

Mechanical properties at nano-level

Gemma Safont Camprubí
Lund University
2011

Abstract

Nowadays the recent growing interest for nanotechnology leads us to want to understand the materials at nano-scale.

This thesis aims to deepen into the materials mechanical properties from a molecular point of view. Molecular Dynamics (MD) is the simulation technique that is going to be used to understand molecules behavior.

The software that uses this kind of simulation is called Lammmps and during this thesis we will see how to perform a simulation to extract the elastic constants and other mechanical properties. In order to get a real and useful simulation, a copper nanowire is going to be performed with this program, from which its Young's Modulus is going to be find and used to describe the mechanical properties.

Contents

1. Introduction	4
2. Molecular Dynamics	5
2.1. First assumptions	5
2.2. Interatomical potentials	6
2.3. Equations of motion	8
2.4. How a simulation runs	9
2.5. Limitations of the simulation	12
3. Modeling and simulation	14
3.1. Using LAMMPS	14
3.2. Definition of the nanowire studied	16
3.3. Input Script	17
4. Results and discussion	21
5. Conclusions and future considerations	27
6. References	28

1. Introduction

Every year new technologies that are trying to explain different kinds of phenomena appear. Nowadays with the research and empirical experiments we can describe lots of these phenomena but there are also left some without explanation, because of the methods or the availability of the experimental tools. This is the reason why other techniques are being developed as simulation and modeling methods.

As Richard Feynman once said, if you want to understand life you should be aware that things are made of atoms and everything that living things do can be understood in terms of the movements of atoms. This means that if we are able to understand and model atoms behavior in the material we will know how the material undergoes deformation, phase changes, etc. and why each one has its own properties.

Starting from this idea, atomistic simulation tools have recently been required as a bridge between microscopic molecular understanding and macroscopic continuum engineering. As mechanics is the base of mechanical engineering, nano-mechanics will play an important role in the development and application of molecular mechanical engineering.

With the growing interest for nanoscale technology and the availability of new tools and techniques we are closer to understand the gap between the atomic and the macroscopic world in mechanics and materials. Those tools also help scientists to see the atoms movements in a way which it is impossible in laboratory experiments.

To discover the real potential of nanoelectromechanical systems (NEMS) [1], we need to know more about mechanical properties of metal nanowires because these are the important components of nanoscale devices.

This thesis aims to deepen in the materials mechanical properties from a molecular point of view. A copper nanowire will be studied to find out the mechanical properties using molecular dynamic simulation. Nowadays the most part of the simulations to study mechanical properties were focused on static simulation, but in this thesis another method is going to be used.

Using Molecular Dynamics simulation programs [2,3], we will find out the molecular behavior of different materials when they are exposed to different situations. With this technique we are also going to be able to discriminate between good and bad theories, make experiments that are impossible in the laboratory, test a model comparing with experimental results, etc.

2. Molecular dynamics (MD)

There are two main families of simulation technique that can show us the molecular behavior of the materials which are Molecular Dynamics (MD) and Monte Carlo (MC).

On the one hand, Monte Carlo simulation is a method that involves using random numbers as inputs and probability to solve problems and is often used when the model is complex, nonlinear or has some uncertain parameters. On the other hand, Molecular Dynamics method generates configurations by solving the classical equations of motion for a body system with N particles interacting between themselves.

Molecular Dynamics is a computer simulation based on the physical movements of atoms and molecules using statistical mechanics. It shows a different approach of the material properties that is focused on atoms, the smallest entities. It also can be described as a “virtual microscope” with high temporal and spatial resolution. The MD simulations aim to predict the time-dependent trajectories in a system of interacting particles. The main justification of the MD method is that statistical ensemble averages are equal to time averages of the system, known as the ergodic hypothesis.

This method to study the interactions of hard spheres was introduced by Alder and Wainwright between 1957 and 1959. After that date, the most important advance was made by Rahman and Stillinger, who carried out the first molecular dynamic simulation of a realistic system, a simulation of liquid water.

2.1 First assumptions

To get started in molecular dynamic simulations, there are some basic assumptions that must be taken into account.

First, all the molecules and atoms in the simulation are described as a system of interacting material points. This is because the nuclei are assumed to be much heavier and slower than electrons and the last ones respond instantaneously to any movements of the

nuclei. Hence, the nuclei are the only ones considered as describes the Borh-Oppenheimer approximation. Each of these material points is described using its own vector of positions and velocities that describes its motion.

Secondly, the study of the evolution of this system is done by integrating the classical equations of motion for a desired number of timesteps or until some convergence criteria are met. The procedure that uses Molecular Dynamics consists in those steps:

- Firstly the initial positions and velocities of every atom are described.
- The forces between these atoms are calculated by using the interatomical potentials.
- Once the forces are known, a small time interval happens, and the initial atomic positions and velocities change to a second state.
- When the new positions and velocities are known in this first step, the forces are calculated again and the procedure is repeated until the end of the simulation.

The last but not the least is that there are no mass changes in the system during the simulation, thus, no atoms are entering or leaving the simulation box. Hence, the total energy of the system must remain constant.

2.2. Interatomical potentials

As mentioned above, Molecular Dynamic simulations require the definition of an interatomical potential function, which means that a description of the terms by which particles will interact, must be given.

The potential consists in a function of the positions of the nuclei which represents the potential energy of the system when the atoms are located in a specific configuration. It is usually described using the relative positions of the atoms with respect to each other, rather than from absolute positions. The general structure is the one following:

(1)

$$U(r_1, r_2, \dots) = V_1 + V_2 + V_3$$

In this equation, the first term represents the energy due to an external force field, such as gravitational or electrostatic, but is often ignored. The second shows pair-wise interactions

of the particles. The third term gives the three-body components but it is often included in the second term to reduce the computational expense of simulation. The other following terms are interactions between more than three bodies and they are usually not considered.

There are some potentials that consider many-body interactions, thus the potential energy includes the effects of three or more particles interacting between themselves. In some simulations with two-body potentials some multi-body dependence has been incorporated to describe the multi-body interaction. On the other hand, in many-body potentials these interactions are calculated as a combination of higher-order terms. Some examples of the latest are the Tersoff potential which involves a sum over groups of three atoms considering the angles between the atoms, or the EAM potential [4,5], where the electron density is calculated from a sum of contribution from the surrounding atoms.

In molecular dynamics simulation it is difficult to calculate all the interactions between one atom and the other simulated atoms. To reduce the computational effort only the nearest neighbors of each atom are going to be considered instead of the infinite range of the potential function. Assuming this hypothesis a cut-off radius has to be defined as a maximum value of the modulus of the radius vector. This means that only the interactions between the atoms separated by less than R_c are going to be taken into account. It is also usual to define a “skin” factor f_c which will smooth the transition when the value r approaches R_c .

$$V(r) = \begin{cases} \phi_{LJ}(r) - \phi_{LJ}(R_c) & \text{if } r \leq R_c \\ 0 & \text{if } r > R_c \end{cases} \quad (2)$$

The most common values for the cut-off radius used in Lennard-Jones potential are $2,5\sigma$ and $3,2\sigma$.

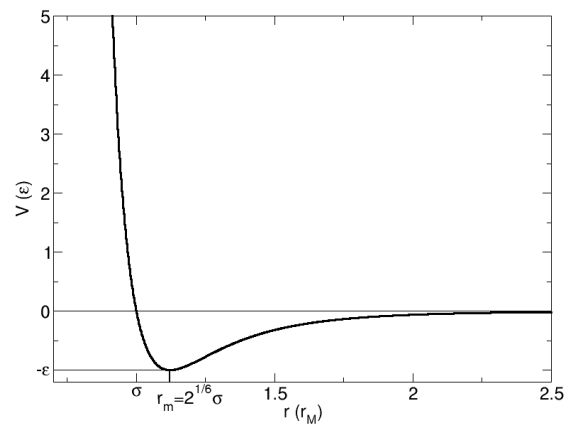
It has to be mentioned that the realism of the simulation depends on the ability of the potential chosen to reproduce the behavior of the material in every simulation performed. In the following section, the most important potential functions are going to be described in order to choose the most suitable for the copper nanowire simulation.

Lennard-Jones potential

The Lennard-Jones 12-6 potential is one of the simplest mathematical model that attempts to describe the interaction between a pair of neutral atoms or molecules. It is used for calculating the Van der Waals forces. The most common expression of the Lennard-Jones potential is the one following:

$$\phi_{LJ}(r) = 4\epsilon \left[\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right] \quad (3)$$

In this equation ϵ shows the bonding/dislocation energy (minimum of the function to occur for an atomic pair in equilibrium), σ is the finite distance at which the potential between two particles is zero, and r is the distance between the particles. As seen in the picture, this potential is strongly repulsive at shorter distance and after reaching the minimum around $1,122 \sigma$, it has an attractive tail for larger r . The parameters ϵ and σ are chosen to fit the physical properties of the material.



The repulsion between atoms when they are very close to each other is modeled by the term $\sim 1/r^{12}$. This is because of the Paulie principle, thus when electronic clouds surrounding the atoms overlap, the energy of the system increases. The second term represents the attractive part, which is more dominating at a large distance. It is originated by the Van der Waals forces, who are weaker but dominate the character of closed shell systems.

However, LJ potential is not adequate to describe covalent bond or metal interaction, where strong localized bond may form or where there is a delocalized electron sea respectively. This potential is more focused on fundamental issues, rather than studying the properties of a specific material.

EAM method

There are some significant problems associated with the application of the pair potentials when the local environment is slightly different from the uniform bulk. This includes problems such as surfaces, grain boundaries, etc.

The *embedded atom method* (EAM) is a multi-body potential usually used for metallic systems. In this argumentation it is considered that each atom is embedded in a host electron gas created by all neighboring atoms. The amount of energy required to insert one atom into the electron gas of a given density is called the embedding function, which considers many-atom effects. In a simulation, the potential energy of an atom, i , is given by

$$E_i = F_\alpha \left(\sum_{i \neq j} \rho_\beta(r_{ij}) \right) + \frac{1}{2} \sum_{i \neq j} \phi_{\alpha\beta}(r_{ij}) \quad (4)$$

where r_{ij} is the distance between atoms i and j , $\phi_{\alpha\beta}$ is a pair-wise potential function, ρ_β is the contribution to the electron charge density from atom j of type β at the location of atom i , and F is an embedding function that represents the energy required to place atom i of type α into the electron cloud.

The same embedding function can be used to calculate the energy of an atom in an alloy and also in the pure material. The pair interaction term is purely repulsive and between two species it can be described as the geometric mean of the pair interaction for individual species.

For a single element system of atoms, three scalar functions must be specified: the embedding function, a pair-wise interaction, and an electron cloud contribution function. For a binary alloy, the EAM potential requires seven functions: three pair-wise interactions (A-A, A-B, B-B), two embedding functions, one for each type of atom, and two electron cloud contribution functions. Generally these functions are provided in a tabularized format and interpolated by cubic splines.

This method has been applied for studying fracture and defects, grain boundaries and other metallic systems and processes.

On the other hand, there are also used some semi-empirical interatomic potential that can deal with more than a million atoms. The atomistic simulations based on the latest option can be useful to analyze and predict defect properties, dynamic behaviors of materials, etc.

2.3. Equations of motion

The simulations carried out by molecular dynamics method are based on solving the classical equations of motion for an N body system interacting through a potential function $V(r_1, \dots, r_N)$. These equations of motion are then given by Newton's laws:

$$m_i \frac{\partial^2 r_i}{\partial t^2} = F_i \quad (5)$$

where F_i is the force of the i th particle and is found from

$$F_i = -\nabla_i V(r_1, \dots, r_N) \quad (6)$$

Lagrangian equations of motion

Written in terms of Lagrangian function L , the equation of motion of a system of interacting particles can be described as the one following:

$$L = \sum_{i=1}^N \frac{m\dot{r}^2}{2} \quad (7)$$

This equation of motion provides identical equations of motion in all coordinate systems. In addition, the interaction between particles needs to be considered too, thus a function of atomic coordinates V can be added in the equation. This function depends on the properties of this interaction and it is known as the interatomic potential interaction.

$$L = \sum_{i=1}^N \frac{m\dot{r}^2}{2} - V(r_1, \dots, r_N) \quad (8)$$

The fact that the potential energy term only depends on spatial configuration implies that if there is any change in its positions, the motion of the particles simulated will change too.

The Lagrangian formulation is commonly used for those molecular dynamics simulations that want to focus on the analysis of detailed atomic motion, rather than obtaining averaged characteristics. The equations of motion can also be written in the Newtonian form:

$$m_i \cdot \ddot{r}_i = - \frac{dV(r_1, \dots, r_N)}{dr_i} = F_i \quad (9)$$

This formulation is more convenient when external forces and constraints are imposed, as well as the post-processing and visualization of the results.

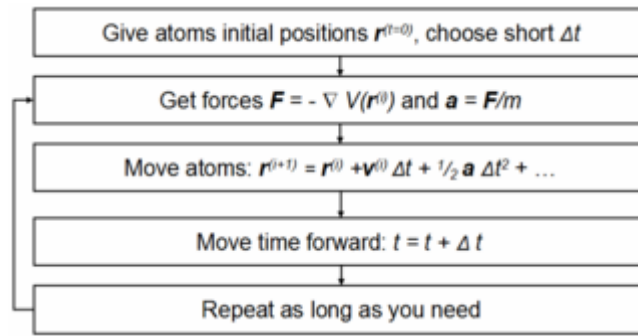
On the other hand, there also exist the Hamiltonian equations of motion whose formulation is more used to study general or averaged features of the simulated systems, such as the energy distribution and thermal flow. In this case, the statistical mechanics methods are employed to describe the state and evolution of a many-particle system.

This kind of formulation is more focused in studying the dynamic evolution of atomic structure and bonds, thus the Lagrangian function is the one that is going to be used in this simulation.

2.4. How a simulation runs

The first step to do in a Molecular Dynamic simulation is to define the system that is going to be simulated. This means that the dimensions have to be established and also the initial position of the atoms. Once these elements are defined and a potential function is selected a time step has to be chosen.

The simulation starts by obtaining the forces between the atoms of the system. Using the time integration algorithm to solve the equations of motion the atoms of the simulation are moved to a new position. Once this new state has been reached, the time is advanced one time step to continue with the simulation. These steps are repeated until the simulation comes to some conditions imposed previously or until it reaches a specified number of time steps.



Time integration algorithm

MD simulation programs use different kind of algorithms to solve the equations of motions, which are explained in the above pages. Time integration algorithms are based on *finite difference methods*, where time is discretized with a time step Δt that is the distance between consecutive points on the grid created. The integration scheme gives the positions and some of their time derivatives at a certain time $t + \Delta t$ once those quantities are known on the previous time t . The time evolution of the system can be followed by iterating this procedure.

However these algorithms are approximate and there can appear some errors associated with them. Basically there are two kinds of errors, the truncation errors and the Round-off errors. The first one is related to the accuracy of the finite difference method to describe the true solution. These methods are based on a Taylor expansion truncated at some term, and that is the reason why are called like this. However, the second kinds of errors are related with the particular implementation of the algorithm, such as the finite number of digits used in computer arithmetic's.

It must be said that both errors can be reduced by decreasing the Δt , but for large Δt the truncation errors are the ones that dominate. On the other hand, round-off errors decrease slower than the truncation ones with a decreasing Δt and dominate in the small Δt limit.

The Verlet algorithm

In molecular dynamics simulation, the most commonly used time integration algorithm is the Verlet algorithm. This consists in writing two Taylor expansions for positions $r(t)$, one forward and the other backward in time. If it is written until third-order there are going to appear the velocities, $v(t)$; the accelerations, $a(t)$; and third order derivatives of the position $b(t)$.

$$\begin{aligned} r(t - \Delta t) &= r(t) - v(t)\Delta t + \frac{1}{2}a(t)\Delta t^2 - \frac{1}{6}b(t)\Delta t^3 + O(\Delta t^4) \\ r(t + \Delta t) &= r(t) + v(t)\Delta t + \frac{1}{2}a(t)\Delta t^2 + \frac{1}{6}b(t)\Delta t^3 + O(\Delta t^4) \end{aligned} \quad (9)$$

Once those expressions are added it results on the following one, which is the basic form of the Verlet algorithm.

$$r(t + \Delta t) = 2r(t) - r(t - \Delta t) + a(t)\Delta t^2 + O(\Delta t^4) \quad (10)$$

As said before, the Newton's equations are the ones that are going to be integrated with this algorithm. Hence, $a(t)$ is just the force divided by the mass, and the force is also function of the positions $r(t)$.

$$a(t) = -\frac{1}{m}\nabla V(r(t)) \quad (11)$$

This algorithm is popular among molecular dynamic simulators because it is simple to implement, and at the same time accurate and stable. The simulation proceeds iteratively by alternatively calculating forces and solving the equations of motion based on the accelerations obtained from the new forces.

However, this version of the Verlet algorithm does not calculate velocities because they are not directly needed, but sometimes it is necessary. Moreover, they are required to compute the kinetic energy K , whose evaluation is necessary to test the conservation of the total energy $E=K+V$. This is usually the most important test to verify that a MD simulation is proceeding correctly. The velocities can be computed from the positions using the following expression.

$$v(t) = \frac{r(t + \Delta t) - r(t - \Delta t)}{2\Delta t} \quad (12)$$

NVE and NVT ensembles

To study properly the evolution of a simulation there are some theoretical tools that must be used. First of all, the main assumption for extrapolating the results obtained in the simulation to the macroscopic system is that time averages are equivalent to the ensemble averages. To assure that, the tool needed is a statistical ensemble, which is composed by an infinite number of microstates (descriptions of possible states). If these microstates are bundled, it can lead us to the same properties in the macrostate.

The NVE ensemble, or microcanonical ensemble, is a kind of statistical ensemble which keeps the number of atoms, the volume and the energy constant during the time integration. Unfortunately, as in most of the experiments happens, the temperature is a parameter that is usually kept constant. To fix this value, the NVE ensemble it is not useful, hence, NVT ensemble must be used. The NVT ensemble consists in using the Nose-Hoover thermostat that is a deterministic method used to keep the temperature around the average.

2.5. Limitations of the simulation

To be able to draw correct conclusions from a simulation, MD must be applied carefully considering his range of validity. It is a powerful technique but has also some limitations that have to be taken into account.

Realism of forces

Another question that can appear while simulating a concrete bulk of atoms is the realism of the forces which describe the interaction between atoms. These atoms move under the action of these instantaneous forces and consequently their relative position and the forces change as well.

The more real is the representation of the interatomic forces, the more accurate will be the simulation. As the forces are obtained as the gradient of a potential energy function, the realism will depend on the ability of the potential chosen to reproduce the behavior of the material.

Time and size limitations

Molecular dynamics simulations are able to perform systems with thousands of atoms, and at the same time with a large space time resolution. In fact, with MD it is possible to focus on a system with times ranging from hundreds of nanoseconds to only a few picoseconds. Despite its possibilities, it also may have some limitations due to the computer efficiency.

Boundary conditions

A Molecular Dynamic simulation can only render a small representative group of atoms of the system under analysis due to the limitation of the computer efficiency. The fact is that rigid or periodic boundary conditions are usually imposed to overcome that problem. However this means that the true effect of the surrounding media is ignored.

Considering that the simulated system will be so small compared with the macroscopic piece of matter, the ratio between the number of surface atoms and the total number of atoms will be different from the reality. That means that most of the atoms will be located on the surface of the domain and will have fewer neighbors than atoms inside. As a result, the surface effects will be more important, and that is the reason why periodic boundary conditions are often used.

The periodic boundary conditions (PBC) consist in the replication of a box where the particles are enclosed. These “image” particles fill all the space but in fact there is only one of the boxes represented in the computer program.

Nevertheless, there is an inconvenient with this assumption because it introduces artificial periodicity into the simulated system.

Time step

The time step is one of the most important parameters in a MD simulations as it may influence how long will last the simulation and also the computational expense. To choose a correct time step there are two criteria to be followed. The first one is that the time step should be small enough so that the simulation is realistic and the atoms motion is continuous. The other is that time step should be large enough so that the simulation don't last too much.

Sometimes it exist the possibility of choosing a non-constant time step, which will improve the efficiency of the simulation. It is useful when the atoms velocity varies during the simulation or in large time-scale simulations.

Finally it has to be said that a simulation is "valid" when the simulation time is longer than the relaxation time of the properties we are interested in.

3. Modeling and simulation

As said before, the aim of the project is to perform a copper nanowire simulation to find its mechanical properties. Lammmps software [3] was chosen to perform this simulation because it is a classical Molecular Dynamics code written in C++ that can be easily modified.

This software can model an ensemble of particles in a solid, liquid or gaseous state and also atomic, metallic and biological systems using different kinds of boundary conditions and force fields. Another interesting feature is that the program can handle a simulation of a system with millions or billions of particles, but at the same time it is also accurate with smaller systems.

Lammmps is a software which integrates Newton's equations of motion for a group of atoms that interact between themselves. To describe this interaction, some information about the interatomical potentials of different kind of materials is included in the program. For computational efficiency Lammmps uses neighbor lists to keep track of nearby particles.

3.1. Using Lammmps

Installation

Before installing Lammmps certain prerequisites must be fulfilled. This software uses fast Fourier transformation to solve the simulation, thus FFTW program needs to be installed. It is also recommended to install MPICH2 program that allows the option of parallel processing.

Starting with the FFTW2 program, Lammmps requires version 2, thus it has to be downloaded from its website. Once it is downloaded it has to be copied in a Cygwin directory called "home" so it can be installed through Cygwin. That directory mentioned above is the one where Lammmps will also be installed. The following commands must be written in the cygwin bash shell:

- `ls` (to make sure the program is in the correct directory)
- `gunzip fftw-2.1.5.tar.gz`
- `tar xvf fftw-2.1.5.tar`

Once those commands have been written, the program is unzipped and a new directory is created. After this you should enter to this directory and type the next commands:

- ./configure
- make
- make install

For the MPICH2 program the steps mentioned above for the FFTW2 program just have to be repeated. This installation takes more time to configure and to make than the one before.

After those programs have been installed you can proceed with the Lammmps set up. The C++ “source tarball” must be downloaded from Lammmps website and located in the same directory as before. Then, you need to type the first three commands from before and it will also turn into a folder. Once you are into this folder you should enter to the subfolder called “src” and after this also enter to the subfolder make. In this subfolder you will find the cygwin makefile and you will need to modify the next lines using the Notepad++ program:

- "-L/cygdrive/c/cygwin/mpich2-1.0.4p1/lib" to
 - "-L/cygdrive/c/cygwin/home/*NAME*/mpich2-1.2.1p1/lib"
- "-L/cygdrive/c/cygwin/fftw/lib" to
 - "-L/cygdrive/c/cygwin/usr/local/lib"
- "-DFFTW_FFTW -I/cygdrive/c/cygwin/fftw/include" to
 - "-DFFTW_FFTW -I/cygdrive/c/cygwin/usr/local/include"

Afterwards, you just have to come back to the “src” directory and type “make cygwin”. If all the steps are followed and no error messages appear, then the program is correctly installed. This means that a file called “lmp_cygwin.exe” has appeared in the “src” subfolder, which is the file needed to run a simulation.

Running Lammmps

There are two important things that Lammmps needs to run a Molecular Dynamic simulation, which are the input script and a potential file. The first one is where all the instructions that the program will execute are written. Otherwise, the potential file is the one that includes the parameters of the interatomical potential of the simulated system.

The structure of an input script in Lammmps is the one that follows:

1. Initialization
2. Atom and Lattice definition
3. Force fields
4. Settings
5. Run

The first step is to set the units that are going to be used, define if it is going to be 2D or a 3D simulation, and decide which kind of boundary conditions are going to be considered. The next step is to choose the lattice type (fcc, bcc, diamond, etc.) and its dimensions. Once it is done, a simulation box is defined and the atoms are created inside it. Later, the force fields that will describe the interaction between particles are chosen. This interaction can be described using potential files or, on the other hand, changing some parameters in a formula, as it happens in the Lennard-Jones potential.

After those previous system definitions, the simulation settings are defined. This means that it is time to think about what you want to represent in the simulation and which things you want to know about it, for example, pressure, temperature, potential energy, etc. The last thing to choose is the time step and start running the simulation. To do this last part you have to write the next order in the Cygwin Bash Shell:

```
“ ./lmp_cygwin <name_of_the_file ”
```

3.2. Definition of the nanowire studied (design constraints)

As the focus of this paper is to perform a copper nanowire simulation and study its Young's Moduli, an FCC lattice is going to be considered. This means that the lattice constant is $3,615\text{\AA}$, hence, the sides of the nanowire are not going to be exact measures. In Lammmps program an FCC lattice is performed as a cube with 4 basis atoms, one at the corner and 2 at the cube face centers.

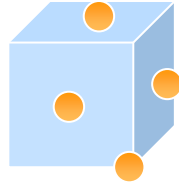


Figure 1: FCC lattice in Lammmps program

The nanowire is going to be performed choosing a simulation box and applying periodic boundary conditions in one direction. In the other two directions a shrink-wrapped boundary is going to be set. This means that the simulation box will deform following the atoms motion. The x, y, z coordinate axes represent the lattice direction of [100], [010] and [001] respectively. This simulation box pretends to have 40 atoms in x direction and 40 more in y direction, while in z it will be only 12 atoms. The final dimensions of the nanowire can be seen in the following picture:

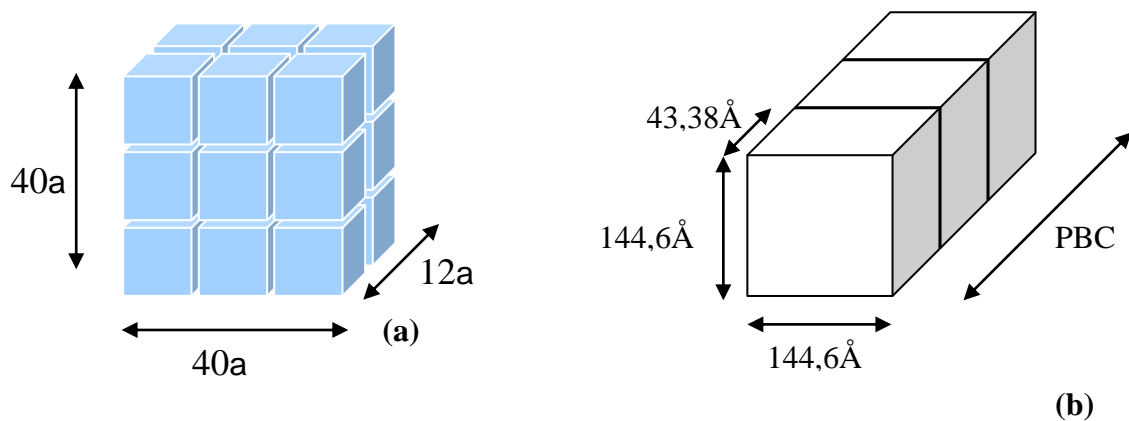


Figure 2: (a) Simulation box with dimensions $40a \times 40a \times 12a$. (b) Nanowire studied.

Another thing that needs to be decided before simulation starts, is the potential that is going to be used. In this case, the interatomical potential that matches better is the EAM potential as described in the above sections. Once the simulation with the EAM has been performed, it is also going to be considered the LJ potential, as it is the most known, and this way we will be able to compare the results using both.

The simulation involves applying a displacement in the y direction and extract the Young's Modulus from the stress-strain plot. Unfortunately, there is a drawback in applying a displacement in Lammmps program, because this instruction (fix deform) only works in a direction where periodic boundary conditions are applied. To solve this problem, a velocity is going to be set in the 4 upper layers by using the instruction "fix velocity" while the four lower layers are locked.

The stress-strain plot is going to be performed by computing the stress per atom that is in pressure*volume units. Once it is stored, the stress per atom is going to be summed for all the atoms in the simulation box in the desired direction and at the end divided by the volume of the whole system.

To initialize the simulation, a relaxation state is defined and the atoms are left freely with a random velocity using NVT. The temperature is going to be set around 0K during all the simulation and the volume and number of atoms too. Different simulations with increasing strain rates will be performed to confirm that the Young Modulus does not depend on this variable.

Once the velocity is set in the upper layers and the lower layers are locked, the simulation will be executed using NVT which performs time integration on non-Hamiltonian equations of motion, thus, generates positions and velocities sampled from canonical ensembles. On the middle atoms the position and velocity will be updated using NVE integration which keeps the volume and the energy of the system constant. This creates a system trajectory consistent with the microcanonical ensemble.

3.3. Input Script

In this part the input script used to perform the simulation is described with all the details needed to understand how it works.

Initialization

Definition of the units that are going to be used during the simulation. The “metal” units are: distance = Å, time = picoseconds, energy = eV, velocity = Å/ps, temperature = K, pressure = bar.

units metal

Definition of the simulation dimensions, in this case 3D.

dimension 3

Boundary conditions in each direction. For x and y directions the boundary is type s, thus the position of the face is set so as to encompass the atoms in that dimension, it is shrink-wrapped. In z direction the boundary is periodic, which means that particles interact across the boundary.

boundary s s p

Atom and Lattice definition

Definition atoms attributes. The atomic type is the one that matches better with metals simulation.

atom_style atomic

Storage of the lattice constant.

variable latconst equal 3.615

Definition of the type of lattice

lattice fcc \${latconst}

Definition the region that is going to be studied. It is a block with dimensions $40a \times 40a \times 12a$, this means that there are 40 lattices in x direction, 40 lattices in y direction and 12 lattices in z direction. The last command of this group is what generates the atoms the right position in the lattice.

```
Region nanowire block 0 40 0 40 0 12
create_box 3 nanowire
lattice fcc  $\{\text{latconst}\}$  orient x 1 0 0 orient y 0 1 0 orient z 0 0 1
create_atoms 1 region nanowire
```

Force fields

Choice of the interatomical potential that is going to be used and the directory to find the parameters. The “*” indicates that the potential is going to be applied between all the types of atoms defined.

```
pair_style eam
pair_coeff * * Cu_u3.eam
```

Settings

Definition of the upper and lower region, where the velocity is going to be set.

```
variable L equal 144.6
variable dy equal 14.46
variable y1 equal -0.0
variable y2 equal  $\{\text{dy}\}$ 
region Lower block INF INF  $\{\text{y1}\}$   $\{\text{y2}\}$  INF INF units box
group Lower region Lower
variable y1 equal  $\{\text{L}\} - \{\text{dy}\}$ 
variable y2 equal  $\{\text{L}\}$ 
```



```

region Upper block INF INF ${y1} ${y2} INF INF units box
group Upper region Upper
group boundary union Lower Upper
group middle_atoms subtract all boundary

```

Give some initial velocity to all the atoms randomly at a temperature of 0.01K.

```

velocity all create 0.01 511124 rot yes mom yes

```

Compute the stress for each atom and sum each component for all atoms.

```

compute peratom all stress/atom pair virial
compute fy all reduce sum &
  c_peratom[1] c_peratom[2] c_peratom[3] &
  c_peratom[4] c_peratom[5] c_peratom[6]
compute p all reduce sum c_peratom[1] c_peratom[2] c_peratom[3]

```

Create a variable called sigmavolume that stores the y component of the stress tensor.

```

variable sigmavolume equal c_fy[2]/vol

```

Create a variable called strain, which calculates the strain during the simulation

```

variable strain equal (ly-v_L)/v_L

```

Create a variable called press that computes the total pressure.

```

variable press equal -(c_p[1]+c_p[2]+c_p[3])/(3*vol)

```

Definition of the frequency with which the output information is going to be registered and which variables are going to be recorded.

```

thermo 10000
thermo_style custom step ly vol v_strain v_sigmavolume temp etotal press v_press

```

Run the simulation

Run the simulation a few timesteps to reach the relaxation state using NVT.

```
fix      relax all nvt temp 0.01 0.01 .01
timestep 0.005
run      500
unfix    relax
```

Set force to zero in all directions, applying some velocity in the upper layers and zero velocity in the lower layers.

```
variable upper_vel equal 0.01
fix      zeroing_force_on_lower Lower setforce 0.0 0.0 0.0
fix      zeroing_force_on_upper Upper setforce 0.0 0.0 0.0
velocity Lower set 0.0 0.0 0.0 units box
velocity Upper set 0.0 ${upper_vel} 0.0 units box
```

Run the simulation during more time. The motion of the middle_atoms is integrated using NVT while the boundary uses NVE.

```
fix      fix1 middle_atoms nvt temp 0.01 0.01 0.01
fix      fix2 boundary nve
run      300000
```

Set again the velocity to zero and run some steps more.

```
velocity Upper set 0 0 0
run      1000
```

SIMULATION DONE

```
print "All done"
```

4. Results and discussion

To perform the first simulations, increasing strain rates are going to be used so we are able to see differences in the material behavior. According to previous works [6] the strain rates for strain simulations are around 10^8s^{-1} and 10^9s^{-1} . In this case, the simulations performed were from 10^7s^{-1} to 10^8s^{-1} , due to the larger simulation time. To reduce the computational expense and the simulation time, the time step that was chosen was 5fs. Usually the time step used for this kind of tensile simulations is around 1fs, but in previous works [7] where they have studied the time step effects in a simulation they conclude that under 5fs the simulation results are still trustable. As said before, all the simulations are going to be carried out under absolute zero temperature (0K).

To check the correct value of the strain rates found in other works some simulations with higher strain rates were performed. In these cases it is observed an unexpected behavior of the copper nanowire. There is no similarity observable between the two plots due to a non-homogeneous behavior of the material at that strain rates. The stress-strain curve shows two peaks instead of one and there is no possibility to find the right Young's Modulus because of this non-homogeneous behavior.

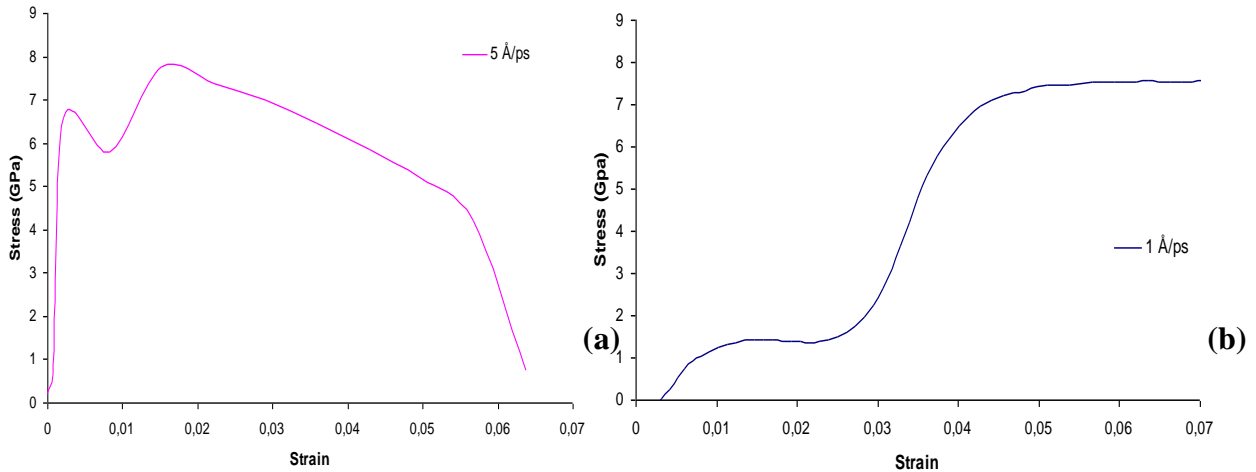


Figure 3: Stress-strain curves for Cu nanowires of $40\text{a} \times 40\text{a} \times 12\text{a}$.
(a) Strain rate of 5 \AA/ps . (b) Strain rate of 1 \AA/ps .

In order to validate the time step chosen some simulations with increasing time steps have been performed.

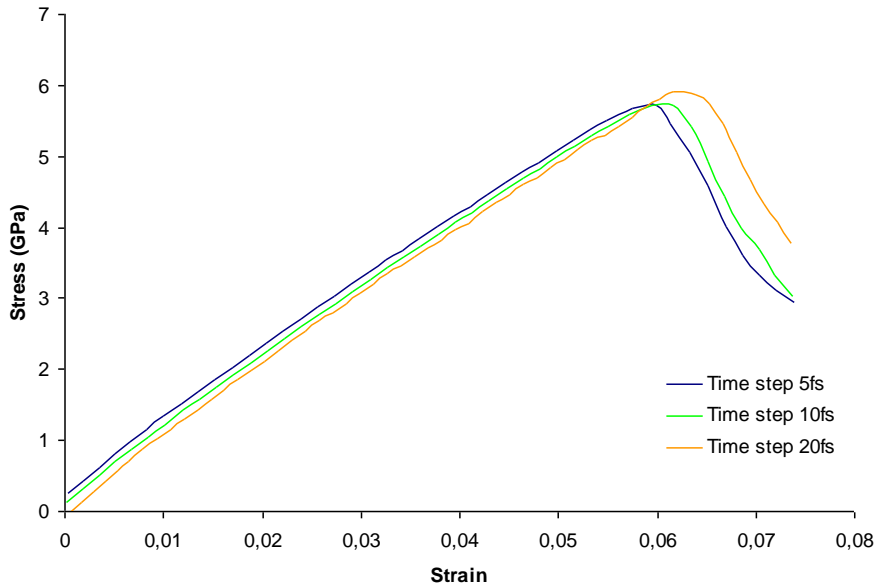


Figure 4: Stress-strain curves for Cu nanowires of 40a x 40a x 12a at different time steps.

Figure 4 shows the stress-strain curves for time steps ranging from 5fs to 20fs. As expected the Young's Modulus does not depend on the time step chosen. On the other hand, the Yield strength seems to change a little bit with different time steps. It should be remembered that smaller time steps mean that the results are more accurate but the simulation may last longer.

There is a possibility that with larger time steps the particle-particle interaction could not be calculated frequently enough, and then may appear two or more particles occupying the same position at the same time. This could cause a big repulsion between them and consequently the particles speed increases considerably. To avoid this problem the time step should be smaller than the collision time that is around 10fs [7].

The first simulations are performed using the EAM potential, which is supposed to be the one which describes better the copper behavior as it is used for metal materials. As shown in the following picture, the stress-strain curves have different behaviors after reaching the Yield strength for different strain rates. In fact, the Yield strength, which is the maximum stress, also changes depending on the strain rate. The lowest strain rate reduces the stress sharply after the peak is reached while the highest strain rate shows a smoother slope.

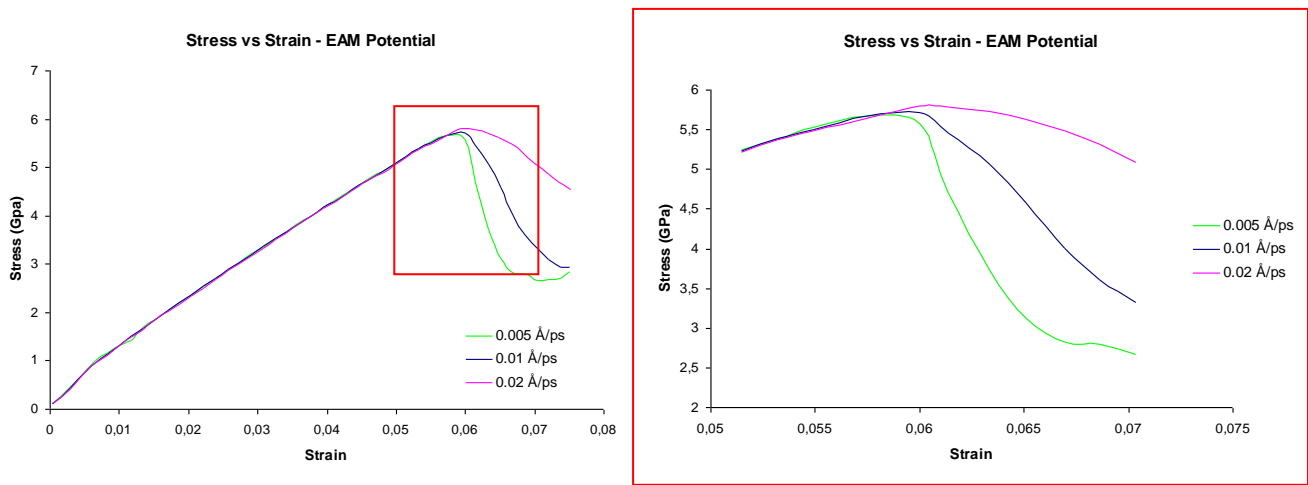


Figure 5: Stress-strain curves for Cu nanowires and EAM potential at different strain rates and zoom.

It is also observed that during the elastic deformation, the curves coincide, so the Young's Modulus does not change with different strain rates. From here we can extract that the displacement or the strain rate has no influence in the Young's Modulus.

The next step is to focus on the plots individually and extract the Young's Modulus by adding a trend line to see the value, as it follows the next equation:

$$E = \frac{\sigma}{\varepsilon} \quad (13)$$

In Table 1, it seems to appear a possible dependence between the Young's Modulus and the strain rate, but in fact the difference between those values is not so big and may be due to the values extracted from the trend line. In some cases if we take a shorter range of values for the trend line, the Young's Modulus also varies a little bit. In this case the range of values chosen it is the same for all three strain rates.

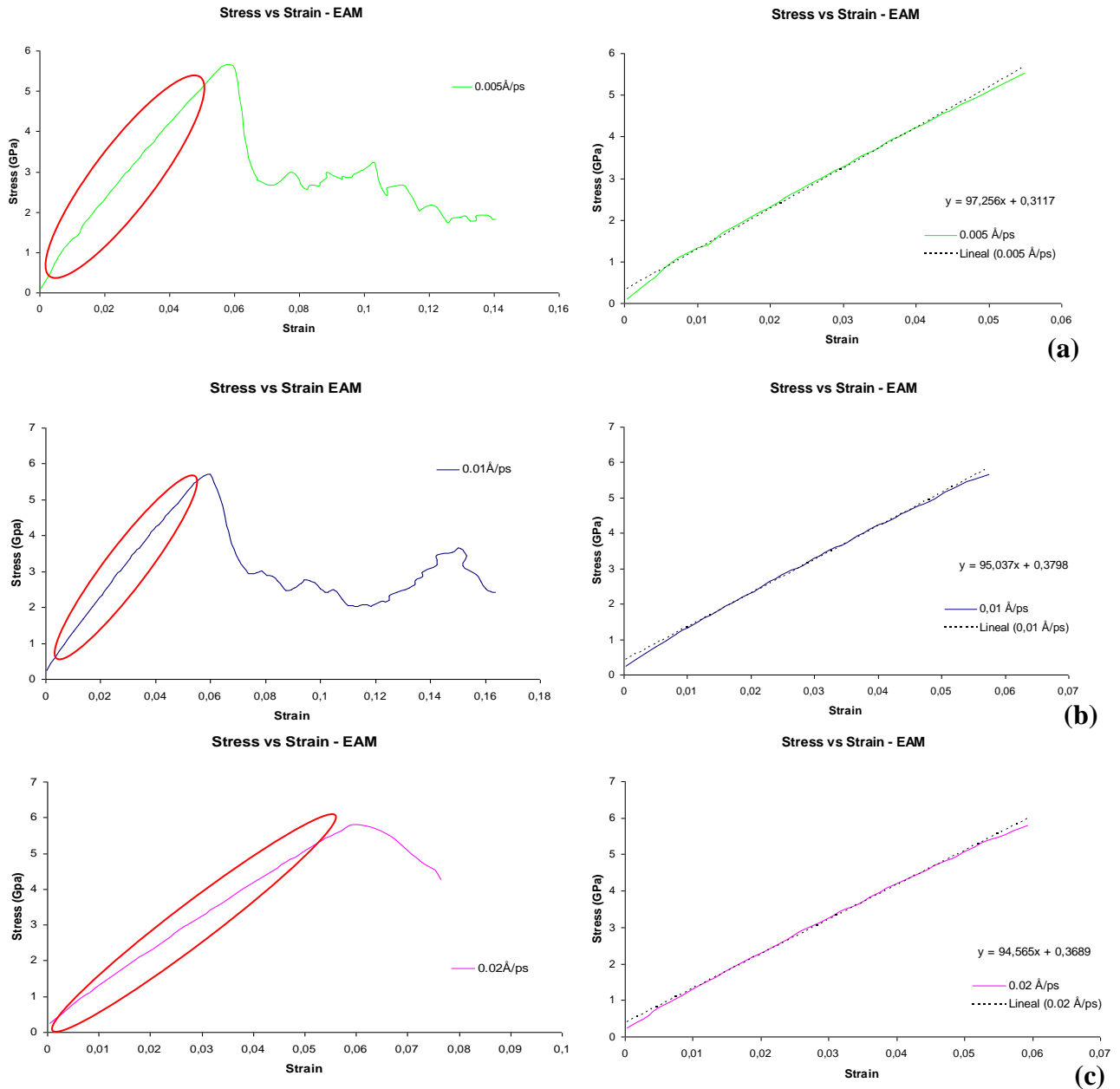


Figure 6: Stress-strain curves for Cu nanowires of 40a x 40a x 12a and EAM potential at different strain rates.

(a) Strain rate of 0.005 Å/ps. (b) Strain rate of 0.01 Å/ps. (c) Strain rate of 0.02 Å/ps.

Once the Young's Modulus have been extracted from the plots it is time to compare them with the theoretical one, which from previous researches is around 125GPa in that direction.

Strain rate (Å/ps)	Young's Modulus (Gpa)	Error (Theoretical = 125GPa)
0.005	97.256	0.2522
0.01	95.037	0.2724
0.02	94.656	0.2758

Table 1: Effects of the strain rate in Young's Modulus for EAM potential and error compared with theoretical value $E = 125\text{GPa}$

Comparing those values it can be said that they differ a bit from the theoretical one, but between themselves they are similar as it was seen in the previous plots. From the values in the table it is easy to say that Young's Modulus for copper nanowires is lower than the theoretical value, as other researchers reported for other materials[8], but more simulations are needed to reach that conclusion.

For the Lennard-Jones potential two simulations with the strain rates 0.01Å/ps and 0.02Å/ps have been carried out. In this case the simulations last longer due to the difference in how the potential is evaluated. For the EAM potential the forces are calculated by searching the value of the embedding function F in a list included in the program. On the other hand, the Lennard-Jones calculation needs to evaluate the formula of the potential each time step. In figure 7 it is possible to see that, as happened with the EAM potential, the Young's Modulus at different strain rates coincide.

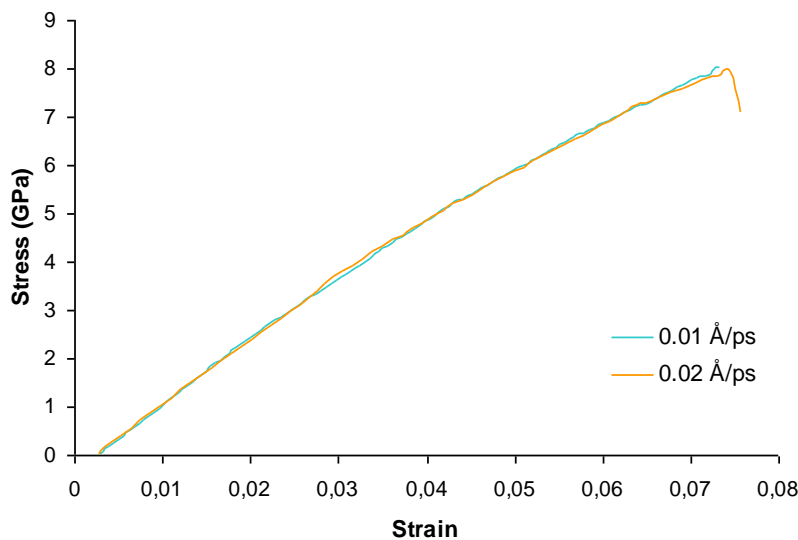


Figure 7: Stress-strain curves for Cu nanowires of $40a \times 40a \times 12a$ and Lennard – Jones potential at different strain rates.

Focusing on each plot individually and extracting the Young's Modulus for the Lennard-Jones potential simulations it is possible to see values more near to the theoretical ones.

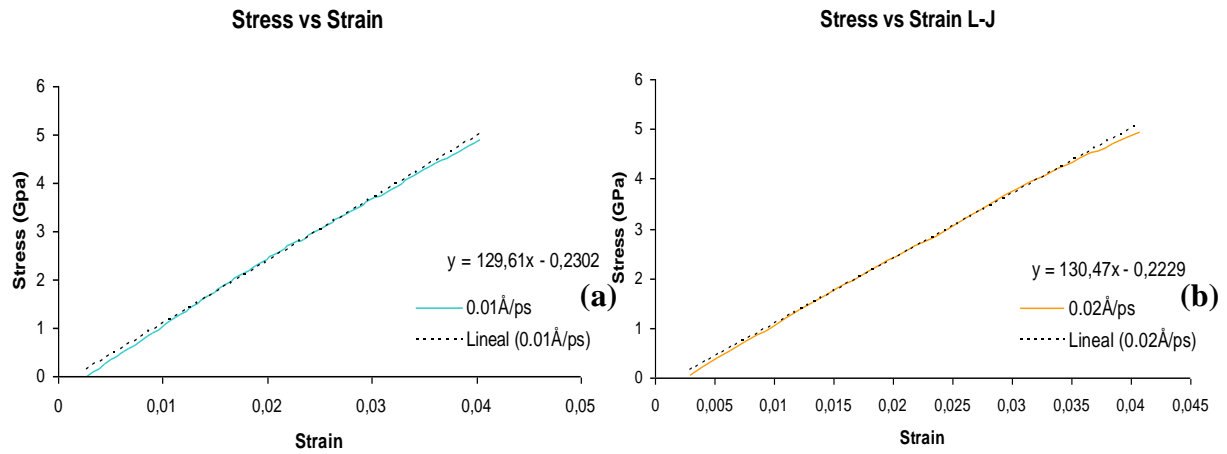


Figure 8: Stress-strain curves for Cu nanowires of 40a x 40a x 12a and Lennard – Jones potential at different strain rates. (a) Strain rate of 0.01 Å/ps. (b) Strain rate of 0.02 Å/ps.

Strain rate (Å/ps)	Young's Modulus (Gpa)	Error (Theoretical = 125GPa)
0.01	129.61	0.03688
0.02	130.47	0.04376

Table 2: Effects of the strain rate in Young's Modulus for Lennard-Jones potential and error compared with theoretical value $E = 125\text{GPa}$

To decide which potential should be used for further studies a plot with both potential functions and a strain rate of 0.01 Å/ps has been performed.

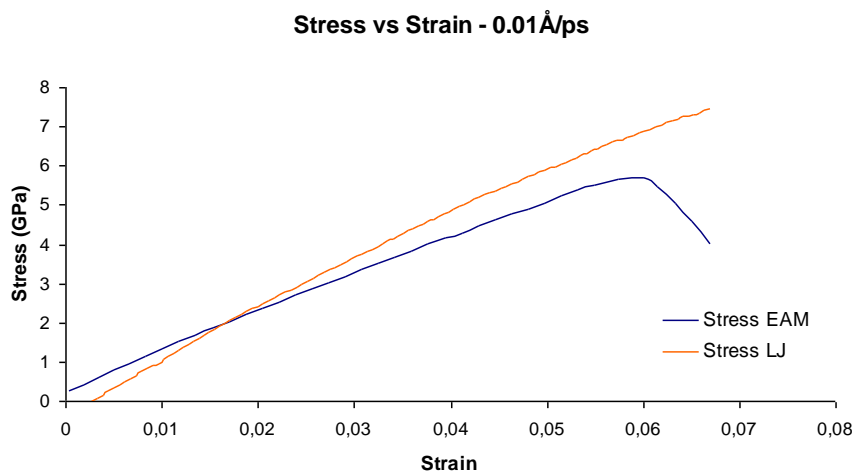


Figure 9: Stress-strain curves for Cu nanowires of 40a x 40a x 12a at 0.01 Å/ps for both potentials

According to the above tables comparing the theoretical Young's Modulus with the one extracted from the simulations, the Lennard-Jones potential has less error, hence, is the one that fits better. There is the possibility that Young's Modulus increases with increasing size and then the EAM potential will show a closer value.

On the other hand, it has to be remembered that the EAM is supposed to describe better metallic systems behavior. In fact, probably the EAM potential represents better the elastic and the plastic zone in a tensile test than the Lennard-Jones potential.

The next simulations performed are going to show if there is any dependence between the size of the nanowire and the Young's Modulus.

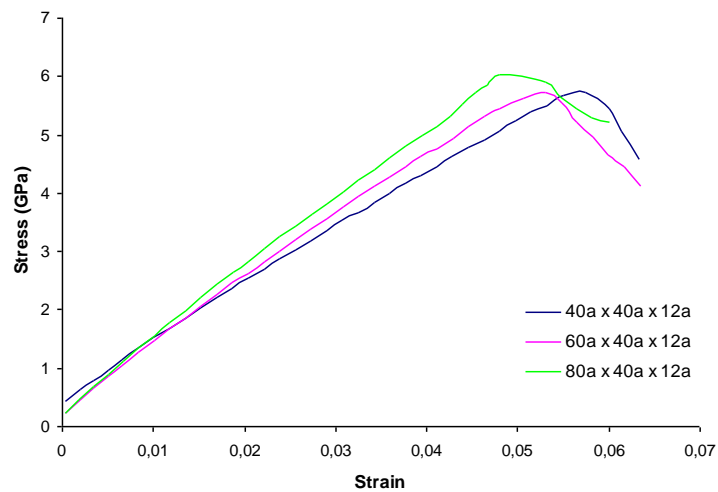
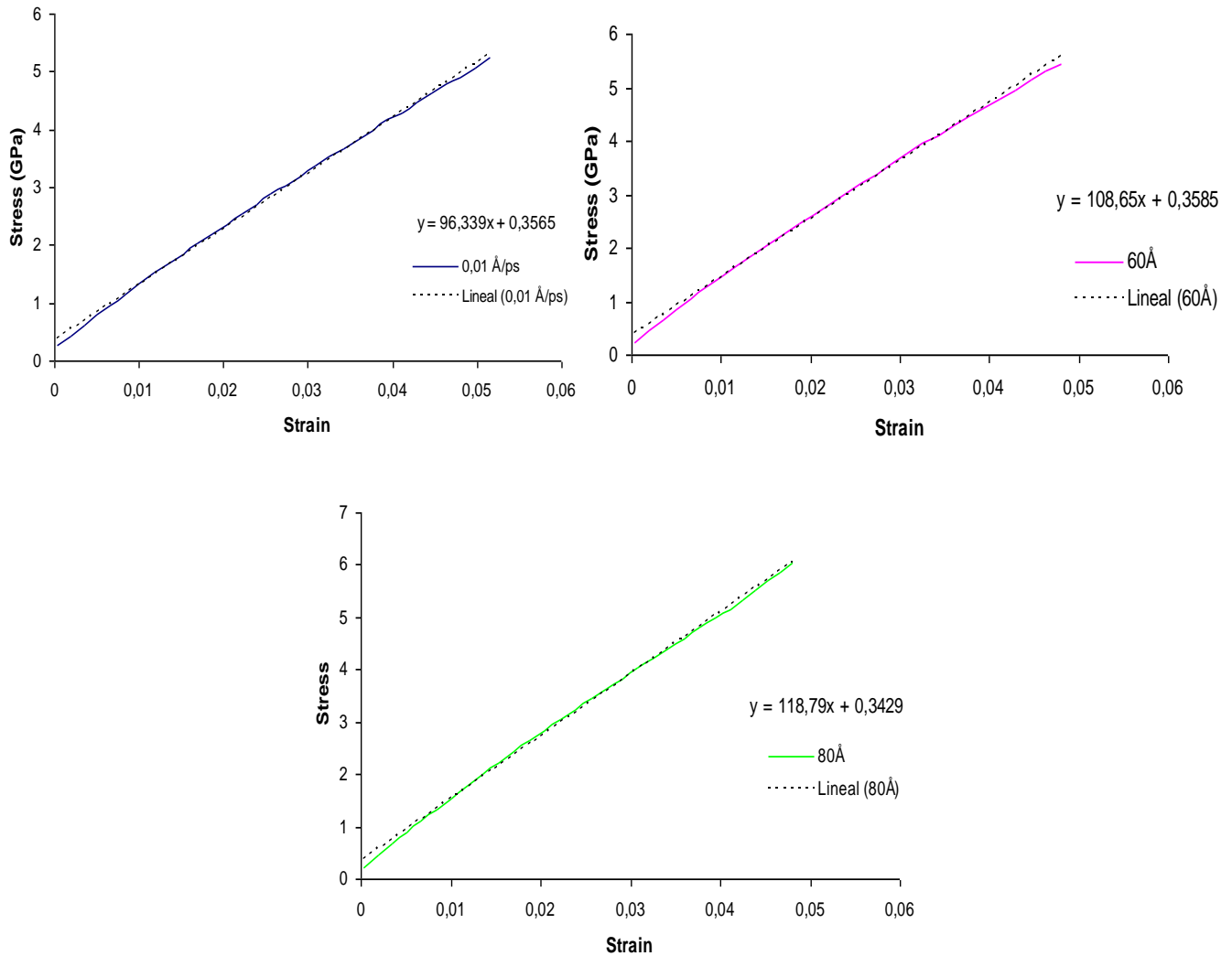


Figure 10: Stress-strain curves for Cu nanowires with different dimensions at 0.01 Å/ps for EAM potential



As it can be seen in figure 10, the Young's Modulus do not coincide for different dimensions in direction x for the nanowire. On the following plots, Young's Modulus are extracted from each case.

Dimensions	Young's Modulus (Gpa)	Error (Theoretical = 125GPa)
40a x 40a x 12a	96,339	0,2293
60a x 40a x 12a	108,65	0,1308
80a x 40a x 12a	118,79	0,0497

(a)

(b)

Table 3: Effects of the x dimension in Young's Modulus for EAM potential and error compared with theoretical value $E = 125\text{GPa}$

From table 3 is it possible to assure that Young's Modulus has a clear dependence on the x dimension of the nanowire, as it increases with growing width. The range of values for the different dimensions does not deviate so much from the theoretical value. Those results were expected as we are getting closer to the bulk properties. This increasing Young's Modulus may come from the inhomogeneity of the cross-section of the nanowire, which is a result from the surface effects, as other previous studies predicted [9, 10].

5. Conclusions and future considerations

Molecular Dynamics simulations have been carried out on a copper nanowire to extract its Young's Modulus. First of all some simulations with increasing time step have been performed to assure that there was no dependence of that parameter on Young's Modulus. Once it has been checked, some other simulations with different strain rates have been performed and the conclusion reached is that the strain rate chosen has no dependence on Young's Modulus. Later simulations have been carried out with a strain rate of 0.01 \AA/ps and a time step of 5fs.

On the other hand, there seems to be dependence between strain rate and the behavior after the Yield strength because there are no dislocations created for those strain rates. For higher strain rates the behavior in the non-elastic zone appears to be smoother than for the lower ones. As reported by others researchers[11,12], for strain rates lower than 0.005 \AA/ps the dislocations appear, which hardens the material and a zigzag curve appears after the Yield strength. The first sharp drop happens because of the occurrence of dislocation while the following ones are created by the dislocation movements.

Comparing the EAM potential with the Lennard-Jones potential, the behavior of the copper nanowire is better described with the first one, as it is the one used for metallic systems. In the EAM potential simulations the transition between before and after the Yield strength can be easily seen, while with the Lennard-Jones potential it does not appear, since the simulation stops before it happens.

Once the comparisons between the different x dimensions are done, it can be concluded that the Young's Modulus found with EAM potential does not divert much from the theoretical value. However it seems to appear a clear relationship between the growing x dimension and the Young's Modulus. This fact can be due to the surface effects as many studies showed previously, for example the ones made by Guo and Zhao[13].

This study has been useful to observe some mechanical properties of copper at nano-level and it could be important for future research and application in nanotechnology. Some future papers about this subject could be focused on the dependence of the temperature on the Young Modulus or the study of the behavior of the material with different kinds of surface defects.

Another possibility is to get started with multi-scale simulation methods which aim to be a bridge between atomistic simulations and continuum modeling. The continuum simulation is used in all the regions of a metallic material and considers a homogeneous and smooth deformation. On the other hand, as it was explained before, it only captures the individual atomistic dynamics of a small subregion. If we are able to extrapolate from atomic level to macro-scale, we will understand other phenomena as dislocation patterns in fatigue among others.

6. References

- [1] Craighead H. G., *Nanoelectromechanical systems*. Science **290**(2000) p1532-1535.
- [2] Plimpton S J., *Computational Physics* 117 (1995)
- [3] Lammmps 2011, <http://lammmps.sandia.gov>.
- [4] Mishin, Y., et al., *Structural stability and lattice defects in copper: Ab initio, tight-binding, and embedded atom calculations*. Physical Review B, **63**(2001) p224106.
- [5] Foiles, S., M. Baskes, and M. Daw, *Embedded-atom-method functions for the fcc metals Cu, Ag, Au, Ni, Pd, Pt, and their alloys*. Physical Review B, **33** (1986) p7983-7991
- [6] C. H. Ersland, C. Thaulow, D. Farkas, E. Ostby. *Atomistic studies and comparison of α -Fe potentials in mode I fracture*.
- [7] H.F. Zhan, Y.T. Gu, Prasad Yarlagadda, *Atomistic numerical investigation of single-crystal copper nanowire with surface defects*. (2010)
- [8] Y. Tong et al., *Molecular dynamics study of metal properties of bismuth telluride nanofilm*. Physica B **405** (2010) p3193.
- [9] Hasmy, A. and E. Medina, *Thickness induced structural transition in suspended fcc metal nanofilms*. Physical Review Letters, **88**(2002): p96103.
- [10] H. A. Wu, *Molecular dynamics study of metal nanowire*, *Mechanics Research Communications* **33** (2006) p12-16.
- [11] Ikeda, H., et al., *Strain Rate Induced Amorphization in Metallic Nanowires*. *Physical Review Letters*, **82** (1999). p 2900.
- [12] H. A. Wu, *European Journal of Mechanics A/Solids* **25** (2006) p374-376.
- [13] J. G. Guo and Y. P. Zhao, *Surface Review Letters*. **14** (2007). p667.
- [14] Kang, J. W., Hwang, H.J., *Mechanical deformation study of copper nanowire using atomistic simulation*. Nanotechnology **12**(2001) p295-300.
- [15] Ercolessi, F. *A molecular dynamics primer*, <http://www.fisica.uniud.it/~ercolessi/>
- [16] Markus J. Buehler *From nano to macro: introduction to atomistic modeling techniques*. Lecture series, CEE (2006)
- [17] W.K.Liu, E.G. Karpov, S. Zhang, H.S. Park, *An introduction to computational nanomechanics and materials*. Comput. Methods Appl. Mech. Engrg. **193**(2004) 1529-1578