computing elastic constants by DFT: stress tensor procedure

Stefaan Cottenier

November 2018

This document guides you through the procedure to compute the elastic constants of a given crystal by DFT, via the stress tensor procedure.

Whereas this document is written mainly in a code-independent way, the procedure is illustrated with DFT calculations for fcc-Al with Quantum Espresso, using the following settings: PBE exchange-correlation functional with the Al.pbe-n-kjpaw_psl.0.1.UPF pseudopotential for Al, ecutwfc=60 Ry, ecutrho=300 Ry, 20×20×20 k-mesh, conventional unit cell (4 atoms) with lattice parameter 4.037529Å (fully geometry-optimized for this XC-functional and pseudopotential).

1 step 1: standard root tensor of the crystal

At the start, we have the unit cell of a crystal. It is fully geometry-optimized, using well-converged DFT-calculations.

Choose an axis system, and put the unit cell in that axis system. Take one of the corners in the origin of the axis system. Often, yet not necessarily so, you take the axes aligned with the edges of the unit cell.

For all three lattice vectors that together define the unit cell: write down their components in this axis system. Apart from the general notation, an example for the conventional unit cell of fcc-aluminum is immediately added (a=4.037529 Å):

$$\vec{a} = (a_x, a_y, a_z) = (a, 0, 0) \quad (1)$$

$$\vec{b} = (b_x, b_y, b_z) = (0, a, 0) \quad (2)$$

$$\vec{c} = (c_x, c_y, c_z) = (0, 0, a) \quad (3)$$

Herewith, we form *the standard root tensor* of the initial cell:

$$R_1 = \begin{bmatrix} a_x & b_x & c_x \\ a_y & b_y & c_y \\ a_z & b_z & c_z \end{bmatrix} = \begin{bmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & a \end{bmatrix} \quad (4)$$

(For Quantum Espresso users: note that the standard root tensor is the transposed version of the `CELL_PARAMETERS` matrix in the input file of Quantum Espresso.)

2 step 2: deform the crystal

Now apply a deformation to this unit cell. A deformation is described by a *deformation gradient F*, which is a 3x3 matrix. In our example, we'll work with one specific choice of F, but in principle the matrix elements of F can have any value:

$$F = \begin{bmatrix} F_{xx} & F_{yx} & F_{zx} \\ F_{xy} & F_{yy} & F_{zy} \\ F_{xz} & F_{yz} & F_{zz} \end{bmatrix} = \begin{bmatrix} 1 & \delta & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5}$$

The way how F deforms the unit cell is found by multiplying the coordinate of any point in the undeformed cell with F. The result is the coordinate of that point after the deformation:

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} F_{xx} & F_{yx} & F_{zx} \\ F_{xy} & F_{yy} & F_{zy} \\ F_{xz} & F_{yz} & F_{zz} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \tag{6}$$

(coordinates before the deformation are labeled by '1', after the deformation by '2') In particular, you can apply F to the components of a lattice vector (to be considered as the coordinates of the end point of that vector). It returns a lattice vector of the deformed cell:

$$\begin{bmatrix} a_{2x} \\ a_{2y} \\ a_{2z} \end{bmatrix} = \begin{bmatrix} F_{xx} & F_{yx} & F_{zx} \\ F_{xy} & F_{yy} & F_{zy} \\ F_{xz} & F_{yz} & F_{zz} \end{bmatrix} \begin{bmatrix} a_{1x} \\ a_{1y} \\ a_{1z} \end{bmatrix} \tag{7}$$

This can be written for all three lattice vectors simultaneously: multiply the deformation gradient with the standard root tensor of the undeformed cell, to get the standard root tensor of the deformed cell:

$$\begin{bmatrix} a_{2x} & b_{2x} & c_{2x} \\ a_{2y} & b_{2y} & c_{2y} \\ a_{2z} & b_{2z} & c_{2z} \end{bmatrix} = \begin{bmatrix} F_{xx} & F_{yx} & F_{zx} \\ F_{xy} & F_{yy} & F_{zy} \\ F_{xz} & F_{yz} & F_{zz} \end{bmatrix} \begin{bmatrix} a_{1x} & b_{1x} & c_{1x} \\ a_{1y} & b_{1y} & c_{1y} \\ a_{1z} & b_{1z} & c_{1z} \end{bmatrix} \tag{8}$$

For the example deformation of fcc aluminum, this becomes:

$$\begin{bmatrix} a & a\delta & 0 \\ 0 & a & 0 \\ 0 & 0 & a \end{bmatrix} = \begin{bmatrix} 1 & \delta & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & a \end{bmatrix} \tag{9}$$

3 step 3: construct DFT-input for the deformed crystal

Create a cif file for the deformed cell (unless your DFT code allows a handy input for deformed cells, for instance because the standard root tensor or the lattice vector matrix is part of the input file – in that case you can enter the deformed cell there, without the need to go via the cif file), and convert it to input for your DFT code. The cell edges and cell angles of the deformed cell (which you can put directly in a P1-cif) are found in this way from the information in the standard root tensor (apply cyclic permutations for the other cell vectors and angles):

$$a = \sqrt{a_x^2 + a_y^2 + a_z^2} \tag{10}$$

$$\alpha = \arccos\left(\frac{b_x c_x + b_y c_y + b_z c_z}{bc}\right) \tag{11}$$

This results in these values for the deformed aluminum cell (with $\delta = 0.03$):

$$\begin{array}{rcll} \mathrm{a}_2 & = & \mathrm{a} & = & 4.037529 \ \text{Å} \\ \mathrm{b}_2 & = & \mathrm{a}\sqrt{1+\delta^2} & = & 4.039345 \ \text{Å} \\ \mathrm{c}_2 & = & \mathrm{a} & = & 4.037529 \ \text{Å} \\ \alpha & = & 90^\circ & & \\ \beta & = & 90^\circ & & \\ \gamma & = & 88.28087^\circ & & \end{array} \tag{12}$$

4 step 4: compute the stress tensor for the deformed crystal

Do a static DFT calculation for your deformed cell, and inspect whether or not there are forces on any of the atoms. If there are internal degrees of freedom, **then optimize the positions of the atoms** (only the positions, do not change the volume or shape of the deformed cell).

Once you have optimized the positions, determine the stress tensor for this deformed cell. The deformed fcc-Al example we are examining here has no forces on the atoms, hence we can compute the stress tensor right away and read it from the output of the DFT code. Most DFT codes report the stress tensor as the *external stress*: the stress that has to be applied to keep the crystal in the given shape (defined as positive when pointing inward). What we will need in the stress-strain relation we are building is, however, the *internal stress*: the stress by which the material keeps itself in the given shape (defined as positive when pointing outward). Both have opposite signs[1]. Therefore, if your DFT

[1] Compare this to the force of a mass on an elongated spring: either you consider the force you need to exert on the mass to keep that elongation, or you consider the force by which the spring pulls back on the mass. Both forces are identical in magnitude, yet have opposite signs.

code reports the external stress (as most do), **you will need to invert the signs of the stress tensor components in order to construct the tensor** Σ (units: GPa):

$$\Sigma = \begin{bmatrix} -0.166 & 0.937 & 0.000 \\ 0.937 & -0.203 & 0.000 \\ 0.000 & 0.000 & 0.011 \end{bmatrix} \longrightarrow \begin{bmatrix} -0.166 \\ -0.203 \\ 0.011 \\ 0.000 \\ 0.000 \\ 0.937 \end{bmatrix} \tag{13}$$

The column notation at the right hand side is Voigt notation, which we will need later on. It is only a different way of writing the 6 independent elements that are present in this symmetric 3x3 matrix. The conversion between matrix and Voigt notation for the stress tensor is:

$$\begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{bmatrix} \longrightarrow \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{xz} \\ \sigma_{xy} \end{bmatrix} \tag{14}$$

In contrast to the notation conversion for the strain tensor that we will meet soon, there are **no** factors of 2 involved in this conversion.

5 step 5: determine the strain tensor for the deformed crystal

Now compute the Green-Lagrange strain tensor for this deformation. This does not require DFT, and can be done in two ways :

5.1 strain tensor via the Bilbao server

Fill out the values for the undeformed and deformed cell edges and angles in the strain tool of the Bilbao server at http://cryst.ehu.es/cryst/strain.html . For our example, these are the two input strings needed for the undeformed and the deformed cell:

```
4.037529 4.037529 4.037529 90 90 90
4.037529 4.039345 4.037529 90 90 88.28087
```

The output you have to look at is the "Finite Lagrangian Strain Tensor (finite deformation)":

$$E = \begin{bmatrix} 0.000000 & 0.015007 & 0.000000 \\ 0.015007 & 0.000450 & 0.000000 \\ 0.000000 & 0.000000 & 0.000000 \end{bmatrix} \tag{15}$$

The Bilbao server gives this tensor as a symmetric 3x3 matrix. We will need it in Voigt notation (column with 6 elements) rather than in 3x3 matrix notation (there is no information loss, because the Green-Lagrange strain tensor is always symmetric). The translation from one to the other notation is:

$$\begin{bmatrix} \epsilon_{xx} & \epsilon_{xy} & \epsilon_{xz} \\ \epsilon_{xy} & \epsilon_{yy} & \epsilon_{yz} \\ \epsilon_{xz} & \epsilon_{yz} & \epsilon_{zz} \end{bmatrix} \longrightarrow \begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ 2\epsilon_{yz} \\ 2\epsilon_{xz} \\ 2\epsilon_{xy} \end{bmatrix} \tag{16}$$

(Mind the factors 2! These are not there in the Voigt notation of the stress tensor.) For the aluminum example, the output of the Bilbao server in Voigt notation should therefore be written as:

$$E = \begin{bmatrix} 0.000000 \\ 0.000450 \\ 0.000000 \\ 0.000000 \\ 0.000000 \\ 0.030014 \end{bmatrix} \tag{17}$$

5.2 strain tensor via its definition

If you don't have access to the Bilbao server strain tensor tool, you can create the strain tensor manually via the definition of the Green-Lagrange strain tensor:

$$E = \frac{1}{2}\left(F^T F - I\right) \tag{18}$$

(T stands for transposed, I is the unit matrix) For our example, this gives:

$$E = \begin{bmatrix} 0 & \frac{\delta}{2} & 0 \\ \frac{\delta}{2} & \frac{\delta^2}{2} & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{19}$$

We will need this strain tensor in Voigt notation (column with 6 elements) rather than in 3x3 matrix notation (there is no information loss, because the Green-Lagrange strain tensor is always symmetric). The translation from one to the other notation is the same as given above, and is repeated here:

$$\begin{bmatrix} \epsilon_{xx} & \epsilon_{xy} & \epsilon_{xz} \\ \epsilon_{xy} & \epsilon_{yy} & \epsilon_{yz} \\ \epsilon_{xz} & \epsilon_{yz} & \epsilon_{zz} \end{bmatrix} \longrightarrow \begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ 2\epsilon_{yz} \\ 2\epsilon_{xz} \\ 2\epsilon_{xy} \end{bmatrix} \tag{20}$$

(Mind the factors 2! These are not there in the Voigt notation of the stress tensor.) For our deformed fcc-Al example, this becomes:

$$E = \begin{bmatrix} 0 \\ \frac{\delta^2}{2} \\ 0 \\ 0 \\ 0 \\ \delta \end{bmatrix} \tag{21}$$

Verify that this is identical to what the Bilbao server gave (for $\delta = 0.03$).

The strain tool of the Bilbao server is most useful to determine the strain tensor in a case where you know the volume and shape of the undeformed and deformed unit cells. In the case of the aluminum example, however, we do already know the deformation gradients, because we imposed them. This will always be the case when you're goal is to calculate elastic constants via the stress tensor procedure. Therefore, it is straightforward to write down the strain tensor directly via the definition, without need for the Bilbao server.

6 step 6: stress-strain relation

The stiffness tensor relates the strain components with the stress components via the elastic constants C_{ij} in the following way (mind the factors 2) :

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{xz} \\ \sigma_{xy} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{12} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{13} & C_{23} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{14} & C_{24} & C_{34} & C_{44} & C_{45} & C_{46} \\ C_{15} & C_{25} & C_{35} & C_{45} & C_{55} & C_{56} \\ C_{16} & C_{26} & C_{36} & C_{46} & C_{56} & C_{66} \end{bmatrix} \begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ 2\epsilon_{yz} \\ 2\epsilon_{xz} \\ 2\epsilon_{xy} \end{bmatrix} \tag{22}$$

As we know for the aluminum example both the strain components ϵ and the stress components σ, the above expression gives a set of equations with the elastic constants C_{ij} as the unknowns. In the next step, we will collect even more of these equations, until we have a system of equations that is sufficiently large to find all C_{ij}.

7 step 7: repeat from step 2 (6x)

Repeat the preceding procedure starting from step 2, with other deformation gradients that are independent from each other, until you have it done for 6 deformations. Any set of 6 deformations is a good one, as long as they are independent. However, the following choice of six qualitatively different deformations is convenient and recommended:

$$F_1 = \begin{bmatrix} 1+\delta & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad F_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1+\delta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad F_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1+\delta \end{bmatrix}$$

$$F_4 = \begin{bmatrix} 1 & \delta & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad F_5 = \begin{bmatrix} 1 & 0 & \delta \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad F_6 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \delta \\ 0 & 0 & 1 \end{bmatrix} \tag{23}$$

For F_1 to F_3 (tensile deformations), $\delta = \delta_1 = 0.01$ is a good standard choice. For F_4 to F_6 (shear deformations), take $\delta = \delta_2 = 0.03$.

8 step 8: solve the system of equations

In order to obtain the elastic constants, write the stress-strain relations (Eq. 22) for each of these 6 deformations, and put them in one matrix equation:

$$\begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} & \sigma_{14} & \sigma_{15} & \sigma_{16} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} & \sigma_{24} & \sigma_{25} & \sigma_{26} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} & \sigma_{34} & \sigma_{35} & \sigma_{36} \\ \sigma_{41} & \sigma_{42} & \sigma_{43} & \sigma_{44} & \sigma_{45} & \sigma_{46} \\ \sigma_{51} & \sigma_{52} & \sigma_{53} & \sigma_{54} & \sigma_{55} & \sigma_{56} \\ \sigma_{61} & \sigma_{62} & \sigma_{63} & \sigma_{64} & \sigma_{65} & \sigma_{66} \end{bmatrix} = \tag{24}$$

$$\begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{12} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{13} & C_{23} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{14} & C_{24} & C_{34} & C_{44} & C_{45} & C_{46} \\ C_{15} & C_{25} & C_{35} & C_{45} & C_{55} & C_{56} \\ C_{16} & C_{26} & C_{36} & C_{46} & C_{56} & C_{66} \end{bmatrix} \begin{bmatrix} \epsilon_{11} & \epsilon_{12} & \epsilon_{13} & \epsilon_{14} & \epsilon_{15} & \epsilon_{16} \\ \epsilon_{21} & \epsilon_{22} & \epsilon_{23} & \epsilon_{24} & \epsilon_{25} & \epsilon_{26} \\ \epsilon_{31} & \epsilon_{32} & \epsilon_{33} & \epsilon_{34} & \epsilon_{35} & \epsilon_{36} \\ 2\epsilon_{41} & 2\epsilon_{42} & 2\epsilon_4 3 & 2\epsilon_{44} & 2\epsilon_{45} & 2\epsilon_{46} \\ 2\epsilon_{51} & 2\epsilon_{52} & 2\epsilon_{53} & 2\epsilon_{54} & 2\epsilon_{55} & 2\epsilon_{56} \\ 2\epsilon_{61} & 2\epsilon_{62} & 2\epsilon_{63} & 2\epsilon_{64} & 2\epsilon_{65} & 2\epsilon_{66} \end{bmatrix}$$

Mind the indices in this equation. The elastic constants are repeated from Eq. 22, with indices that emphasize this is a symmetric matrix. In σ_{ij} or ϵ_{ij}, double digits are now used as indices, in contrast to the double characters we used up to now. The column index j refers to the number of the deformation (first deformation, second deformation,...), while the row index i refers to the position in the Voigt (or column) notation: σ_{24} is the yy-component of the stress tensor for the 4^{th} deformation.

For the fcc-Al example, this is the resulting matrix with all six stress tensors (you'll recognize the 4^{th} column, which is the stress tensor for the 4^{th} deformation, as we found before):

$$\Sigma = \begin{bmatrix} 1.120 & 0.502 & 0.502 & -0.166 & -0.166 & 0.011 \\ 0.502 & 1.120 & 0.502 & -0.203 & 0.011 & -0.166 \\ 0.502 & 0.502 & 1.120 & 0.011 & -0.203 & -0.203 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.937 \\ 0.000 & 0.000 & 0.000 & 0.000 & 0.937 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.937 & 0.000 & 0.000 \end{bmatrix} \tag{25}$$

In the matrix with all strain tensors, you'll recognize the 4$^{\text{th}}$ column too:

$$E = \begin{bmatrix} \delta_1 + \frac{\delta_1^2}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \delta_1 + \frac{\delta_1^2}{2} & 0 & \frac{\delta_2^2}{2} & 0 & 0 \\ 0 & 0 & \delta_1 + \frac{\delta_1^2}{2} & 0 & \frac{\delta_2^2}{2} & \frac{\delta_2^2}{2} \\ 0 & 0 & 0 & 0 & 0 & \delta_2 \\ 0 & 0 & 0 & 0 & \delta_2 & 0 \\ 0 & 0 & 0 & \delta_2 & 0 & 0 \end{bmatrix} \tag{26}$$

The last step is now to search the inverse of the E-matrix (see Sec. 9, either symbolically or numerically), right-multiply both sides of Eq. 24 by this inverse, work out the multiplication of the Σ-matrix with this inverse (see Sec. 9), and eventually you can read all C-values from this product.

As this procedure is using always the same 6 deformations, the inverse of the E-matrix can be computed symbolically once and for all:

$$E^{-1} = \begin{bmatrix} \frac{1}{\delta_1} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{\delta_1} & 0 & 0 & 0 & \frac{-\frac{\delta_2}{2}}{\delta_1+\frac{\delta_1^2}{2}} \\ 0 & 0 & \frac{1}{\delta_1} & \frac{-\frac{\delta_2}{2}}{\delta_1+\frac{\delta_1^2}{2}} & \frac{-\frac{\delta_2}{2}}{\delta_1+\frac{\delta_1^2}{2}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\delta_2} \\ 0 & 0 & 0 & 0 & \frac{1}{\delta_2} & 0 \\ 0 & 0 & 0 & \frac{1}{\delta_2} & 0 & 0 \end{bmatrix} \tag{27}$$

Filling out the values δ_1=0.01 and δ_2=0.03 we used here, yields:

$$E^{-1} = \begin{bmatrix} 99.50249 & 0.00000 & 0.00000 & 0.00000 & 0.00000 & 0.00000 \\ 0.00000 & 99.50249 & 0.00000 & 0.00000 & 0.00000 & -1.49254 \\ 0.00000 & 0.00000 & 99.50249 & -1.49254 & -1.49254 & 0.00000 \\ 0.00000 & 0.00000 & 0.00000 & 0.00000 & 0.00000 & 33.33333 \\ 0.00000 & 0.00000 & 0.00000 & 0.00000 & 33.33333 & 0.00000 \\ 0.00000 & 0.00000 & 0.00000 & 33.33333 & 0.00000 & 0.00000 \end{bmatrix} \tag{28}$$

The product of Σ (Eq. 25) and E^{-1} (Eq. 28) gives all elastic constants for fcc-aluminum (in GPa):

$$\Sigma E^{-1} = C = \begin{bmatrix} 111.4 & 50.0 & 50.0 & -0.4 & -6.3 & -6.3 \\ 50.0 & 111.4 & 50.0 & -6.3 & -0.4 & -8.4 \\ 50.0 & 50.0 & 111.4 & -8.4 & -8.4 & -0.4 \\ 0.0 & 0.0 & 0.0 & 31.2 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 31.2 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 31.2 \end{bmatrix} \tag{29}$$

This result must be a symmetric 6x6 matrix. The fact that it isn't exactly symmetric (the upper right 3×3 block) is due to numerical noise: more converged

digits in the stresses, more digits used in the matrix elements,... would help. To fight against such numerical noise, multiple choices for δ_1 and δ_2 can be made (in particular also choices with the opposite sign: $\delta_1 = \pm 0.01$, $\delta_2 = \pm 0.03$), after which the resulting stiffness tensors (C-matrices) are averaged.

In any case, the result in Eq. 29 suggests that there are three non-zero elastic constants:

$$\begin{aligned} C_{11} &= 111 \text{ GPa } (= C_{22} = C_{33}) \\ C_{12} &= 50 \text{ GPa } (= C_{23} = C_{13}) \\ C_{44} &= 31 \text{ GPa } (= C_{55} = C_{66}) \end{aligned} \tag{30}$$

The numerical uncertainty is of the order of magnitude of the largest values that should have been zero by symmetry: 5-10 GPa. We can compare these results with the DFT predictions for fcc-Al in Materials Project, which are at the same PBE-level of theory: C_{11}=104 GPa, C_{12}=73 GPa, C_{44}=32 GPa (https://materialsproject.org/materials/mp-134). Except perhaps for C_{12}, this is fair agreement.

The procedure as described here, is valid for any crystal, regardless how (un)symmetric it is. However, as the stress tensor values for the fcc-Al example suggests, you could use less than 6 deformations if you consider symmetry. For fcc-Al, one tensile and one shear deformation would have been sufficient, when complemented with the proper symmetry considerations. If you don't want to care about the symmetry of your crystal, then just do all six deformations.

The same elastic constants can be calculated using the total energy procedure as well, and compared to the values found here.

9 resources and tools

In the spirit of this course, every work flow should be possible using free online tools. Hence, here are some resources to help with the matrix math:

Matrix inversion online :
http://matrix.reshish.com/inverse.php
https://www.symbolab.com/solver/matrix-inverse-calculator
http://onlinemschool.com/math/assistance/matrix/inverse/

Matrix multiplication online :
http://matrix.reshish.com/multiplication.php
http://www.bluebit.gr/matrix-calculator/matrix$_m$*ultiplication.aspx*
http : *//onlinemschool.com/math/assistance/matrix/multiply/*

If you want to have more details about the formalism behind the procedure discussed in this document, then you might find this open online course on continuum mechanics useful: http://www.continuummechanics.org . Concise, yet very clear. In particular the sections about *Deformation Gradients* (IV.A),

Polar Decomposition (IV.B) and *Green & Almansi Strains* (IV.F) play a role here.

Another description of the underlying theory, more focussing on the DFT aspect, is available at https://www.materialsproject.org/docs/calculationselasticity. It is basically a repetition of the information in http://dx.doi.org/10.1038/sdata.2015.9 (de Jong et al. 2015). The original literature on this method is available at http://dx.doi.org/10.1103/PhysRevB.65.104104 (Le Page and Saxe, 2002).