JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Numerical algorithm for the prediction of pressure induce phonon instabilities via computational modeling of Acoustic wave propagation in isotropic and anisotropic crystals.

[1]Oscar Guerrero
[1]Retired
[1]2609 Lake Victoria Dr, El Paso, TX, 79936, USA

Abstract: In this writing, I will present a fast-efficient computational Pseudo code to find acoustic phonon instabilities in crystals. The code uses the stress-strain relations to numerical find sound velocities in terms of second-order elastic constants for various wave modes in uniaxial and hydrostatically compressed crystals for a given wave direction $< \boldsymbol{k} >$. I use Tantalum and Copper as the testing materials. The results are compared with Molecular dynamics simulations using the EAM potential formalism, DFT calculations via VASP and lattice stability function uniaxial compression by determining the phonon-dispersion relation over the entire BZ. The implementation of the code is discussed using the free, open-source software OCTAVE and the symbolic program MATHEMATICA.

Index Terms **– Phonon instability, speed sound, Molecular Dynamics, Single Crystal Metals**

I. CALCULATION OF THE VELOCITY OF SOUND ON CRYSTALLINE SYSTEM

The Christoffel equation that governs the propagation of elastic waves is

$$\rho v^2 w_i = \sum_{jkl} \left(C_{ijkl}\right) \hat{k}_j \hat{k}_l \, w_k \tag{1.1}$$

Equation (1.1) represents the eigenvalue problem, where the phase velocities ρv^2 are the eigenvalues, and polarization of the wave are the eigenvectors.

The calculation of the velocity of sound based on the elastic constants in equation (1.1) is only valid for the case of zero stress ($P_{ij} = 0$). To generalize to the non-zero stress case, one can formulate the stability conditions in terms of the elastic stiffness coefficients, and the elastic coefficients B_{ikjl} must therefore replace C_{ikjl}:

$$\rho v^2 w_i = \sum_{jkl} \left(\sigma^o_{jl}\delta_{ik} + B_{ijkl}\right) \hat{k}_j \hat{k}_l \, w_k \tag{1.2}$$

The term between the parenthesis in equation (1.2) I define as the dispersion matrix $L_{ik} = \sigma^o_{jl}\delta_{ik} + B_{ijkl}$

The reader can seek the book titled Thermodynamics of Crystals written by D. Wallace [3] for a better understanding of the derivation of equations (1.1) and (1.2)

Observe that equation (1.2) has three solutions, denoted by s = 1, 2, 3. Thus for a given direction, k in a solid, three-small amplitude plane waves with perpendicular polarizations can propagate. Waves with polarization vectors parallel or perpendicular to the propagation direction $< k >$ are defined as longitudinal or transverse (shear) waves, respectively. However, in general, a wave is neither purely longitudinal nor transverse.

Two directions are of common interest in wave propagation experiments: (1) pure mode directions, which have one pure longitudinal wave and two transverse purse waves; and (2) quasi-pure mode directions, which have one pure transverse wave and two mixed waves. The symmetry directions of a crystal are always pure mode directions. In the following sections, I will provide step-by-step examples of how to find the solution to equation (1.2) for the wave directions K <0,0,1> and K <1,1,0> for the Cubic symmetry crystals, and K <1,0,1>

for Orthorhombic symmetry; this is to help the reader better understand the solving process. After that, I can move confidently into explaining the general symbolic and numerical solution to equation (1.2)

1.1 Example: How to calculate the solution to the propagation of sound for Cubic crystals for the directions K<1,0,0> and K<1,1,0> at zero pressure

I first choose to solve the problem of calculating the elastic wave propagation for a cubic crystal symmetry. It will set the foundation to help me describe the general scenario of an arbitrary propagation direction $< k >$ given any so-called crystal symmetry. It is also a well-known problem that has already been solved in the book of Kittel, Charles - Introduction to Solid State Physics 8Th Edition page 80.

The theory of elasticity gives the basis for describing elastic wave propagation. Hooke's law describes the relationship between stress and strain of a flexible material. I can represent this relation by the tensorial equation:

$$\begin{pmatrix}\sigma_1\\ \sigma_2\\ \sigma_3\\ \sigma_4\\ \sigma_5\\ \sigma_6\end{pmatrix} = \begin{pmatrix}C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16}\\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26}\\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36}\\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46}\\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56}\\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66}\end{pmatrix}\begin{pmatrix}\varepsilon_1\\ \varepsilon_2\\ \varepsilon_3\\ \varepsilon_4\\ \varepsilon_5\\ \varepsilon_6\end{pmatrix} \tag{1.3}$$

Table 1.1: Scheme of the independent second-order elastic constants for each group only the subscripts $\alpha\beta$ for each $C_{\alpha\beta}$ are listed. The scheme is the same for two hexagonal groups (denoted as H) and two cubic groups (denoted as C), N stands for None, M for Monoclinic, O for orthorhombic, R for Rhombohedral, T for Tetragonal and I for Triclinic

N	M	O	TII	TI	RII	RI	C	I	H
11	11	11	11	11	11	11	11	11	11
12	12	12	12	12	12	12	12	12	12
13	13	13	13	13	13	13	12	12	13
14	0	0	0	0	14	14	0	0	0
15	15	0	0	0	15	0	0	0	0
16	0	0	16	0	0	0	0	0	0
22	22	22	11	11	11	11	11	11	11
23	23	23	13	13	13	13	12	12	13
24	0	0	0	0	-14	-14	0	0	0
25	25	0	0	0	-15	0	0	0	0
26	0	0	-16	0	0	0	0	0	0
33	33	33	33	33	33	33	11	11	33
34	0	0	0	0	0	0	0	0	0
35	35	0	0	0	0	0	0	0	0
36	0	0	0	0	0	0	0	0	0
44	44	44	44	44	44	44	44	44	44
45	0	0	0	0	0	0	0	0	0
46	46	0	0	0	-15	0	0	0	0
55	55	55	44	44	44	44	44	44	44
56	0	0	0	0	14	14	0	0	0
66	66	66	66	66	66	66	44	44	66

Therefore, the value of the elastic constants C_{ij} is required, and there are different methods, both experiment and computational, to calculate its value. The interested reader is referred to the following referemces [4-8].

Its well-known crystals have their rotational symmetry (point group). These rotational symmetries enforce extra constrain on the elastic constants, further reducing the number of independent elastic constants (look at Table 1.1).

For cubic symmetry, there are only three required:

$$\begin{pmatrix}\sigma_1\\ \sigma_2\\ \sigma_3\\ \sigma_4\\ \sigma_5\\ \sigma_6\end{pmatrix} = \begin{pmatrix}C_{11} & C_{12} & C_{12} & 0 & 0 & 0\\ C_{12} & C_{11} & C_{12} & 0 & 0 & 0\\ C_{12} & C_{12} & C_{11} & 0 & 0 & 0\\ 0 & 0 & 0 & C_{44} & 0 & 0\\ 0 & 0 & 0 & 0 & C_{44} & 0\\ 0 & 0 & 0 & 0 & 0 & C_{44}\end{pmatrix}\begin{pmatrix}\varepsilon_1\\ \varepsilon_2\\ \varepsilon_3\\ \varepsilon_4\\ \varepsilon_5\\ \varepsilon_6\end{pmatrix} \tag{1.4}$$

A personal thought I would like to communicate to the reader of this writing is the following: The problem of finding the speed of sound using the stress-strain relations in a crystal can also be interchangeable with the inverse problem of finding the elastic constants if the speed of sound is given. For example, many scientists in The Geophysics field use ultrasonic waves in specific directions to experimentally calculate the values of the elastic constants on the medium [9]. In any case, let me go back to demonstrate how to find the solution for K <1,0,0> and K <1,0,1>

1.1.1 Solution for K <1,0,0>

Having $\hat{k} = (1,0,0)$ the dispersion matrix L_{ik} is given by

$$L_{ik} = \begin{pmatrix}C_{1111} - P_{11} & C_{1121} & C_{1131}\\ C_{2111} & C_{2121} - P_{11} & C_{2131}\\ C_{3111} & C_{3121} & C_{3131} - P_{11}\end{pmatrix} \tag{1.5}$$

Introducing the Voigt notation, the above equation can be reduced as follows

$$L_{ik} = \begin{pmatrix}C_{11} - P_{11} & C_{16} & C_{15}\\ C_{61} & C_{66} - P_{11} & C_{65}\\ C_{51} & C_{56} & C_{55} - P_{11}\end{pmatrix} \tag{1.6}$$

Now introducing Cubic symmetry (see eq 1.4), we can neglect elastic coefficients which will be zero, therefore the problem reduces even more:

$$L_{ik} = \begin{pmatrix}C_{11} - P_{11} & 0 & 0\\ 0 & C_{44} - P_{11} & 0\\ 0 & 0 & C_{44} - P_{11}\end{pmatrix} \tag{1.7}$$

the eigenvalues L_{ik} defined by $\lambda_s = \rho v_s^2$ can easily be calculated

$$\lambda_1 = C_{11} - P_{11} \quad \lambda_2 = C_{44} - P_{11} \quad \lambda_3 = C_{44} - P_{11} \tag{1.8}$$

The eigenvectors indicate the polarization of the sound wave and, for the above case, are (1,0,0), (0,1,0), and (0,0,1), respectively. One may think the above elastic constants wave velocity relations are straightforward, but it's not because P_{11} (Pxx) and the coefficients C_{ij} are strain/stress dependent. In other words, the sound wave velocity changes as a function of the strain/pressure; this gives a hint there are specific crystallographic directions in a crystal where the sound wave velocity may tend to zero. it's possible to predict at what strain and pressure the dynamical instability caused by the softening of the acoustic phonon branches evaluated at $dw/dq|_0$ occurs.

For the scenario of a unstress lattice the equation (1.8) can be simplify to

$$\lambda_1 = C_{11} \quad \lambda_2 = C_{44} \quad \lambda_3 = C_{44} \tag{1.9}$$

1.1.2 Solution for K <1,1,0>

Having $\hat{k} = \frac{1}{\sqrt{2}}(1,1,0)$ the dispersion matrix L_{ik} is given by

$$L_{ik}=\frac{1}{2}\begin{pmatrix} C_{1111}-P_{11}+\dots & C_{1121}+C_{1122}+\dots & C_{1131}+C_{1132}+\dots \\ C_{1112}-P_{12}+\dots & C_{1221}+C_{1222} & C_{1231}+C_{1232} \\ C_{1211}-P_{21}+\dots & & \\ C_{1212}-P_{22} & C_{2121}-P_{11}+\dots & \\ & C_{2122}-P_{12}+\dots & \\ C_{2111}+C_{2112}+\dots & C_{2221}-P_{21}+\dots & C_{2131}+C_{2132}+\dots \\ C_{2211}+C_{2212} & C_{2222}-P_{22} & C_{2231}+C_{2232} \\ & & C_{3131}-P_{11}+\dots \\ C_{3111}+C_{3112}+\dots & C_{3121}+C_{3122}+\dots & C_{3132}-P_{12}+\dots \\ C_{3211}+C_{3212} & C_{3221}+C_{3222} & C_{3231}-P_{21}+\dots \\ & & C_{3232}-P_{22} \end{pmatrix} \tag{1.10}$$

Now introducing Cubic symmetry (see eq 1.4), the Voigt notation and for a unstress crystal Pij = 0, we can neglect elastic coefficients which will be zero, therefore the problem reduces even more:

$$L_{ik}=\frac{1}{2}\begin{pmatrix} C_{11}+C_{44} & C_{12}+C_{44} & 0 \\ C_{12}+C_{44} & C_{11}+C_{44} & 0 \\ 0 & 0 & 2C_{44} \end{pmatrix} \tag{1.11}$$

the eigenvalues L_{ik} defined by $\lambda_s=\rho v_s^2$ can easily be calculated

$$\lambda_1=\frac{1}{2}(C_{11}-C_{12})\quad \lambda_2=C_{44}\quad \lambda_3=\frac{1}{2}(C_{11}+C_{12}+2C_{44}) \tag{1.12}$$

the eigenvectors indicate the polarization of the sound wave and for the above case are (-1,1,0), (0,0,1) and (1,1,0) respectively. This result (eq. 1.12) is well known in Kittel, Charles - Introduction to Solid State Physics 8Th Edition page 80.

1.2 Example: How to calculate the solution to the propagation of sound for Orthorhombic crystals for the directions K<1,0,1> With No Shear

Having $\hat{k}=\frac{1}{\sqrt{2}}(1,0,1)$ the dispersion matrix L_{ik} is given by

$$L_{ik}=\frac{1}{2}\begin{pmatrix} C_{11}-P_{11}+\dots & C_{16}+C_{14}+\dots & C_{15}+C_{13}+\dots \\ C_{15}-P_{13}+\dots & C_{56}+C_{54} & C_{55}+C_{53} \\ C_{51}-P_{31}+\dots & & \\ C_{55}-P_{33} & C_{66}-P_{11}+\dots & \\ & C_{64}-P_{13}+\dots & \\ C_{16}+C_{14}+\dots & C_{46}-P_{31}+\dots & C_{65}+C_{63}+\dots \\ C_{56}+C_{54} & C_{44}-P_{33} & C_{45}+C_{43} \\ & & C_{55}-P_{11}+\dots \\ C_{15}+C_{13}+\dots & C_{65}+C_{63}+\dots & C_{53}-P_{13}+\dots \\ C_{55}+C_{53} & C_{45}+C_{43} & C_{35}-P_{31}+\dots \\ & & C_{33}-P_{33} \end{pmatrix} \tag{1.13}$$

Now introducing Orthorhombic symmetry (see Table 1.1), we can neglect elastic coefficients which will be zero and no shear on the system thus $P_{12}=P_{13}=P_{23}=0$, for orthorhombic system, L_{ik} is given by

$$L_{ik}=\frac{1}{2}\begin{pmatrix} C_{11}-P_{11}+C_{55}-P_{33} & 0 & C_{13}+C_{55} \\ 0 & -P_{11}+C_{66}-P_{33}+C_{44} & 0 \\ C_{13}+C_{55} & 0 & C_{55}-P_{11}-P_{33}+C_{33} \end{pmatrix} \tag{1.14}$$

The eigenvalues for the matrix L_{ik} are given by

$$\begin{aligned} \lambda_1&=\left(\frac{C_{11}+C_{33}}{2}+C_{55}-P_{11}-P_{33}\right)-\sqrt{\frac{(C_{33}-C_{11})^2}{4}+(C_{55}+C_{13})^2} \\ \lambda_2&=\left(\frac{C_{11}+C_{33}}{2}+C_{55}-P_{11}-P_{33}\right)-\sqrt{\frac{(C_{33}-C_{11})^2}{4}+(C_{55}+C_{13})^2} \\ \lambda_3&=C_{44}+C_{66}-P_{11}-P_{33} \end{aligned} \tag{1.15}$$

and the eigenvectors are

$$\overrightarrow{x_1}=\frac{1}{\sqrt{2}}\left(1+\frac{a}{\sqrt{a^2+4b^2}}\right)^{1/2}\left(\frac{a}{2b}-\frac{\sqrt{a^2+4b^2}}{2b},0,1\right)$$

$$\overrightarrow{x_2} = \frac{1}{\sqrt{2}}\left(1 - \frac{a}{\sqrt{a^2+4b^2}}\right)^{1/2}\left(\frac{a}{2b} + \frac{\sqrt{a^2+4b^2}}{2b}, 0, 1\right) \quad (1.16)$$

$$\overrightarrow{x_3} = (0,1,0)$$

here $a = C_{11} - C_{33}$ and $b = C_{13} + C_{55}$

II. The general Symbolic solution to the problem of Calculating the velocity of sound on crystalline systems for an arbitrary direction <K>

In sections 1.1 and 1.2, I provided examples of solving the Christoffel- modified equation for a non-zero stress scenario (dispersion matrix) for standard propagation directions using the crystalline group symmetries to reduce terms and obtain a closed-form expression. It's also possible to use a symbolic computation. The computer code is relatively easy to implement. I use the software MATHEMATICA version 13.0. [10]

The code asks only for the direction of propagation that I defined for example purposes as kwave = <1,0,0>. Notice the output dispersion matrix Lik obtained using MATHEMATICA is as on equation (1.5), yet I could enter any other arbitrary direction such as kwave = <1,1,1>, kwave = <1,1,0>, etc., and obtain the corresponding dispersion Matrix. However, it's only possible to find the eigenvalues and the eigenvectors by hand for minimal particular scenarios with the help of the crystal symmetries provided in Table 1.1. The code written in MATHEMATICA is shown below:

Computer Code 2.1: Written Code in MATHEMATICA for the symbolic solution to the Christoffel- modified equation for a non-zero stress scenario eq. (1.2). The input variable is the direction of propagation kwave.

```
In[1]:= kwave = {1, 0, 0}
  kn = kwave/(kwave . kwave)^(1/2)
  Out[1]= {1, 0, 0}
  Out[2]= {1, 0, 0}

  In[3]:=
```

$$\text{Lik} = \text{Table}\left[\sum_{j=1}^{3}\sum_{l=1}^{3} \text{kn}[[j]]\text{kn}[[l]]\left(C_{i,j,k,l} - \delta_{i,k}P_{j,l}\right), \{i,1,3\}, \{k,1,3\}\right];$$

```
  Lik // MatrixForm
    Eigenvalues[Lik] ;
  Out[4]//MatrixForm=
```

$$\begin{pmatrix} C_{1,1,1,1} - P_{1,1} & C_{1,1,2,1} & C_{1,1,3,1} \\ C_{2,1,1,1} & C_{2,1,2,1} - P_{1,1} & C_{2,1,3,1} \\ C_{3,1,1,1} & C_{3,1,2,1} & C_{3,1,3,1} - P_{1,1} \end{pmatrix}$$

2.1 The general Symbolic solution to the problem of Calculating the velocity of sound on crystalline systems for an arbitrary direction <K> implementing the Voigt notation

In Computer Code 2.1, I used Mathematica to find the solution to the Christoffel- modified equation selection symbolically. I can adjust the same code to find the eigenvalues for any direction <K> and take the Voigt notation into account. I'm showing below the code I wrote in Mathematica (Computer Code 2.2):

Computer Code 2.2: Written Code in MATHEMATICA for the symbolic solution to the Christoffel- modified equation and implementing the Voigt notation

```
In[1]:= kwave = {1, 0, 0}
  k = kwave/(kwave . kwave)^(1/2)
  Out[1]= {1, 0, 0}
  Out[2]= {1, 0, 0}

  In[3]:= stress = {{PXX, PXY, PXZ}, {PYX, PYY, PYZ}, {PZX, PZY, PZZ}}
  Out[3]= {{PXX, PXY, PXZ}, {PYX, PYY, PYZ}, {PZX, PZY, PZZ}}

  In[4]:= Cnm =
    Table[$C_{n, m}$, {n, {1, 6, 5}}, {m, {1, 6, 5}}];
```

```
L11 = Sum[ (-stress[[j, l]] + Cnm[[j, l]])*k[[j]]*k[[l]], {j, 3}, {l,
   3}]

Out[5]=
```

$$C_{1,1} - \text{PXX}$$

```
In[6]:= Cnm =
  Table[$C_{n, m}$, {n, {1, 6, 5}}, {m, {6, 2, 4}}];
L12 = Sum[ (Cnm[[j, l]])*k[[j]]*k[[l]], {j, 3}, {l, 3}]
Out[7]=
```

$$C_{1,6}$$

```
  In[8]:= Cnm =
  Table[$C_{n, m}$, {n, {1, 6, 5}}, {m, {5, 4, 3}}];
L13 = Sum[ (Cnm[[j, l]])*k[[j]]*k[[l]], {j, 3}, {l, 3}]
Out[9]=
```

$$C_{1,5}$$

```
  In[10]:=
  Cnm = Table[$C_{n, m}$, {n, {6, 2, 4}}, {m, {6, 2, 4}}];
  L22 = Sum[ (-stress[[j, l]] + Cnm[[j, l]])*k[[j]]*k[[l]], {j, 3}, {l,
     3}]
  Out[11]=
```

$$C_{6,6} - \text{PXX}$$

```
  In[12]:=
  Cnm = Table[$C_{n, m}$, {n, {6, 2, 4}}, {m, {5, 4, 3}}];
  L23 = Sum[ (Cnm[[j, l]])*k[[j]]*k[[l]], {j, 3}, {l, 3}]
  Out[13]=
```

$$C_{6,5}$$

```
  In[14]:=
  Cnm = Table[$C_{n, m}$, {n, {5, 4, 3}}, {m, {5, 4, 3}}];
  L33 = Sum[ (-stress[[j, l]] + Cnm[[j, l]])*k[[j]]*k[[l]], {j, 3}, {l,
     3}]
  Out[15]=
```

$$C_{5,5} - \text{PXX}$$

```
  In[16]:= Lij = {{L11, L12, L13}, {L12, L22, L23}, {L13, L23, L33}};
Lij // MatrixForm
Out[17]//MatrixForm=
```

$$\begin{pmatrix} C_{1,1} - \text{PXX} & C_{1,6} & C_{1,5} \\ C_{1,6} & C_{6,6} - \text{PXX} & C_{6,5} \\ C_{1,5} & C_{6,5} & C_{5,5} - \text{PXX} \end{pmatrix}$$

```
In[18]:= (Eigenvalues[Lij]) ;
```

III. NUMERICAL SOLUTION TO THE CHRISTOFFEL EQUATION USING BCC TANTALUM AS THE TESTING MATERIAL

Here I show the numerical solution to the Christoffel equation using the software octave. The testing material here is BCC Tantalum. Therefore, we need to provide the elastic constants for Tantalum. The speed of sound calculated is in good agreement with other references. Look at D. A. Orlikowski [11] and Oguz Gülseren [12] (around 4000 m/s longitudinal and 2000 m/s transverse)

I ran 14,000 different directions of wave propagation kwave= <kx,ky,kz>. The code written running on OCTAVE version 7.2.0 is indeed relatively fast. Then I calculated the speed of sound components v1,v2, and v3 in the 3D sphere region with radius of one (Figure 1.1). The vector of propagation direction <K> must be a unit vector. In other words, the norm of the vector must be equal to one.

Computer Code 3.1: Written Code in Octave for the numerical solution to the Christoffel equation and implementing the Voigt notation

```
# CALCULATION OF to the Numerical solution for the Christoffel equation
# Made by Oscar Guerrero - Miramontes

# Input Elastic constants values (For Tantalum) Pa units
# Input density for tantalum units kg/m^3

density_material = 16690.0;

C11 = 262.563*10^9;
C21 = 160.699*10^9;
C31 = 160.699*10^9;
C41 = 0;
C51 = 0;
C61 = 0;

C12 = 160.699*10^9;
C22 = 262.563*10^9;
C32 = 160.699*10^9;
C42 = 0;
C52 = 0;
C62 = 0;

C13 = 160.699*10^9;
C23 = 160.699*10^9;
C33 = 262.563*10^9;
C43 = 0;
C53 = 0;
C63 = 0;

C14 = 0;
C24 = 0;
C34 = 0;
C44 = 81.805*10^9;
C54 = 0;
C64 = 0;

C15 = 0;
C25 = 0;
C35 = 0;
C45 = 0;
C55 = 81.805*10^9;
C65 = 0;

C16 = 0;
C26 = 0;
C36 = 0;
C46 = 0;
C56 = 0;
C66 = 81.805*10^9;

# Input the Pressure here we consider the unstress scenario therefore P = 0

  PXX = 0;
  PYY = 0;
  PZZ = 0;
  PYZ = 0;
  PXZ = 0;
  PXY = 0;
```

```
  PYX = PXY;
  PZY = PYZ;
  PZX = PXZ;

# Create elastic constant and presssure tensor

sigma = [ PXX PXY PXZ ; PYX PYY PYZ ; PZX PZY PZZ ];

Cjl = [ C11 C12 C13 C14 C15 C16 ;
        C21 C22 C23 C24 C25 C26 ;
        C31 C32 C33 C34 C35 C36 ;
        C41 C42 C43 C44 C45 C46 ;
        C51 C52 C53 C54 C55 C56 ;
        C61 C62 C63 C64 C65 C66 ];

#*******************************************************#
# Now we calculate the sound propagation in the medium  #
#*******************************************************#

# Define a direction of propagation.
# Generate points sampled uniformly at a sphere r = 1
# we only care about the Unit Vector
# using sphere(np) creates a np+1 x np + 1 matrix
# the number of rows [kx,ky,kz] are 3*(np+1)*(np+1)

np = 150;
[X,Y,Z] = sphere(np);
kx = [0.5*X(:); 0.75*X(:); X(:)];
ky = [0.5*Y(:); 0.75*Y(:); Y(:)];
kz = [0.5*Z(:); 0.75*Z(:); Z(:)];

clear X
clear Y
clear Z

# We only want the vectors [kx,ky,kz] have a norm = 1 (surface sphere)

newElem = 1;
for loop_kn = 1:3*(np+1)*(np+1)
tmpV = [kx(loop_kn,1),ky(loop_kn,1),kz(loop_kn,1)];
if(norm(tmpV)==1)
ksurfx(newElem,:) = kx(loop_kn,1);
ksurfy(newElem,:) = ky(loop_kn,1);
ksurfz(newElem,:) = kz(loop_kn,1);
newElem = newElem + 1;
endif
endfor

clear kx
clear ky
clear kz

for loop_kn = 1:newElem-1

kn = [ksurfx(loop_kn,1),ksurfy(loop_kn,1),ksurfz(loop_kn,1)];

#Calculate the matrix component L11

irow = 1;
icol = 1;
for n = [1,6,5]
 for m = [1,6,5]
```

```
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L11 = 0;
for j = 1:3
 for l = 1:3
addition = (-sigma(j,l) + Cnm(j,l))*kn(j)*kn(l);
L11 = L11 + addition;
  endfor
    endfor

#Calculate the matrix component L12
irow = 1;
icol = 1;
for n = [1,6,5]
 for m = [6,2,4]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L12 = 0;
for j = 1:3
 for l = 1:3
addition = ( Cnm(j,l))*kn(j)*kn(l);
L12 = L12 + addition;
  endfor
    endfor

#Calculate the matrix component L13
irow = 1;
icol = 1;
for n = [1,6,5]
 for m = [5,4,3]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L13 = 0;
for j = 1:3
 for l = 1:3
addition = (Cnm(j,l))*kn(j)*kn(l);
L13 = L13 + addition;
  endfor
    endfor

#Calculate the matrix component L22
irow = 1;
icol = 1;
for n = [6,2,4]
 for m = [6,2,4]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
```

```
     endfor
L22 = 0;
for j = 1:3
 for l = 1:3
addition = (-sigma(j,l) + Cnm(j,l))*kn(j)*kn(l);
L22 = L22 + addition;
  endfor
    endfor

#Calculate the matrix component L23
irow = 1;
icol = 1;
for n = [6,2,4]
 for m = [5,4,3]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L23 = 0;
for j = 1:3
 for l = 1:3
addition = (Cnm(j,l))*kn(j)*kn(l);
L23 = L23 + addition;
  endfor
    endfor

#Calculate the matrix component L33
irow = 1;
icol = 1;
for n = [5,4,3]
 for m = [5,4,3]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L33 = 0;
for j = 1:3
 for l = 1:3
addition = (-sigma(j,l) + Cnm(j,l))*kn(j)*kn(l);
L33 = L33 + addition;
  endfor
    endfor

# Create the dispersion matrix
Lij = [ L11 L12 L13 ; L12 L22 L23 ; L13 L23 L33 ];

# calculate the speed component by taking the eigenvalues of the dispersion    # matrix,
remember pv2 = eigenvalues(Lij) the units here are m/s

[evector,evalues] = eig(Lij);
speed(1,:) = sqrt(diag(evalues)./density_material);
evector1 = transpose(evector(:,1));
evector2 = transpose(evector(:,2));
evector3 = transpose(evector(:,3));

output1(loop_kn,:)= [ loop_kn speed kn evector1 evector2 evector3 norm(kn)];

endfor
```

```
# store speed component to an outputfile
# the output file ispeed have the columns:
# np, v1,v2,v3 ,kx,ky,kz,evector1, evector2, evector3, norm

dlmwrite('ispeed.csv',output1,'delimiter','\t','precision','%4.6f');

clf;
colormap ("default");
# Make 3D plot with filled circles coloured in the scheme of the
# measurements
s = scatter3(output1(:,5),output1(:,6),output1(:,7),10,output1(:,3));
# view (3) sets the viewpoint to azimuth = -37.5 and elevation = 30, which is the default
for 3-D graphs
view(3);
caxis([min(output1(:,3)),max(output1(:,3))])
colorbar; % Throw in the colour bar.
waitfor(1)

clear
quit
```

The velocities obtained from Code 3.1 are shown below (Figure 3.1):

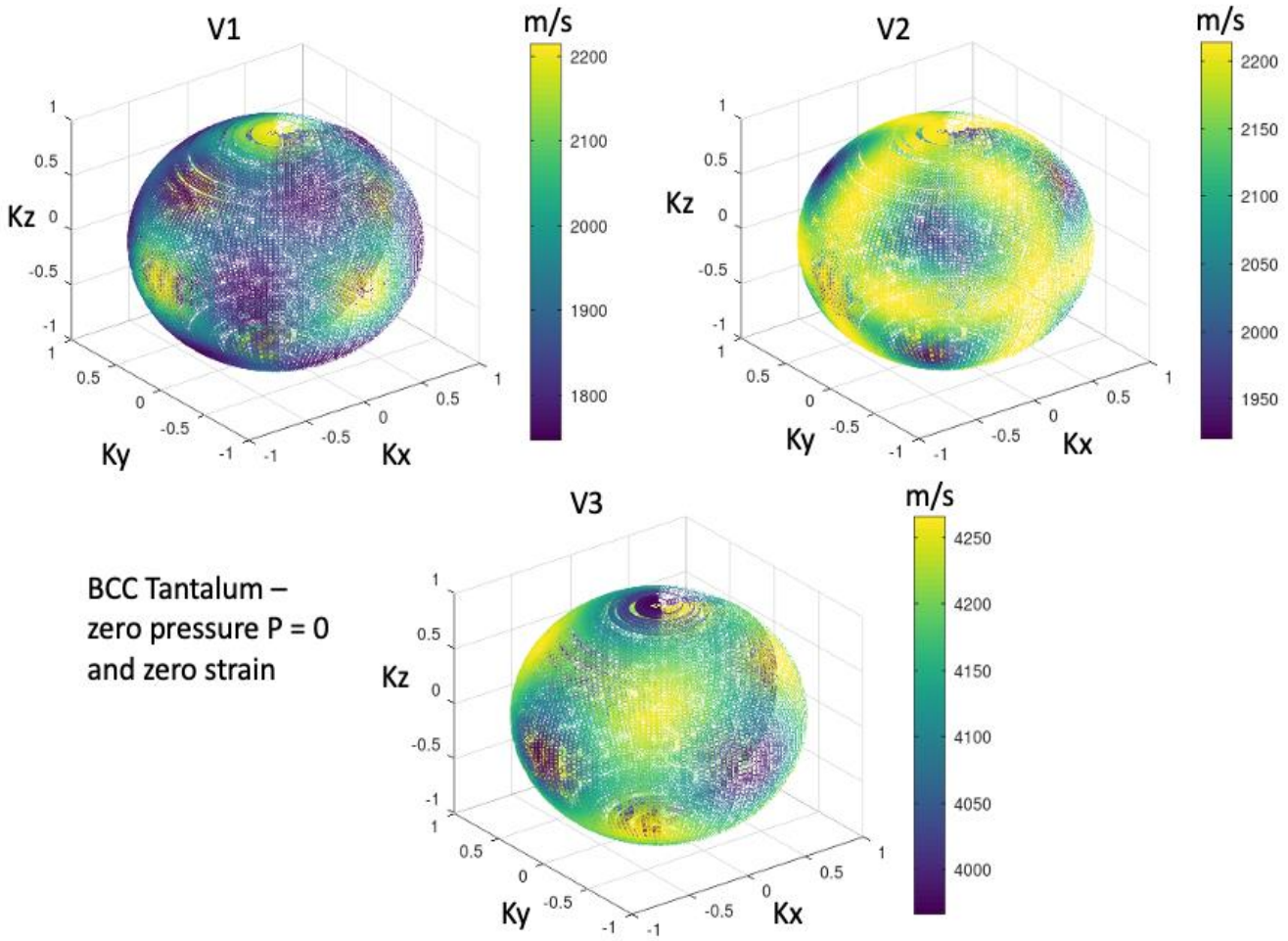

Figure 3.1 Calculation of the speed of sound v1, v2 and v3 for BCC Tantalum for around 14,000 different directions of wave propagation.

IV. PREDICTION OF PRESSURE INDUCE INSTABILITIES

In the last chapter, I explored solving the Christoffel equation for the case of an unstressed crystal lattice. I'm now ready to go into explaining the solution to the Christoffel-modified equation eq. (1.2) when external pressure is applied to the material. As the external

pressure/strain increases, there will be specific directions <k> where the speed of sound may go to zero. The dynamical instability is defined as:

$$\lambda_s = \rho v_s^2 = 0 \tag{4.1}$$

The null velocity is equivalent to the softening of acoustic phonon branches evaluated at $dw/dq|_0$, producing negative (imaginary frequencies). These dynamical instabilities are responsible for the elastic-plastic transition above Mbar pressure. You can read the work by O. Guerrero et al., A. Landa et al., S.M Shapiro et al.,S.V. Makarov et al. and N. Amadou [1,13-18] I implemented the following pseudo-code to find the phonon instabilities in Perfect singles Crystals. The algorithm uses the stress-strain relations obtained from an external MD program (such as LAMMPS) [19]. For this purpose, we use the free open-source OCTAVE [20] for the numerical solution.

The pseudo-code that is required to find the phonon instabilities:

1. Find the stress-strain relation e.g Pxx , Pzx , Pzz along the direction of compression.
2. Calculate the elastic-stiffness coefficients $B_{ijkl} = {d\sigma_{ij}}/{d\varepsilon_{kl}}$
3. Construct the Dispersion Matrix and evaluate the sound propagation for a given wave vector direction $< k >$
4. If the speed of sound (eigenvalue) $\lambda_s = \rho v_s^2 = 0$, then store the critical strain value in memory
5. Repeat step (4) , but for a different value of $< k >$, and store the new critical strain value
6. From all the random directions of propagation, choose the direction $< k >$, in which the critical strain is the lower

4.1 Prediction Phonon Instabilities for Uniaxial Compression along [100] BCC Tantalum

I will provide the prediction of phonon instabilities for Uniaxial compression along [100] in BCC Tantalum. The obtention of the stress-strain relations is using the Ravelo Potential [21-22] using the Molecular dynamics program LAMMPS [19]. I used the plotting program WebPlot Digitizer [23] to verify that the pressure-strain relations obtained using the software LAMMPS match the values in Ravelo's publication (see figure 4.2). The Ravelo Potential offers a good match for the elastic constants vs. DFT calculation. Of course, the methodology explained in this chapter is independent of the interatomic interaction, In other words If another interaction potential is used as example the MEAM, ADP, MGPT or Experimental dataset, all the calculated values may be different and the instability while might occur at different compressive strains, the observed mechanism remains the same. When BCC Tantalum structure (cubic crystal) is compressed along the x-axis [100] it changes to a Tetragonal crystal lattice. The image (4.1) provides more clarification on the change in symmetry.

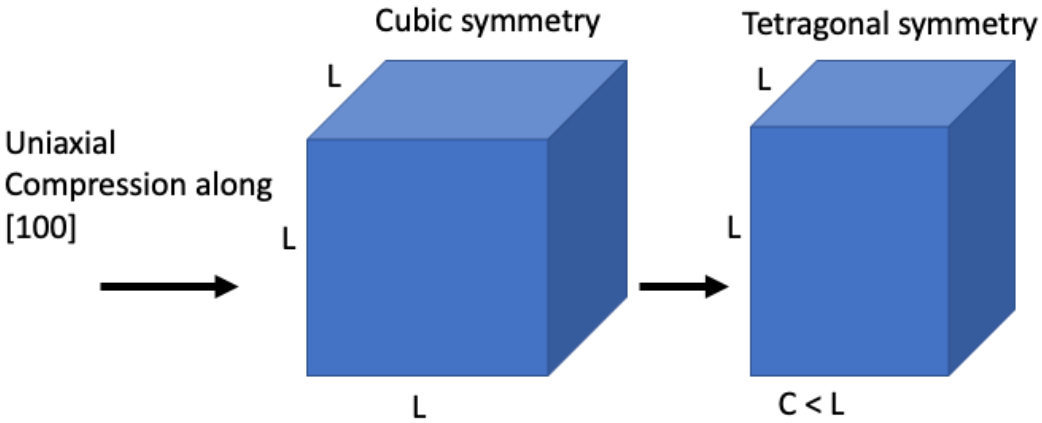

Figure 4.1: The Tetragonal crystal lattices result from stretching/compressing a cubic lattice along one of its lattice vectors so that the cube becomes a rectangular prism with a square base (L by L) and height (C, which is different from L).

For tetragonal I structure there are six independent elastic constants

$$\begin{pmatrix}\sigma_1\\ \sigma_2\\ \sigma_3\\ \sigma_4\\ \sigma_5\\ \sigma_6\end{pmatrix} = \begin{pmatrix} B_{11} & B_{12} & B_{13} & 0 & 0 & 0\\ B_{12} & B_{11} & B_{13} & 0 & 0 & 0\\ B_{13} & B_{13} & B_{33} & 0 & 0 & 0\\ 0 & 0 & 0 & B_{44} & 0 & 0\\ 0 & 0 & 0 & 0 & B_{44} & 0\\ 0 & 0 & 0 & 0 & 0 & B_{66}\end{pmatrix}\begin{pmatrix}\varepsilon_1\\ \varepsilon_2\\ \varepsilon_3\\ \varepsilon_4\\ \varepsilon_5\\ \varepsilon_6\end{pmatrix} \tag{4.2}$$

Using pseudocode steps 1 through 6, I modified the computer code 3.1 in OCTAVE to solve equation (1.2) and ran around 14,000 <K> points. The results show the lower critical strain in which the velocity of sound is null is around 0.15709 along the direction <k> = <1,0,0>. The information of the critical strain found numerically is saved to an output file with the name of *icritical.csv* (look at image 4.2)

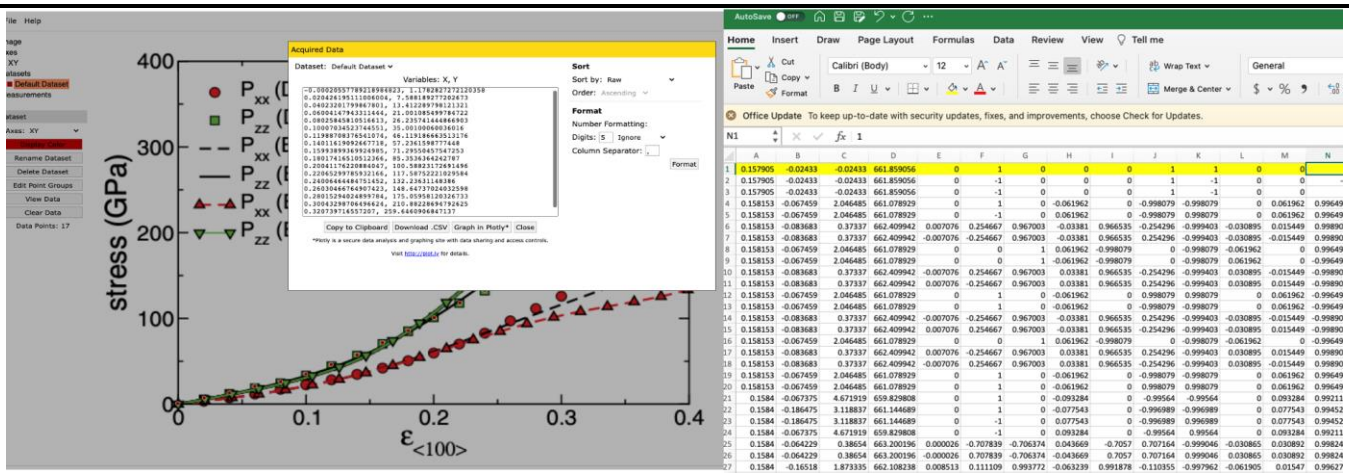

Figure 4.2: Left image: I used the plotting program WebPlot Digitizer to verify that the pressure-strain relations obtained using the software LAMMPS match the values in Ravelo's publication. The Ravelo Potential offers a good match for the elastic constants vs. DFT calculation. Right image: critical strain values found numerically using computer code 2.1 saved to an output file with the name of icritical.csv

The waves propagated along K <1,0,0> can be classified as a pure mode direction, which has one pure longitudinal wave and two transverse purse waves. This can be easily proven by solving the dispersion matrix by hand which yield the eigenvalues $\lambda_1 = B_{11} - P_{11}$ $\lambda_2 = B_{66} - P_{11}$ $\lambda_3 = B_{44} - P_{11}$ Note for tetragonal symmetry have two equal shear elastic constants (B44 = B55 and B66). The eigenvectors indicate the polarization of the sound wave and for the above case are (1,0,0) , (0,1,0) and (0,0,1) respectively I can visually confirm the critical instability found using the Numerical methods; the critical strain starts around 0.15709. The image (4.3) shows the eigenvalues λ_1, λ_2 and λ_3 for the direction K <1,0,0> as function of the compressive strain

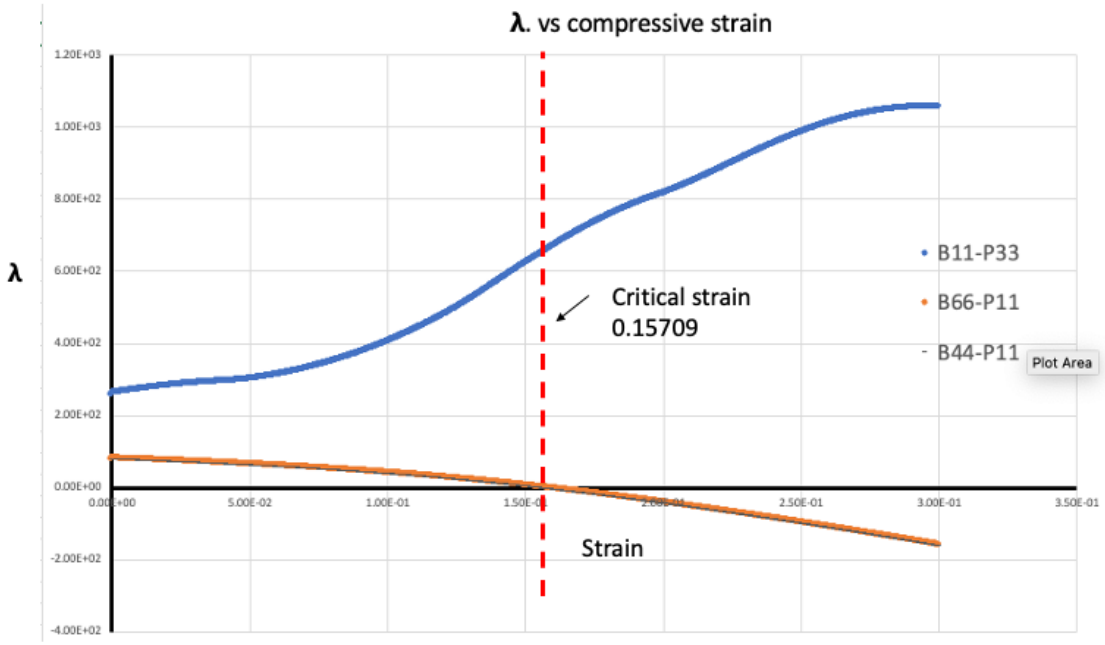

*Figure 4.3: Eigenvalues λ_1, λ_2 and λ_3 for the direction K <1,0,0> as function of the compressive strain for Uniaxial Compression BCC Tantalum along [100] the critical strain is found around 0.15709 *Note tetragonal symmetry have two equal shear elastic constants (B44 = B55 and B66) and two equal tensile-compliant elastic constants (B12 = B23 and B13)*

The code implemented to solve the modified Christoffel equation is below:

Computer Code 4.1: Written Code in Octave for the Prediction phonon instabilities solving numerical the Modified Christoffel equation

```
# CALCULATION OF CRITICAL STRAIN VALUE
# Made by Oscar Guerrero - Miramontes

# The dynamical instability is given by the condition:
# pv^2 =0 -> v^2 = 0 here we set the density to one

density_material = 1

# stress.dat have the following format
# strain, pxx , pyy, pzz, pyz, pxz, pxy
# calculation of the elastic constants depends on what file your loading
# data1 = xx data2 = yy data3 = zz data4 = yz data5 = xz data6 = xy
```

```
data1 = load ("-ascii",  "stress.1.dat") ;
data2 = load ("-ascii",  "stress.2.dat") ;
data3 = load ("-ascii",  "stress.3.dat") ;
data4 = load ("-ascii",  "stress.4.dat") ;
data5 = load ("-ascii",  "stress.5.dat") ;
data6 = load ("-ascii",  "stress.6.dat") ;

# calculation of the elastic constants Bijkl = dStress_ij/dStrain_kl

# Deformation in the X-DIRECTION
# 1 = xx 2 = yy 3 = zz 4 = yz 5 = xz 6 = xy

B11 = diff(data1(:,2))./diff(data1(:,1)) ;
B21 = diff(data1(:,3))./diff(data1(:,1)) ;
B31 = diff(data1(:,4))./diff(data1(:,1)) ;
B41 = diff(data1(:,5))./diff(data1(:,1)) ;
B51 = diff(data1(:,6))./diff(data1(:,1)) ;
B61 = diff(data1(:,7))./diff(data1(:,1)) ;

# Deformation in the Y-Direction

B12 = diff(data2(:,2))./diff(data2(:,1)) ;
B22 = diff(data2(:,3))./diff(data2(:,1)) ;
B32 = diff(data2(:,4))./diff(data2(:,1)) ;
B42 = diff(data2(:,5))./diff(data2(:,1)) ;
B52 = diff(data2(:,6))./diff(data2(:,1)) ;
B62 = diff(data2(:,7))./diff(data2(:,1)) ;

# Deformation in the Z - Direction

B13 = diff(data3(:,2))./diff(data3(:,1)) ;
B23 = diff(data3(:,3))./diff(data3(:,1)) ;
B33 = diff(data3(:,4))./diff(data3(:,1)) ;
B43 = diff(data3(:,5))./diff(data3(:,1)) ;
B53 = diff(data3(:,6))./diff(data3(:,1)) ;
B63 = diff(data3(:,7))./diff(data3(:,1)) ;

# Deformation in the YZ - Direction

B14 = diff(data4(:,2))./diff(data4(:,1)) ;
B24 = diff(data4(:,3))./diff(data4(:,1)) ;
B34 = diff(data4(:,4))./diff(data4(:,1)) ;
B44 = diff(data4(:,5))./diff(data4(:,1)) ;
B54 = diff(data4(:,6))./diff(data4(:,1)) ;
B64 = diff(data4(:,7))./diff(data4(:,1)) ;

# Deformation in the XZ - Direction

B15 = diff(data5(:,2))./diff(data5(:,1)) ;
B25 = diff(data5(:,3))./diff(data5(:,1)) ;
B35 = diff(data5(:,4))./diff(data5(:,1)) ;
B45 = diff(data5(:,5))./diff(data5(:,1)) ;
B55 = diff(data5(:,6))./diff(data5(:,1)) ;
B65 = diff(data5(:,7))./diff(data5(:,1)) ;

# Deformation in the XY - Direction

B16 = diff(data6(:,2))./diff(data6(:,1)) ;
B26 = diff(data6(:,3))./diff(data6(:,1)) ;
B36 = diff(data6(:,4))./diff(data6(:,1)) ;
B46 = diff(data6(:,5))./diff(data6(:,1)) ;
B56 = diff(data6(:,6))./diff(data6(:,1)) ;
```

```
B66 = diff(data6(:,7))./diff(data6(:,1)) ;

#****************************************#

# Save the stress information into variables but remove the last row in the array
# this is done with the purpose that the elastic constants and the stress tensors
# have the same dimension length

mynum = length(data1(:,1)) - 1 ;

ESTRAIN = data1(1:mynum,1);
    PXX = data1(1:mynum,2);
    PYY = data1(1:mynum,3);
    PZZ = data1(1:mynum,4);
    PYZ = data1(1:mynum,5);
    PXZ = data1(1:mynum,6);
    PXY = data1(1:mynum,7);

  PYX = PXY;
  PZY = PYZ;
  PZX = PXZ;

#****************************************#

# store pressure into archive istress.dat
# store elastic constant into archive ieconst.dat

# save data into archive

dlmwrite('ieconst.csv',[ ESTRAIN B11 B12 B13 B14 B15 B16 B22 B23 B24 B25 B26 B33 B34 B35
B36 B44 B45 B46 B55 B56 B66 ],'delimiter','\t', 'precision','%.6E')
dlmwrite('istress.csv',[ ESTRAIN PXX PYY PZZ PYZ PXZ PXY ],'delimiter','\t', 'precision',
'%.6E')

#*****************************************************#
# Now we calculate the sound propagation in the medium   #
#*****************************************************#

# myflag variable is a sentinel help us find when v^2 = 0
myflag = 1;

# Generate points sampled uniformly at a sphere r = 1
# we only care about the Unit Vector
# using sphere(np) creates a np+1 x np + 1 matrix
# the number of rows [kx,ky,kz] are 3*(np+1)*(np+1)

np = 150;
[X,Y,Z] = sphere(np);
kx = [0.5*X(:); 0.75*X(:); X(:)];
ky = [0.5*Y(:); 0.75*Y(:); Y(:)];
kz = [0.5*Z(:); 0.75*Z(:); Z(:)];

clear X
clear Y
clear Z

# We only want the vectors [kx,ky,kz] have a norm = 1 (surface sphere)

newElem = 1;
for loop_kn = 1:3*(np+1)*(np+1)
tmpV = [kx(loop_kn,1),ky(loop_kn,1),kz(loop_kn,1)];
if(norm(tmpV)==1)
```

```
ksurfx(newElem,:) = kx(loop_kn,1);
ksurfy(newElem,:) = ky(loop_kn,1);
ksurfz(newElem,:) = kz(loop_kn,1);
newElem = newElem + 1;
endif
endfor

clear kx
clear ky
clear kz

for loop_kn = 1:newElem-1

disp("Running step, step= "), disp(loop_kn), disp("out of "), disp(newElem);

kn = [ksurfx(loop_kn,1),ksurfy(loop_kn,1),ksurfz(loop_kn,1)];

# Calculate for all strain values
# begins for loop until end of data

for i = 1:mynum

sigma = [ PXX(i), PXY(i), PXZ(i) ; PYX(i), PYY(i), PYZ(i) ; PZX(i), PZY(i), PZZ(i) ];

# Notice we are using the Tetragonal I symmetry for Cjl
Cjl = [ B11(i), B12(i), B13(i), 0     , 0     , 0 ;
        B12(i), B11(i), B13(i), 0     , 0     , 0 ;
        B13(i), B13(i), B33(i), 0     , 0     , 0 ;
        0     , 0     , 0     , B44(i), 0     , 0 ;
        0     , 0     , 0     , 0     , B44(i), 0 ;
        0     , 0     , 0     , 0     , 0     , B66(i) ];

#Calculate the matrix component L11

irow = 1;
icol = 1;
for n = [1,6,5]
 for m = [1,6,5]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L11 = 0;
for j = 1:3
 for l = 1:3
addition = (-sigma(j,l) + Cnm(j,l))*kn(j)*kn(l);
L11 = L11 + addition;
  endfor
    endfor

#Calculate the matrix component L12
irow = 1;
icol = 1;
for n = [1,6,5]
 for m = [6,2,4]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
```

```
     endfor
L12 = 0;
for j = 1:3
 for l = 1:3
addition = ( Cnm(j,l))*kn(j)*kn(l);
L12 = L12 + addition;
  endfor
    endfor

#Calculate the matrix component L13
irow = 1;
icol = 1;
for n = [1,6,5]
 for m = [5,4,3]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L13 = 0;
for j = 1:3
 for l = 1:3
addition = (Cnm(j,l))*kn(j)*kn(l);
L13 = L13 + addition;
  endfor
    endfor

#Calculate the matrix component L22
irow = 1;
icol = 1;
for n = [6,2,4]
 for m = [6,2,4]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L22 = 0;
for j = 1:3
 for l = 1:3
addition = (-sigma(j,l) + Cnm(j,l))*kn(j)*kn(l);
L22 = L22 + addition;
  endfor
    endfor

#Calculate the matrix component L23
irow = 1;
icol = 1;
for n = [6,2,4]
 for m = [5,4,3]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L23 = 0;
for j = 1:3
 for l = 1:3
addition = (Cnm(j,l))*kn(j)*kn(l);
```

```
L23 = L23 + addition;
  endfor
    endfor

#Calculate the matrix component L33
irow = 1;
icol = 1;
for n = [5,4,3]
 for m = [5,4,3]
   Cnm(irow,icol) = Cjl(n,m);
   icol = icol + 1;
    endfor
   icol = 1;
   irow = irow + 1;
     endfor
L33 = 0;
for j = 1:3
 for l = 1:3
addition = (-sigma(j,l) + Cnm(j,l))*kn(j)*kn(l);
L33 = L33 + addition;
  endfor
    endfor

# Create the dispersion matrix
Lij = [ L11, L12, L13 ; L12, L22, L23 ; L13, L23, L33 ];

# calculate the speed component by taking the eigenvalues of the dispersion matrix

[evector,evalues] = eig(Lij);
myspeed(1,:) = diag(evalues);
evector1 = transpose(evector(:,1));
evector2 = transpose(evector(:,2));
evector3 = transpose(evector(:,3));

if ( (myspeed(1) <= 0) || (myspeed(2) <= 0) || (myspeed(3) <= 0))
 disp("Found negative eigenvalue \n");
 output2(myflag,:) = [ ESTRAIN(i) myspeed evector1 evector2 evector3 kn ];
 myflag = myflag + 1 ;
 break % quit the loop i = 1:mynum
endif

endfor
 endfor

# store speed component and critical strain value to an outputfile
dlmwrite('icritical.csv',output2,'delimiter','\t','precision','%4.6f') ;

clear
quit
```

Finally, I Plot the calculated eigenvalues λ_1, λ_2 and λ_3 for around 14,000 <K> points in the 3D sphere region with a radius of one at 0% strain and close to the critical strain value 16%. (Figure 4.4). The vector of propagation direction <K> must be a unit vector. In other words, the norm of the vector must be equal to one.

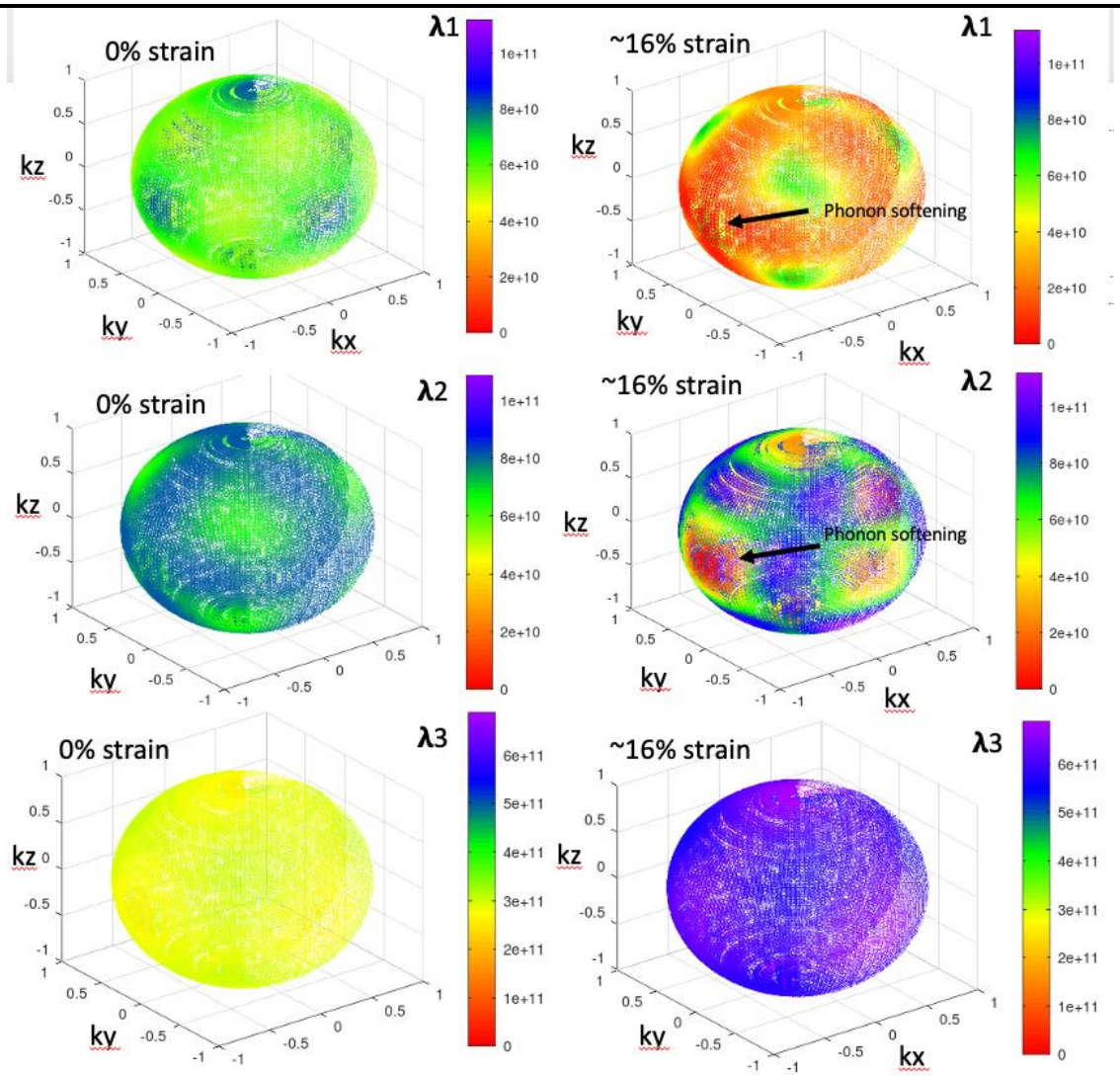

Figure 4.4: Calculation of the eigenvalues λ_1, λ_2 and λ_3 for BCC Tantalum Uniaxial Compressed along [100] for around 14,000 different directions of wave propagation <K> at 0% strain and close to the critical strain value 16% The appearance of values close to null indicates the formation of phonon instabilities (red color)

4.1.1 Molecular Dynamics simulation of BCC Tantalum Compressed along [100] vs The Prediction Phonon instabilities

I ran molecular dynamics simulation of BCC Tantalum uniaxially compressed along the direction [100] at 300K using the Ravelo Potential. The elastic-plastic transition observed by the abrupt decay of the shear $\tau = 0.5(Pxx - 0.5(Pyy + Pzz))$ was correlated with the critical strain obtained from the computer code. The elastic-plastic transition does connect with the critical strain value found using my computer code 4.1. (See Figure 4.5). The molecular dynamics simulation was run using the software LAMMPS using the methodology by O. Guerrero in the publication: A beginner's guide to the modeling of shock/uniaxial/quasi-isentropic compression using the LAMMPS molecular dynamics simulator By: Oscar Guerrero-Miramontes [24] .The strain rate was selected to approximate the rise times of the shock profiles in the NEMD simulations, typically 10^{10}-10^{11} s^{-1}

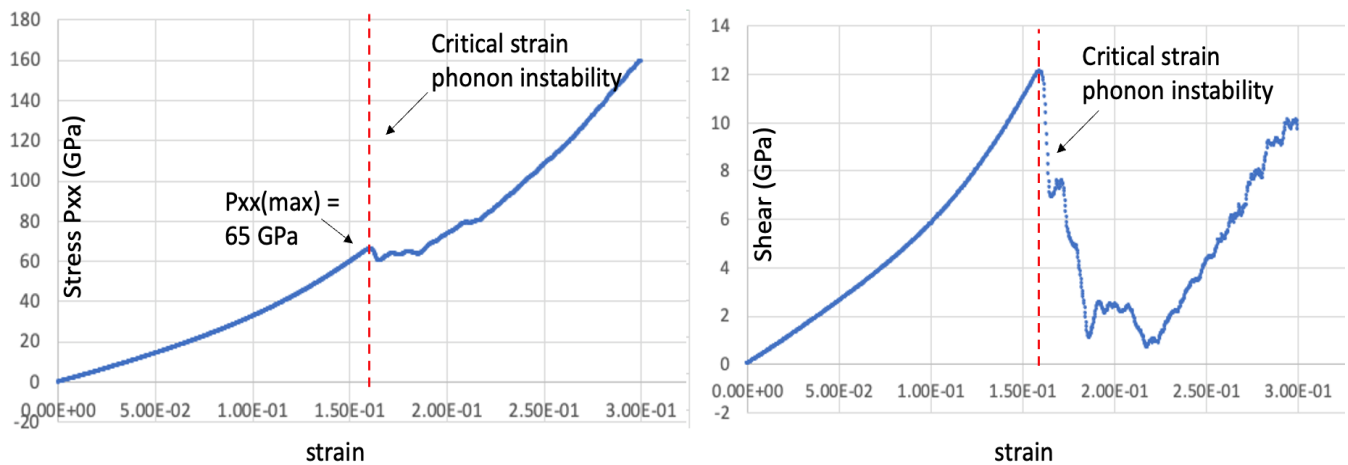

Figure 4.5 Molecular dynamics simulation of BCC tantalum compressed along [100] using around 4,000 atoms at 300K. The image on the left show pressure (Pxx vs. Strain), and the figure on the right show the shear $\tau = 0.5\big(Pxx - 0.5(Pyy + Pzz)\big)$ *vs. strain. The critical strain in which the phono instability occurs match the elastic-plastic transition (red dash line)*

The elastic-plastic transition determined via MD was never found below the soft-mode dynamical instability critical strain

4.2 Molecular Dynamics Simulation of FCC Copper Compressed along [100] vs the prediction phonon instabilities

I also ran a Molecular dynamics simulation in FCC Copper using the Mishin potential [25] and correlated the critical strain found using my Octave Code running at 14,000 <K> points. The results show that the lowest critical strain begins around 0.10543 with the direction <K> = (0.728969, 0.684547,0) eig1 = (0.693, -0.720,0). Interestingly, the lowest critical strain is not a point of high symmetry in the Brillouin zone as in the case of BCC tantalum, where the phonon instability occurs at <K> = (1,0,0) eig2 = (0,1,0) and eig3 = (0,0,1). In addition, there is a second mode decay for copper around the critical strain of 0.30 given by eig2 = (0,0,1). Our Results for FCC Copper are in good agreement with the work of Giles Kimminau et al. [26] that instead determines the phonon-dispersion relation over the entire BZ (my proposed code runs much faster in predicting the phonon instabilities). The following image (4.6) shows the results of the MD simulations vs. the predicted Phonon instability critical strain

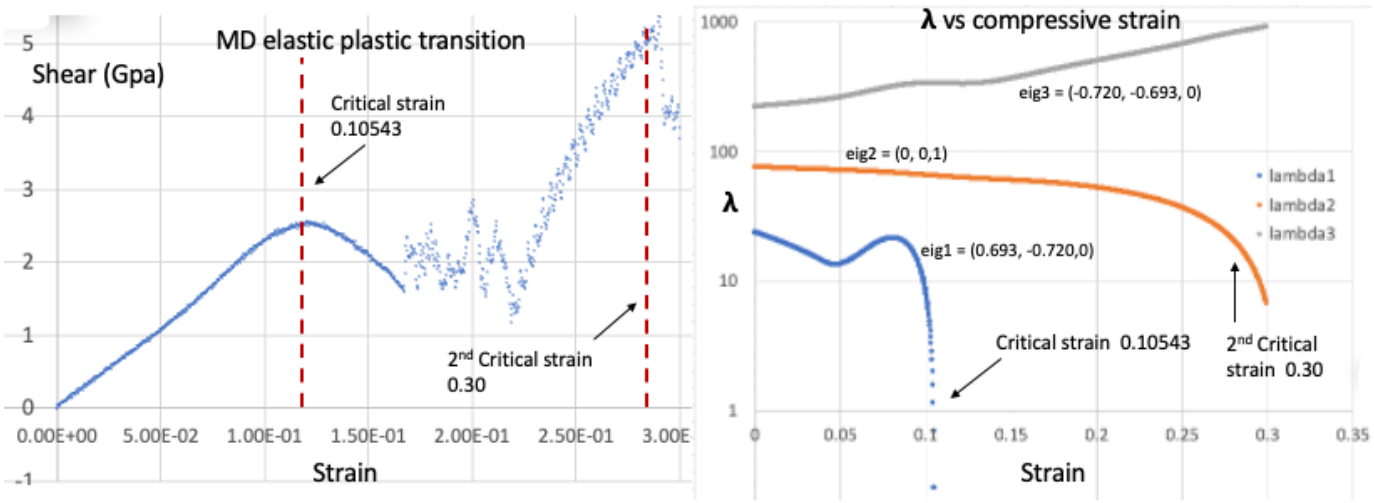

Figure 4.6: Molecular dynamics simulation of FCC Copper compressed along [100] using around 4,000 atoms at 300K. The image on the left shows the shear $\tau = 0.5\big(Pxx - 0.5(Pyy + Pzz)\big)$ *vs. strain. The image on the right shows critical strain in which the phono instability occurs using the computer code 2.1. The phonon instability occurs in the same strain as in the Molecular dynamics simulation (red dash line)*

Finally, I Plot the calculated eigenvalues λ_1, λ_2 and λ_3 for around 14,000 <K> points in the 3D sphere region with a radius of one at 0% strain and close to the critical strain value 11%. The vector of propagation direction <K> must be a unit vector. In other words, the norm of the vector must be equal to one (Figure 4.7).

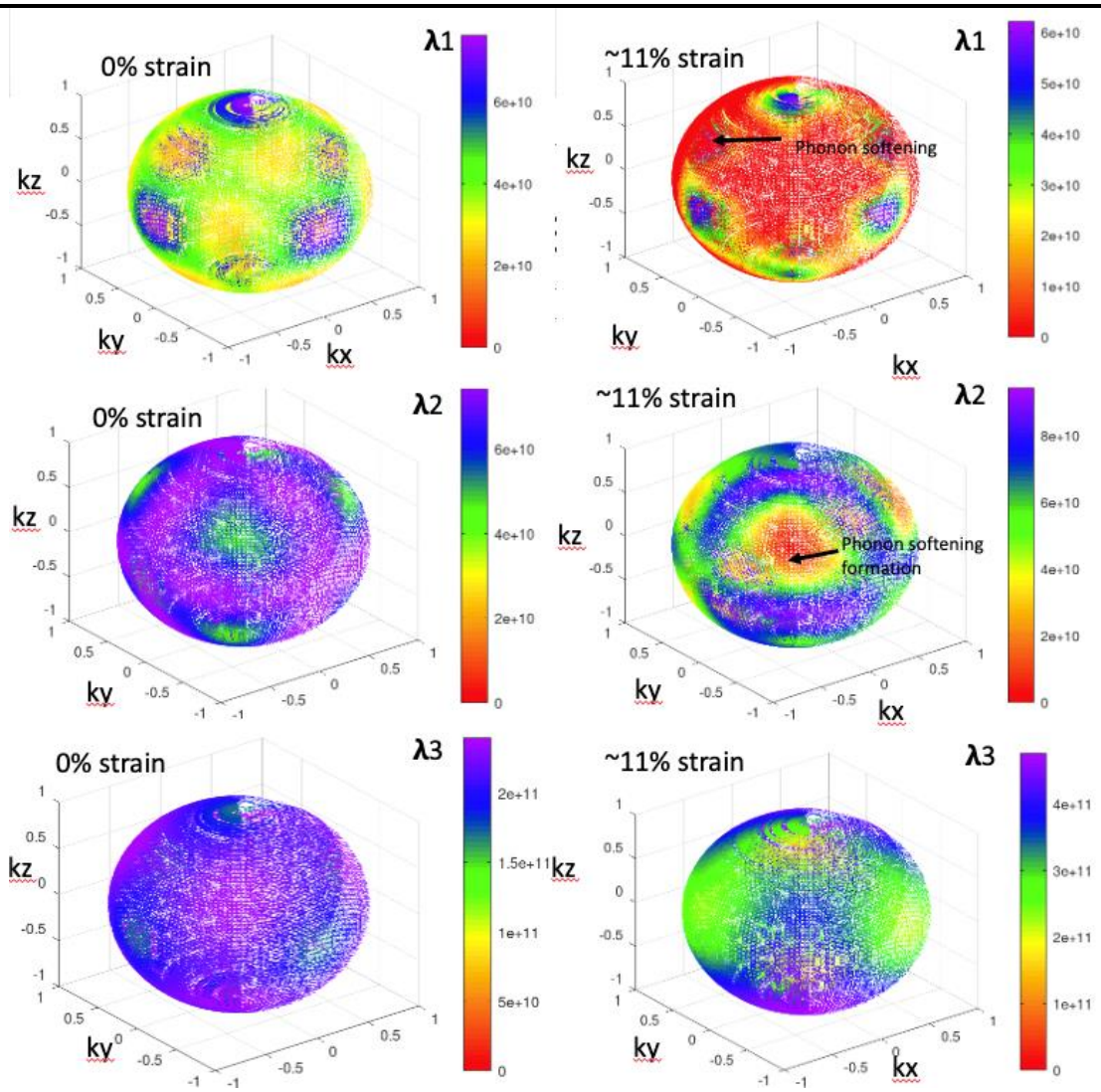

Figure 4.7: Calculation of the eigenvalues λ_1, λ_2 and λ_3 for FCC Copper Uniaxial Compressed along [100] for around 14,000 different directions of wave propagation <K> at 0% strain and close to the critical strain value 11% The appearance of values close to null indicates the formation of phonon instabilities (red color)

Again, I never found the elastic-plastic transition determined via MD below the soft-mode dynamical instability critical strain

CONCLUSIONS

I describe a computational pseudo-code for predicting pressure-induced phonon instabilities in crystals. These dynamical instabilities are one of the factors responsible for the elastic-plastic transition for BCC Tantalum and FCC Copper above Mbar pressure and at strain rates found in the range of Molecular Dynamics simulations. Although I display the use of the pseudo-code only for Uniaxial compression, I can easily alter it for other scenarios such as Hydrostatic pressure, biaxial pressure, Tri-Axial stress, etc., I could use this same approach to predict the change of electromagnetic waves' velocities under pressure using the dielectric constants [30] instead of the elastic constants; however, I'm not going to pursue further that implementation.

ACKNOWLEDGMENT

This research did not receive any grant from any University or national government agency

I want to thank a few people who assisted me in reaching the culmination of my writing: Dr. Mario Borunda (the father) was my Advisor while pursuing my bachelor's degree. I had the pleasure to meet him again in 2021 after more than 15 years without knowing about him, and he encouraged me to continue writing. Thank you to Dr. Marcelo Marucho and Dr. Ramon Ravelo for helpful discussions and valuable comments.

REFERENCES

[1] Guerrero, Oscar, and Marcelo Marucho. *"Elastic-Plastic transition under uniaxial stress BCC tantalum."* Journal of Materials Science and Engineering. B 3.3B (2013): 153.

[2] O. Guerrero, *Pressure induced dynamical instabilities in body center cubic crystals*, Thesis, M.S., UTEP, 2010, p. 35.

[3] Duane C. Wallace, *Thermodynamics of Crystals*, Published January 29th 1998 by Dover Publications (first published April 19th 1972) ISBN: 0486402126 (ISBN13: 9780486402123)

[4] Dominik Lips, *Theoretical Methods for Determining Local Stress and Elastic Constants in Different Statistical Ensembles*, A thesis submitted in fulfillment of the requirements for the degree of Master of Science in Physics, University of Osnabrück, October 12, 2016

[5] John R. Ray, Michael C. Moody, and Aneesur Rahman, *Molecular dynamics calculation of elastic constants for a crystalline system in equilibrium*, Phys. Rev. B 32, 733 – Published 15 July 1985

[6] Krasil'nikov, O.M., Vekilov, Y.K. and Mosyagin, I.Y. *Elastic constants of solids at high pressures*. J. Exp. Theor. Phys. 115, 237-241 (2012). https://doi.org/10.1134/S1063776112070096

[7] Merodio, Jose and Hu, Eryi and Wang, Wenjin, *The Elastic Constants Measurement of Metal Alloy by Using Ultrasonic Nondestructive Method at Different Temperature*, Mathematical Problems in Engineering, 6762076, 2016

[8] Minta C. Akin, Jeffrey H. Nguyen, Martha A. Beckwith, Ricky Chau, W. Patrick Ambrose, Oleg V. Fat'yanov, Paul D. Asimow, and Neil C. Holmes , *"Tantalum sound velocity under shock compression"*, Journal of Applied Physics 125, 145903 (2019) https://doi.org/10.1063/1.5054332

[9] Decremps, F., Fischer, M., Polian, A. et al. *Elasticity of BaFCl single crystal under hydrostatic pressure*. Eur. Phys. J. B 5, 7–13 (1998). https://doi.org/10.1007/s100510050412

[10] Wolfram Research, Inc., System Modeler, Version 13.1, Champaign, IL (2022).

[11] Daniel Orlikowski, Per Söderlind, and John A. Moriarty, *First-principles thermoelasticity of transition metals at high pressure: Tantalum prototype in the quasiharmonic limit*, Phys. Rev. B 74, 054109 – Published 23 August 2006

[12] O. Gülseren and R. E. Cohen, *High-pressure thermoelasticity of body-centered-cubic tantalum*, Phys. Rev. B 65, 064103 – Published 3 January 2002

[13] S.M. Shapiro, E.C. Svensson, C. Vettier, B. Hennion, *Uniaxial-stress dependence of the phonon behavior in the premartensitic phase of Ni62.5Al37.5*, Phys. Rev. B 48 (1993) 13223.

[14] S.M. Shapiro, *Neutron scattering studies of pretransitional phenomena in structural phase transformations*, Metall.Trans. 12A(1990567), Mater. Sci. Forum 33 (1981) 56.

[15] A. Landa, J. Klepeis, P. Söderlind, I. Naumov, O. Velikokhatnyi, L. Vitos, et al., *Ab initio calculations of elastic constants of the bcc V-Nb system at high pressures*, J. Phys. Chem. Solids 67 (2006) 2056.

[16] A. Landa, P. Söderlind, A.V. Ruban, O.E. Peil, L. Vitos, *Stability in BCC transition metals: Madelung and band-energy effects due to alloying*, Phys. Rev. Lett. 103 (2009) 235501

[17] S.V. Makarov, V.A. Plotnikov, B.F. Dem'yanov, A.I. Potekaev. *WEAKLY STABLE STATE OF THE CRYSTAL LATTICE OF METALS AND ALLOYS*. Lett. Mater., 2019, 9(1) 27-32

[18] N. Amadou, T. de Resseguier, A. Dragon, *Influence of point defects and grain boundaries on plasticity and phase transition in uniaxially-compressed iron*, Computational Condensed Matter, Volume 27,2021,e00560,ISSN 2352-2143,https://doi.org/10.1016/j.cocom.2021.e00560.

[19] LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, S. J. Plimpton, Comp Phys Comm, 271 (2022) 10817

[20] John W. Eaton, David Bateman, Søren Hauberg, Rik Wehbring (2020). *GNU Octave version 6.1.0 manual: a high-level interactive language for numerical computations*. URL https://www.gnu.org/software/octave/doc/v6.3.0/

[21] R. Ravelo, Q. An, T. C. Germann and B. L. Holian, *Large-scale molecular dynamics simulations of shock induced plasticity in tantalum single crystals*,AIP Conference Proceedings 1426, 1263 (2012); https://doi.org/10.1063/1.3686510

[22] R. Ravelo, T. C. Germann, O. Guerrero, Q. An, and B. L. Holian, Phys. Rev. B 88, 134101 – Published 3 October 2013; Erratum Phys. Rev. B 89, 099902 (2014)

[23] Ankit Rohatgi, *WebPlotDigitizer*, Website: https://automeris.io/WebPlotDigitizer, Version: 4.5, Date: August, 2021

[24] Oscar Guerrero-Miramontes, *A beginner's guide to the modeling of shock/uniaxial/quasi- isentropic compression using the LAMMPS molecular dynamics simulator*, http://www.researchgate.net/publication/259644293 January 2013, (Access date: 11 Jul 2022).

[25] Y. Mishin, M.J. Mehl, D.A. Papaconstantopoulos, A.F. Voter, and J.D. Kress (2001), *"Structural stability and lattice defects in copper: Ab initio, tight-binding, and embedded-atom calculations"*, Physical Review B, 63(22), 224106. DOI: 10.1103/physrevb.63.224106.

[26] Giles Kimminau, Paul Erhart, Eduardo M. Bringa, Bruce Remington, and Justin S. Wark, *Phonon instabilities in uniaxially compressed fcc metals as seen in molecular dynamics simulations*, Phys. Rev. B 81, 092102 – Published 10 March 2010

[27] P. Contreras, *THE ROLE OF TRANSVERSE ANISOTROPIC ELASTIC WAVES AND EHREN- FEST RELATIONSHIPS IN THE SUPERCONDUCTING STATE OF THE COMPOUND Sr2RuO4*, Canadian Journal of Pure and Applied Sciences Vol. 14, No. 3, pp. 5061-5067, October 2020

[28] P. Contreras, *Visualización de la aproximación elipsoidal de frente de onda elástico en medios de simetría monoclínica*, DOI: 10.13140/RG.2.2.26997.81121

[29] E. B. Zaretsky and G. I. Kanel , *"Tantalum and vanadium response to shock-wave loading at normal and elevated temperatures. Non-monotonous decay of the elastic wave in vanadium"*, Journal of Applied Physics 115, 243502 (2014) https://doi.org/10.1063/1.4885047

[30] José M. Carcione, *Electromagnetic diffusion in anisotropic media*, Radio Science Volume46, Issue1, 2011, https://doi.org/10.1029/2010RS004402

[31] Saurav Goel, Ben Beakeb, Chi-Wai, Chana Nadimul, Haque Faisal, Nicholas Dunne, *Twinning anisotropy of tantalum during nanoindentation, Materials Science and Engineering*, A Volume 627, 11 March 2015, Pages 249-261

[32] Gupta, Mahesh Kumar; Mahapatra, Rajendra Prasad; Panwar, Vinay, *Effect of Temperature and Strain Rate Variation on Tensile Properties of a Defective Nanocrystalline Copper-Tantalum Alloy*, Journal of Scientific & Industrial Research Vol. 81, April 2022, pp. 358-366, DOI: 10.56042/jsir.v81i04.58725

[33] Landa, A.; Söderlind, P.; Naumov, I.I.; Klepeis, J.E.; Vitos, L. *Kohn Anomaly and Phase Stability in Group VB Transition Metals*. Computation 2018, 6, 29. https://doi.org/10.3390/computation6020029