ELASTIC3rd Documentation

Release 1.0

ELASTIC3rd Developers

Oct 18, 2020

Contents

CHAPTER 1

Introduction

Elastic3rd is an open source code to calculate the higher order elastic constants(HOECs) of materials from first principles.

1.1 Features

- Support strain-energy method for calculating TOECs, FOECs for all symmetry and FFOECs for cubic
- Support post processing by considering higher order effect
- Support CASTEP(in Materials Studio, both under Windows and Linux) and VASP
- Support python 2.7.x and pyython 3.5 +

1.2 Features under developing

- Strain-stress method
- Separate elastic constants method (ref. PRL. 121, 216001, 2018)

CHAPTER 2

Installation

2.1 Requirements

python, numpy, scipy, matplotlib

2.2 Installation

- by pip

  `pip install elastic3rd`

- by *setup.py*

  Download the code and entre the folder *ELASTIC3rd/src*, and then run `python setup.py install`

CHAPTER 3

Get Started

3.1 Simplest way to run

- Prepare the *INPUT* file and the input file for first principlec code
- Enter the folder
- Simply run `elastic3rd run`

3.2 Inputs

The inputs for Elastic3rd can be divided into two parts. The first one is the INPUT file, and the second is the input files for first princples code.

3.2.1 INPUT

Parameters	Default	Comments
BaseName	Si	**The folder name of the result. In addition,** For CASTEP, it must equals to the name of *cell* and *param* file For VASP, no above required
CrystalType	Cubic1	**The symmetry of the crystal. It should be one of the** triclinic or n, monoclinic or m, orthorhombic or o, tetragonal1 or t1 tetragonal2 or t2, rhombohedral1 or r1, rhombohedral2 or r2 hexagonal1 or h1, hexagonal2 or h2, cubic1 or c1, cubic2 or c2
Ord	3	The order of elastic constants
EnergyCode	CASTEP	The first principles code, currently support CASTEP and VASP
EnergyRun	1	**The way to run the first principles code** **1 is the default way. For CASTEP: RunCASTE** For VASP: mpirun -np NP vasp_std **0 means user defined. It will read the *energyru*** And it can be overwrite by elastic3rd.energy.glue.write_energyrun(RunStr) function
MaxStrain	5	The maximum strain, in unit of %. e.g. 5 means the strain range is -5% to 5%
STEPS	5	The number of points in each strain mode
NP	1	The number of cores used in the first principles codes

3.2.2 First principles inputs

- CASTEP For CASTEP, two files are required for calculating, *BaseName.cell* and *BaseName.param*. For details, ref. CASTEP files .
- VASP To run VASP, four files are required: *INCAR*, *POSCAR*, *KPOINTS* and *POTCAR*. For details, ref. VASP files .

3.3 Calculation

3.3.1 Using CASTEP or VASP

Currently, Elastic3rd support CASTEP and VASP to calculate the energy or stress. Hence, the user can call vasp or castep by setting the INPUT file.

Note: The CASTEP is tested using the CASTEP module in Materials Studio. By default, the user should provide RunCASTEP.bat (windows) or RunCASTEP.sh (linux) in current folder. Usually, this file loacted in *Materials_Studio_install_pathetcCASTEPbin*

For VASP, it runs vasp by *mpirun -np NP vasp_std*. So, please ensure that mpirun and vasp_std can be found in yout PATH.

3.3.2 Using other first principles code

If the user want to use other first principles code to calculate the energy of stress, the user only need to prepare the following thing:

- write a python file named with the code's name(e.g. qe.py, lower required) under elastic3rd/energy/

  In this file, the user need to write the following functions

  - get_base_vec(BaseName): this function get the lattice vector from the code's input file
  - write_base_vec(BaseName, BaseVec): this function write out the lattice vector to the code's input file
  - run(NP, BaseName): this function return the string to run the code
  - get_energy(BaseName): this function get the energy from the code's output file
  - copy_files(BaseName, Path): this function copy the required files of the code
- Then in the INPUT file, the file can call the code by specify "EnergyCode QE"

3.4 Post

The SOECs and TOECs (only taking second and third order effect into consideration) are printed in the log file on the screen.

If the author want to taken higher-order effect into consideration, the user can run *elastic3rd post* command in the folder where the job submitted or use *Elastic3rdPost.py* script in the *example* folder

For *elastic3rd post* command, please use *-h* for detail help.

For test, the user can run *elastic3rd post* in the *example/Si/CASTEP* folder directly.

3.5 Outputs

3.5.1 The structure of outputs

All the output are stored in the folder named BaseName, of which the value is set in INPUT file. The following is the structure of the results.

```
Si
├─INPUT
├─EEnergy.txt
├─Mode0
├─Mode1
│  ├─Energy_Mode.txt
│  ├─Strain-2.0
│  │  ├─Energy_Strain.txt
│  │  └─...
│  ├─Strain-4.0
│  └─...
├─Mode2
│  └─...
└─...
```

3.5.2 Output files

This section gives an explanation for each file generated by Elastic3rd.

File name	Comments
INPUT	A backup of INPUT file
EEnergy.txt	The energy of each strain mode(each column) and each strain(each row)
Energy_Strain.txt	The energy of single strain, in which there are several values in different unit
Energy_Mode.txt	The eneargy of single strain mode. In unit of eV

Folders	Comments
Mode0	This folder contains the result of undeformed structure
ModeN	This folder contains the result of differen strain modes
StrainN	This folder contains the result of different strains

Note: here N is integer and N>0

3.5.3 Log file

By default, the log file is not saved, but print in the screen. And it can be saved by re-direction.

It can be divided into five parts

- Part1: the logo part

```
+============================================================+
| EEEEE L       A   SSSSS TTTTT IIIII CCCCC 33333 RRRRR DDD   |
| E     L      A A  S       T     I   C         3 R   R D  D  |
| EEEE  L     A   A SSSSS   T     I   C      3333 RRRRR D   D |
| E     L     AAAAA     S   T     I   C         3 R  R  D  D  |
| EEEEE LLLLL A   A SSSSS   T   IIIII CCCCC 33333 R   R DDD   |
+------------------------------------------------------------+
|             Version: ELASTIC3PY  version 2.0               |
|                Date: 2019-01-27                            |
|              Author: Liao Mingqing                         |
```

(continues on next page)

(continued from previous page)

```
|              E_mail: liaomq1900127@163.com                    |
|     FGMS @ Harbin Institute of Technology(HIT)                |
|    URL:https://github.com/hitliaomq/ELASTIC3rd                |
|Please Cite:                                                   |
| Mingqing Liao,ELASTIC3rd,(2018).doi:10.5281/zenodo.2525580    |
+===============================================================+
```

- Part2: The summary of the calculation

```
==================The input parameters==================
The Runing mode is : NEW
The BaseName is : Si
The crystal symmetry is : c1
The order of elastic is : 3.0
The code for calculating the energy : CASTEP
The maximum of the strain : 8.0
The steps run in every mode : 9.0
The core used in energy calculate : 8.0
```

- Part3: the result of intial structure

```
====================Crystal Structure====================
BaseVec:
     5.4647128853     0.0000000210     -0.0000000454
     -0.0000000000     5.4647128382     -0.0000000222
     0.0000000000     0.0000000000     5.4647129020
Lattice Parameter:
    a         b         c      alpha   beta    gamma
5.464713  5.464713  5.464713  90.000  90.000  90.000
Volume : 163.193194972

Energy of undeformed structure:
       eV             Hartree            kJ/mol            kcal/mol
-858.6574758785  -31.5550611783  -82847.8330874176  -19801.1085892908
```

- Part4: the result of different strains and strain modes

```
==================Deformed Crystal=======================
----------------------------------------------------------
Start calculating Mode 1
Start calculating Strain -8.0 in Mode 1
[[0.84 0.   0.  ]
 [0.   1.   0.  ]
 [0.   0.   1.  ]]
BaseVec:
     5.0084920896     0.0000000210     -0.0000000454
     0.0000000000     5.4647128382     -0.0000000222
     0.0000000000     0.0000000000     5.4647129020
Lattice Parameter:
    a         b         c      alpha   beta    gamma
5.008492  5.464713  5.464713  90.000  90.000  90.000
Volume : 149.569033772

Energy:
       eV             Hartree            kJ/mol            kcal/mol
-858.0935122952  -31.5343359114  -82793.4188860286  -19788.1032822074
End of Strain -8.0 in Mode 1
```

(continues on next page)

(continued from previous page)

```
Start calculating Strain -6.0 in Mode 1
[[0.88 0.   0.  ]
 [0.   1.   0.  ]
 [0.   0.   1.  ]]
BaseVec:
     5.1263550880     0.0000000210     -0.0000000454
     0.0000000000     5.4647128382     -0.0000000222
     0.0000000000     0.0000000000     5.4647129020
Lattice Parameter:
    a         b         c      alpha   beta    gamma
5.126355  5.464713  5.464713  90.000  90.000  90.000
Volume : 153.088786718

Energy:
       eV             Hartree            kJ/mol            kcal/mol
-858.3477769585  -31.5436799598  -82817.9516909908  -19793.9667636885
End of Strain -6.0 in Mode 1

                       ...
```

- Part5: the post process part

```
==================Post Processing=========================
[[   77.22101104  -117.3314942 ]
 [  217.27559666  -670.1092832 ]
 [  218.38708611    26.16025079]
 [  227.66199634  -632.04346098]
 [  424.89246809 -1747.12822498]
 [  439.3186686   -408.3765645 ]]
[11 12 44]
[153.05111127  65.0171889   73.22518077]
[111 112 123 144 155 456]
[-703.9889652  -435.4462948   -88.79485798   71.7458725  -257.35598339
  -51.04707056]
========================!!!END!!!=========================
```

CHAPTER 4

Theory Background

The strain-energy method to calculate TOECs is based on the conituumn elasticity theory. The base equation is as follows (note: the Einstein summation notation is used in the following equations):

$$E = E_0 + \frac{1}{2!}V_0 C_{ijkl}\eta_{ij}\eta_{kl} + \frac{1}{3!}V_0 C_{ijklmn}\eta_{ij}\eta_{kl}\eta_{mn} + ...$$

Where

- E is the energy of the deformed structure
- E_0 is the energy of the initial structure
- V_0 is the volume of the initial structure
- C_{ijkl} is SOECs
- C_{ijklmn} is TOECs
- η_{ij} is the Lagrangian strain

And the Lagrangian strain can be written as follows:

$$\eta_{ij} = \frac{1}{2}(F_{ki}F_{kj} - \delta_{ij})$$

The F is the deformation gradient, and can be expressed by the lattic vector of deformed structure(r') and intial structure (r).

$$r^{'} = Fr$$

when using symmetrical strain, the deformation gradient can be expressed by the Lagrangian strain.

$$\begin{aligned} F &= Q\sqrt{\lambda}Q^T \\ Y &= 2\eta + I \\ Q &= (y_1, y_2, y_3) \\ \lambda &= diag(\lambda_1, \lambda_2, \lambda_3) \end{aligned}$$

where y_i and λ_i are the eigenvector and eigenvalue of $\mathbf{Y}$, I is the identity matrix.

For each strain mode, we only adjust the amplitude. Hence, we can express the strain as a function of strain amplitude η, then we have

$$\frac{E - E_0}{V_0} = \frac{1}{2}A_2\eta^2 + \frac{1}{6}A_3\eta^3 + O(\eta^4)$$

Finally, we can choose different strain modes, then get different equations about TOECs and SOECs. By solving the equations, we can get the value of SOECs and TOECs.

CHAPTER 5

Examples

5.1 Help on the *elastic3rd* command

There are two command to submit jobs or do post process.

- *elastic3rd run* command

```
ELASTIC3RD version: 2.5.1
Copyright © FGMS @ HIT

usage: elastic3rd run [-h] [-i INPUT]

optional arguments:
  -h, --help            show this help message and exit
  -i INPUT, --input INPUT
                        The input file for Elastic3rd. Default: 'INPUT'
```

- *elastic3rd post* command

```
ELASTIC3RD version: 2.5.1
Copyright © FGMS @ HIT

usage: elastic3rd post [-h] [-n N] [-i INPUT] [-sm STRAINMODE] [-fig FIG] [-v VOLUME]␣
↪[-e ENERGY]

optional arguments:
  -h, --help            show this help message and exit
  -n N, --n N           The order of elastic constant taken into consideration in the␣
↪post process. Default: 3
  -i INPUT, --input INPUT
                        The input file for Elastic3rd. Default: 'INPUT'
  -sm STRAINMODE, --strainmode STRAINMODE
                        The strainmode file for Elastic3rd, None means using the code␣
↪to generate it. Default: None
```

(continues on next page)

(continued from previous page)

```
  -fig FIG, --fig FIG   Show the fitting fig. Default: 0
  -v VOLUME, --volume VOLUME
                        The volume of initial structure. Default: None
  -e ENERGY, --energy ENERGY
                        The file of the energy result. If None, it will try to find␣
↪in current folder or in the
                        BaseName folder. Default: None
```

5.2 Submit jobs

There are several examples in *example* folder.

For Si and C, both VASP and CASTEP input files are prepared in the folder, and for Mg, only CASTEP input file is prepared.

The user can run these examples by following steps

1. For copy right, the author can't provide the RunCASTEP.bat or RunCASTEP.sh (for CASTEP) and POTCAR (for VASP) files. The user should provide these file in the current folder.
   - For CASTEP, the user should copy *RunCASTEP.bat* or *RunCASTEP.sh* in their Materils Studio installation folder.
   - For VASP, the user should provide the *POTCAR* file
2. Submit the job

There are several ways to submit the job.

- Using *elastic3rd run* command.
- Using the *runElastic3rd.py* (in *example* folder) script by running *python runElastic3rd.py*. If the user want to save the log into file, please run *python runElastic3rd.py >> Result.txt*
- Submit jobs in queue system. If the user want to run Elastic3rd in queue system, the following is an example. Please note that it assume the python and vasp is exists in the *PATH*

```
#PBS -l nodes=1:ppn=24
#PBS -l walltime=48:00:00
#PBS -l pmem=8gb
#PBS -A open
#PBS -j oe

cd $PBS_O_WORKDIR

python runElastic3rd.py
```

5.3 Post

There are several ways to do the post processing.

- Using *elastic3rd post* command

Assumpt that the current folder is the folder where the job has been submitted. When enter the current folder, the user can do the post process by *elastic3rd post* to get the SOECs and TOECs. For more options, please run *elastic3rd post -h*.

For test, the user can run elastic3rd post in the *example/Si/CASTEP* folder directly. It should print the following lines in the screen.

```
ELASTIC3RD version: 2.5.1
Copyright © FGMS @ HIT

BaseVec:
     5.4647128853     0.0000000210     -0.0000000454
     -0.0000000000     5.4647128382     -0.0000000222
     0.0000000000     0.0000000000     5.4647129020
Lattice Parameter:
    a         b         c      alpha   beta    gamma
5.464713  5.464713  5.464713  90.000  90.000  90.000
Volume : 163.1931949715094

[11 12 44]
[153.05111127  65.0171889   73.22518077]
[111 112 123 144 155 456]
[-703.9889652  -435.4462948   -88.79485798   71.7458725  -257.35598339
  -51.04707056]
```

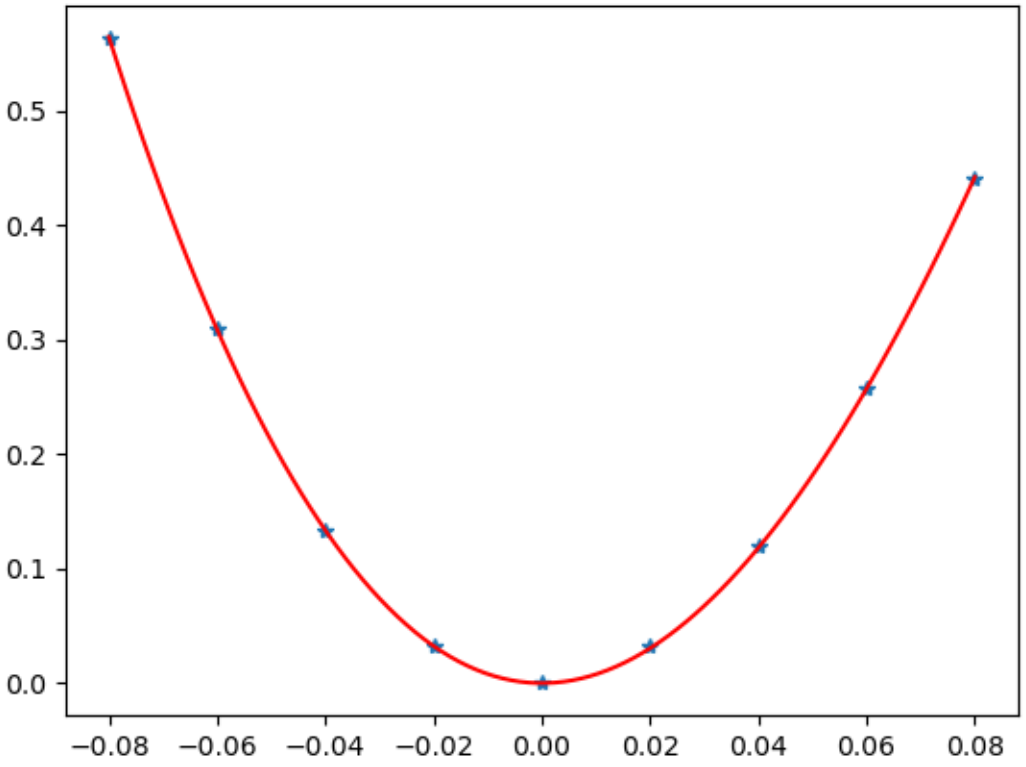
- Using *Elastic3rdPost.py* script

Change some parameters (e.g. *Flag_Ord*) by the user, then run *python Elastic3rdPost.py*

By running this script, it will print the following thing in the screen and show some figures of fitting result (here only show 1 figure)

```
BaseVec:
     5.4647128853     0.0000000210     -0.0000000454
     -0.0000000000     5.4647128382     -0.0000000222
     0.0000000000     0.0000000000     5.4647129020
Lattice Parameter:
    a         b         c      alpha   beta    gamma
5.464713  5.464713  5.464713  90.000  90.000  90.000
Volume : 163.1931949715094

[  77.22101104 -117.33149423]
[ 217.27559666 -670.10928332]
[218.3870861   26.16025386]
[ 227.66199634 -632.04346107]
[  424.89246807 -1747.1282253 ]
[ 439.3186686  -408.37656319]
[153.05111127  65.01718889  73.22518077]
[-703.98896537 -435.44629486  -88.79485803   71.74587404 -257.35598342
  -51.0470704 ]
```



- By calling the *post* or *post_single* or *post_mode* from *elastic3rd.post.post*

Ref *Elastic3rdPost.py*.

CHAPTER 6

Citing

Mingqing Liao, Yong Liu, Puchang Cui e.t. al. Elastic3rd: A tool for calculating third-order elastic constants from first principles submitted to computer physics communication (2020)

and give the following link: Elastic3rd

CHAPTER 7

Author Information

Mingqing Liao()
liaomq1900127@163.com
FGMS group @ Harbin Institute of Technology(HIT)

CHAPTER 8

Indices and tables

- genindex
- modindex
- search