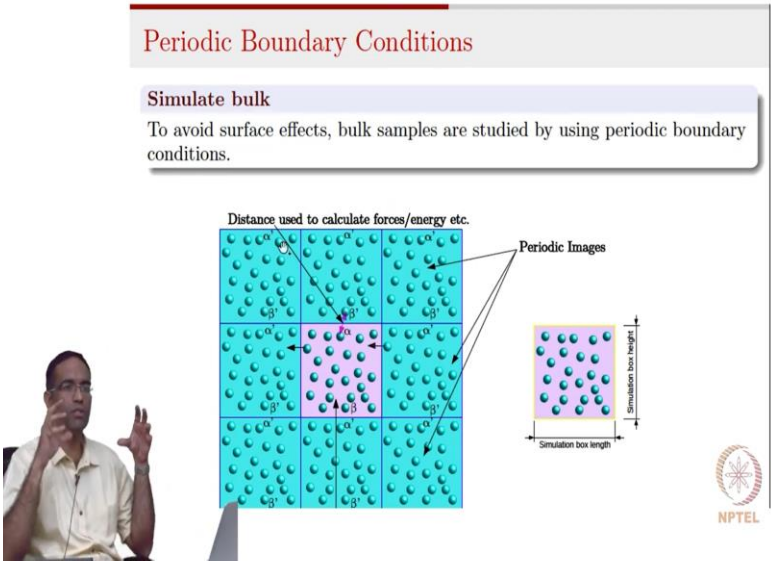
$\phi(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$

So now a third term that you will often encounter when you are doing molecular dynamic simulation is something called as the cut off radius. Have you heard of that term at least? You used it? So when you are talking about the interaction of the effect of the force on atom A due to some other atoms say may pair interaction. You know that the distance is so large that it is actually not going to affect it all. For example, take a look at this figure right here, beyond the value of x or r should be r beyond the value of r greater than say 2.2 the potential is basically 0.

There is a absolutely no effect so it is unlikely that any atom which is greater than 2.5 Armstrong or 2.5 units from the current atom that I am sitting on affects this atom is very very small, it is not going to make any difference. So why unnecessarily make the computer actually calculate this force, you do not need to have the computer calculate this force. So what you do is you specify a cut off radius, you say that any atom consider only atoms that are surrounding a given radius to actually influence the force.

I know that so many atoms are influencing the energy of the current atom that I am sitting on so I do not want to consider atoms that are beyond this radius and therefore I truncate the number of, truncate this sphere over which I calculate the force on given atom. And all the other atoms are not included in the calculations because of which the computational time is significantly reduced. That is the whole idea of cut off so you have give, be careful about the, be careful about the giving the cut off radius. The largest value of the cut off radius that you can get is actually half the size of the simulation box. Especially when you are using periodic boundary conditions and what is periodic boundary condition?

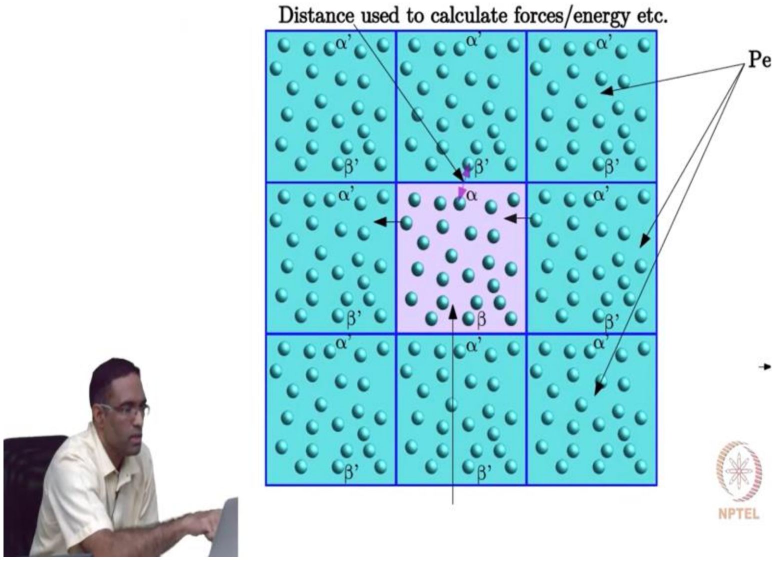
(Refer Slide Time: 40:57)



Periodic Boundary Conditions

Simulate bulk

To avoid surface effects, bulk samples are studied by using periodic boundary conditions.



Periodic boundary condition is this is a following idea. So if you are simulating a bunch of atoms and you want to simulate the bulk behavior. Then you really do not want free surfaces to exist in your simulation cell, you do not surfaces which are exposed to vacuum. You want all of them to be interacting with some atom. So what you do is you enforce a boundary condition refer to as a periodic boundary condition which essentially means so let us take a look at this picture right here, the central box that is shown here is actually the simulation box in which the atoms are all vibrating, in which you created your system.

And then when you enforce periodic boundary condition and there is a lot of coding that goes into it but in LAMMPS you just have to give some simple command. When you enforce periodic boundary conditions what essentially, what essentially LAMMPS does is, if an atom is actually leaving the main simulation box it is equivalent to the same atom entering from the other side. So when atom goes out like these it is equal to some atom coming in like this. So there is a main simulation box and then there are its periodic images. While this movement of atoms is somewhat okay to understand, what is more important to understand, are the forces that is existing between the atoms.

So let us take a look at this atom alpha right here and this atom beta right here inside the main simulation box and the primed quantities are the corresponding periodic images of the same atom. So there is a same atoms here same is here, same atom is here, so when you want to calculate the force that is existing between the two atoms alpha and beta. If you look in the main simulation box it appears as if there are too far of hence they will not interact with each other but that is not actually what is happening, what is happening is this alpha is interacting with its periodic neighbor beta prime. And that is the way the force on atom is alpha is actually calculated.

However, for this atom it is just this because the corresponding atom here is so far of that it is not going to influence at all. So the manner in which these atoms that are close to the boundary of the box are taken care of is extremely important when you are trying to develop a molecular dynamic simulation software. And when you are enforcing periodic boundary conditions these are the kind of things that LAMMPS is actually doing in order to simulate the bulk behavior of the material.

So this the main simulation box and you will have some simulation box length, height and width and so on this is just a 2d image you can imagine that if you are doing a 3 dimensional system then this the actual simulation box is surrounded by 27 26 including the central one it is 27. So you will solve the 26 periodic images. So this is what we will be using we will be using periodic boundary conditions. You can also use other type of boundary conditions which have been programmed to do various other things like a free boundary condition or a fixed boundary condition and all that.