Geometry Optimisation

Dr Jonathan Skelton

Department of Chemistry, University of Manchester
(jonathan.skelton@manchester.ac.uk)

Overview

- Geometry optimisation in the bulk
    - Theory
    - Geometry optimisation in VASP
- Theory *vs.* experiment
    - Choice of exchange-correlation functional
- Troubleshooting: what to do when things go wrong
- Phonon calculations (if time permits)
    - Imaginary modes, dynamical stability and mode mapping

Slides online: https://tinyurl.com/y6l6r6jg

Geometry optimisation: theory

Problem:

Find the set of atomic positions $\boldsymbol{R} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \dots, \boldsymbol{r}_N\}$ and lattice vectors $\boldsymbol{a} = \{\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3\}$ that define the minimum (a stationary point) on the potential-energy surface

Key parameters:

$$\boldsymbol{F}_N = -\frac{\partial E_0}{\partial \boldsymbol{r}_N} = -\left\langle \Psi_0(\boldsymbol{R}, \boldsymbol{a}) \left| \frac{\partial H(\boldsymbol{R}, \boldsymbol{a})}{\partial \boldsymbol{r}_N} \right| \Psi_0(\boldsymbol{R}, \boldsymbol{a}) \right\rangle$$

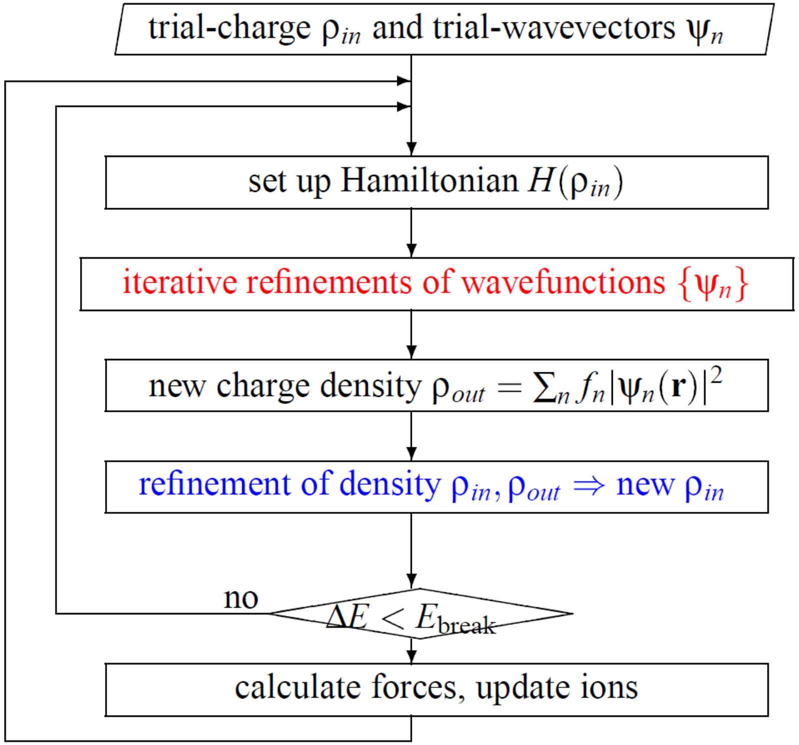
$$\sigma_{ij} = -\frac{\partial E_0(\boldsymbol{a})}{\partial \epsilon_{ij}} = -\left\langle \Psi_0(\boldsymbol{R}, \boldsymbol{a}) \left| \frac{\partial H(\boldsymbol{R}, \boldsymbol{a})}{\partial \epsilon_{ij}} \right| \Psi_0(\boldsymbol{R}, \boldsymbol{a}) \right\rangle$$

Approach:

Adjust an initial $\boldsymbol{R}$ and $\boldsymbol{a}$ to minimise E_0, $\boldsymbol{F}_N$ and/or σ_{ij} *via* an optimisation algorithm

https://www.vasp.at/vasp-workshop/slides/dft_introd.pdf

Geometry optimisation: theory



trial-charge ρ_{in} and trial-wavevectors ψ_n

set up Hamiltonian $H(\rho_{in})$

iterative refinements of wavefunctions $\{\psi_n\}$

new charge density $\rho_{out} = \sum_n f_n |\psi_n(\mathbf{r})|^2$

refinement of density $\rho_{in}, \rho_{out} \Rightarrow$ new ρ_{in}

$\Delta E < E_{\text{break}}$

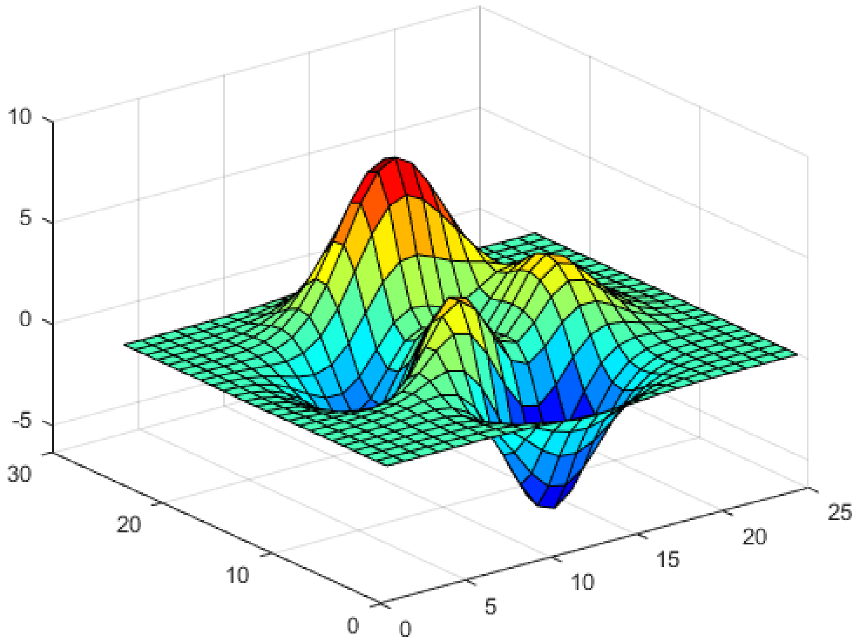
no

calculate forces, update ions

https://www.vasp.at/vasp-workshop/slides/optelectron.pdf

Geometry optimisation: theory

- In general, performing a geometry optimisation involves using an algorithm to locate a minimum on a multidimensional potential-energy surface (PES)



https://www.mathworks.com/help/matlab/visualize/creating-3-d-plots.html

The steepest descent algorithm

Problem:

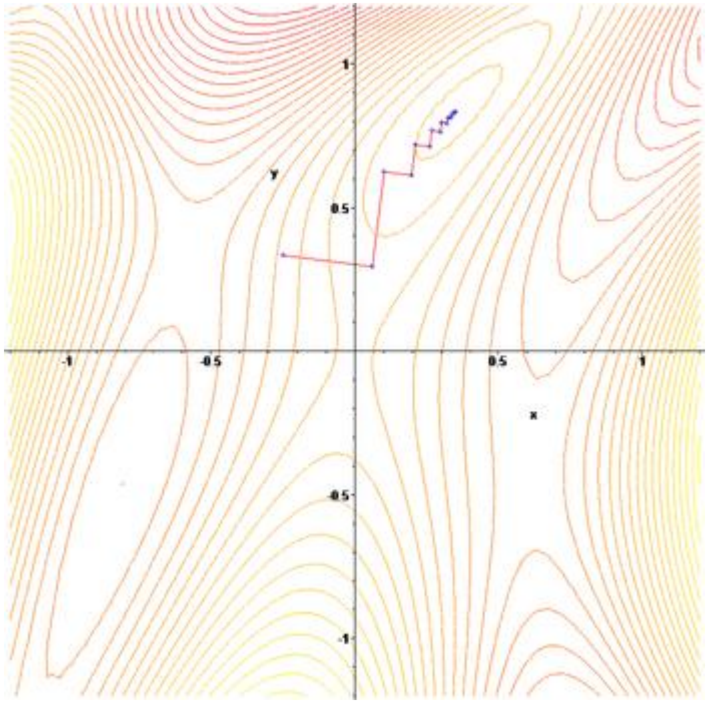
Follow the forces/stresses to the nearest minimum

Solution 1 - Steepest Descent:

$$\boldsymbol{R}_{n+1} = \boldsymbol{R}_n - \gamma \nabla E(\boldsymbol{R}_n) = \boldsymbol{R}_n + \gamma \boldsymbol{F}(\boldsymbol{R}_n)$$

Pros and cons:

Slow “zig-zag” convergence to the minimum; requires many iterations



https://en.wikipedia.org/wiki/Gradient_descent

The Newton-Rhapson method

Problem:

Follow the forces/stresses to the nearest minimum

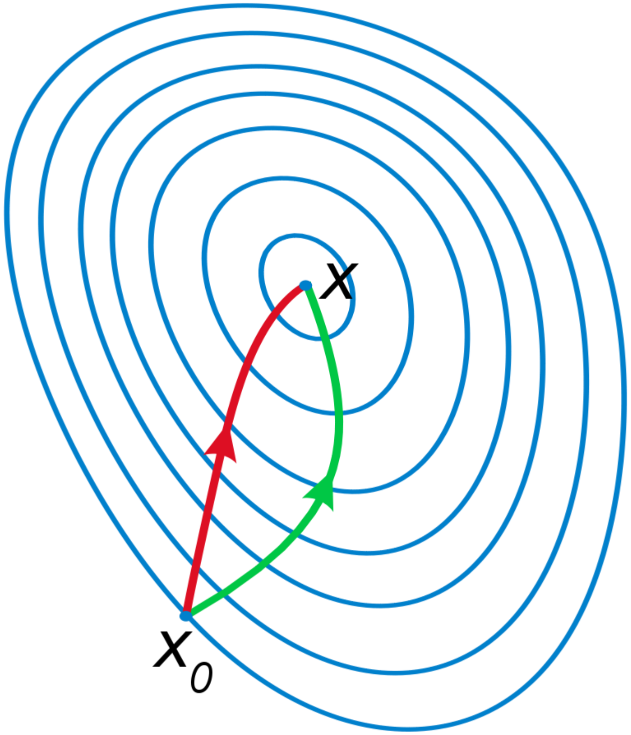
Solution 2 - Newton-Rhapson:

$$\boldsymbol{R}_{n+1} = \boldsymbol{R}_n - \gamma[\boldsymbol{H}(\boldsymbol{R}_n)]^{-1} \times \nabla E(\boldsymbol{R}_n)$$

$$\boldsymbol{H}(\boldsymbol{R}_n) = \partial^2 E / \partial \boldsymbol{R}_n^2$$

Pros and cons:

Fast convergence to nearest stationary point but requires a cheap (approximate) inverse Hessian



https://en.wikipedia.org/wiki/Newton%27s_method_in_optimization

The conjugate-gradient algorithm

Problem:

Follow the forces/stresses to the nearest minimum

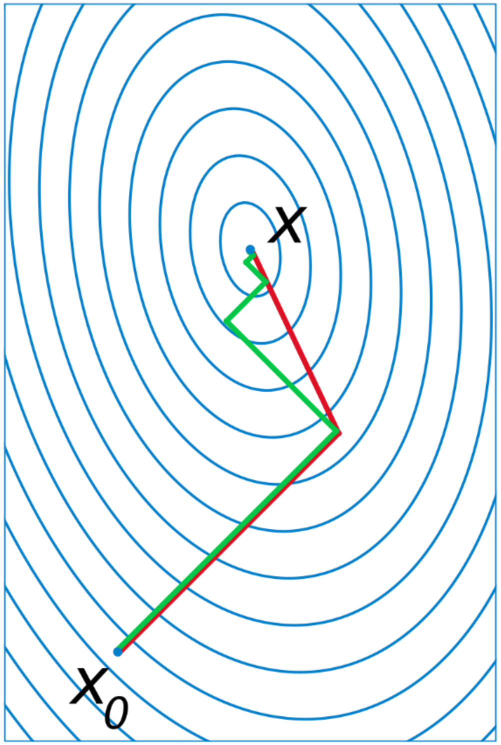
Solution 3 - Conjugate Gradient:

$$\boldsymbol{R}_{n+1} = \boldsymbol{R}_n - k_n \boldsymbol{h}_n$$

$$\boldsymbol{h}_n = \frac{\partial E(\boldsymbol{R}_n)}{\partial \boldsymbol{R}_n} + \gamma_n \boldsymbol{h}_{n-1}$$

Pros and cons:

A good middle ground between steepest descent and Newton-Rhapson for convergence speed and cost

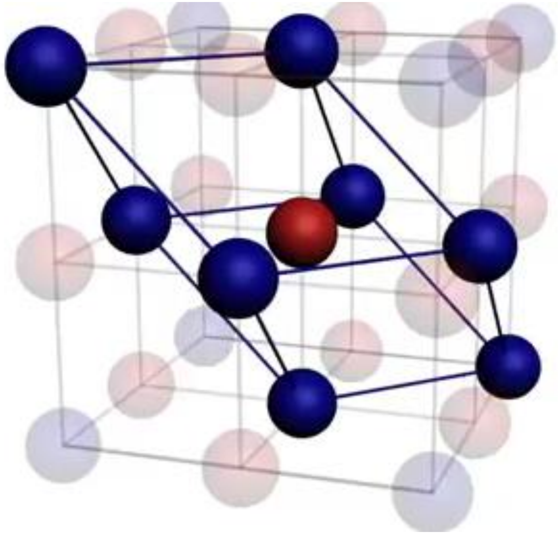


https://en.wikipedia.org/wiki/Gradient_descent
http://spindynamics.org/documents/cqc_lecture_6.pdf

Optimisation and symmetry

- Crystal symmetry can significantly reduce the number of degrees of freedom

- NaCl ($Fm\bar{3}m$):
  - Atoms fixed at $(0, 0, 0)$ and $\left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right)$
  - $a = b = c$ and $\alpha = \beta = \gamma = 90$
  - One DoF: the lattice constant a

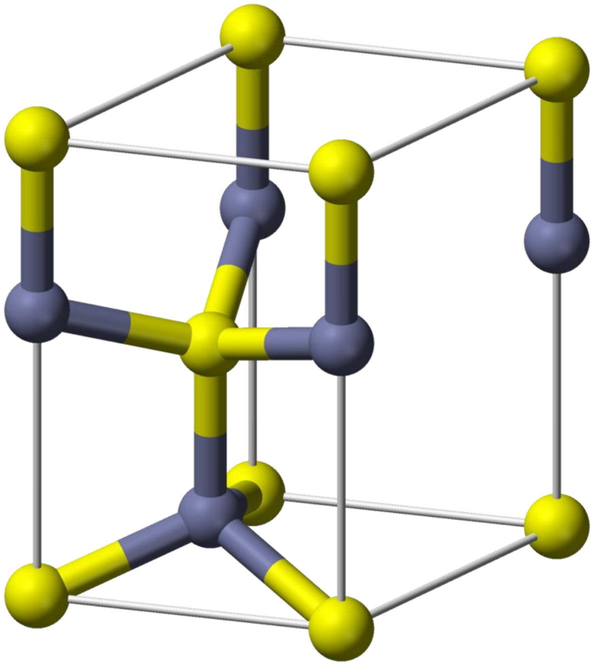


Physics-Animations.com: https://www.youtube.com/watch?v=lmQlxbU9CEI

Optimisation and symmetry

- Crystal symmetry can significantly reduce the number of degrees of freedom

- Wurtzite ZnO ($P6_3mc$):
  - Atoms at $(0,0,0)$ and $\left(\frac{1}{2},\frac{1}{2},\frac{1}{2}\right)$
  - $a = b \neq c, \alpha = \beta = 90$ and $\gamma = 120$
  - Three DoFs: a, c and u



https://en.wikipedia.org/wiki/Wurtzite#/media/File:Wurtzite-unit-cell-3D-balls.png

Important INCAR tags

- Geometry optimisation is activated by specifying the `IBRION` tag:
  - `IBRION = 0` performs ab initio MD

    (Not strictly geometry optimisation, but a few ps of high-temperature MD can help with structures that are trapped in a local minimum.)

  - `IBRION = 1` relaxes the ions and/or cell using a Quasi-Newtonian algorithm

    (Useful for geometries that are close to a minimum.)

  - `IBRION = 2` relaxes using the conjugate gradient algorithm

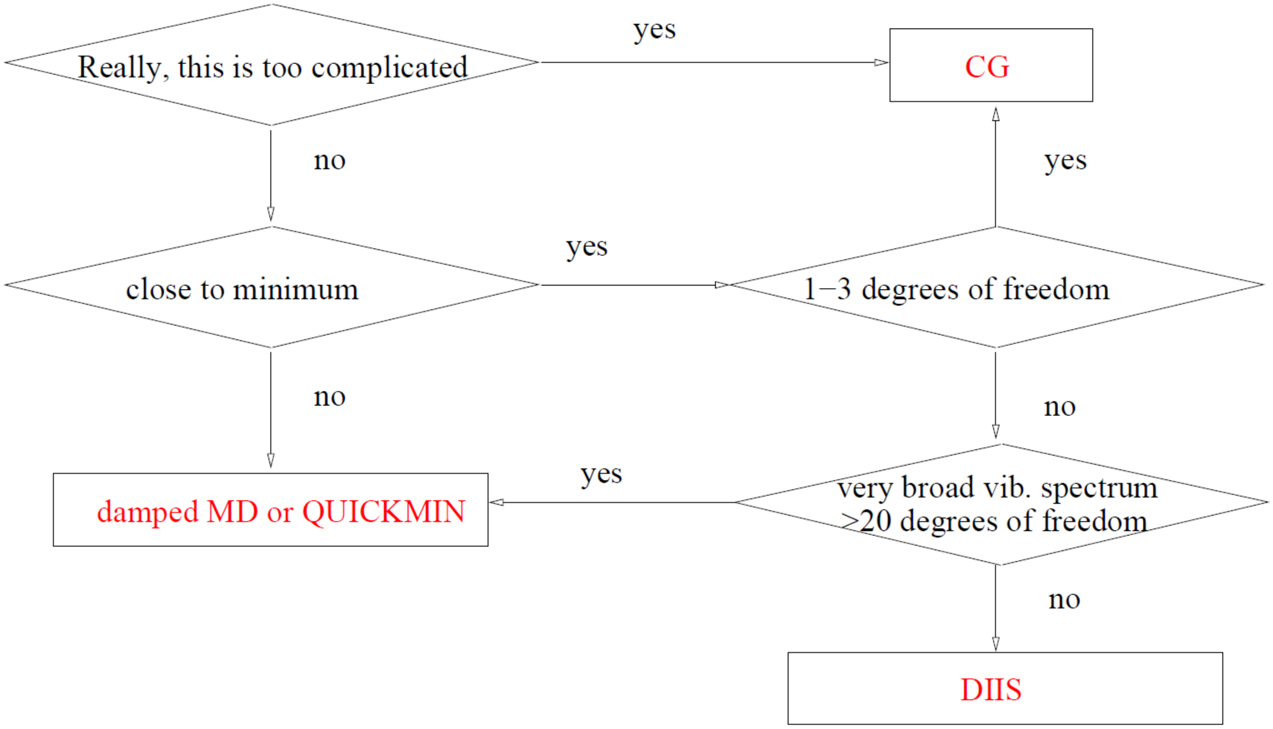
    (Good general choice, particularly for initial geometries that are away from a minimum.)

  - `IBRION = 3` relaxes with damped MD or the steepest-descent algorithm

    (I have never used this so I have no experience with it...)

https://cms.mpi.univie.ac.at/wiki/index.php/IBRION

Choice of IBRION



https://www.vasp.at/vasp-workshop/lectures/VASP_lecture_Basics3.pdf

Important INCAR tags

- The ISIF tag sets the details of what to optimise - choose between different combinations of the atom positions, the cell shape and the cell volume:

	Pos.	Shape	Vol.
`ISIF = 2`	✓	✗	✗
`ISIF = 3`	✓	✓	✓
`ISIF = 4`	✓	✓	✗
`ISIF = 5`	✗	✓	✗
`ISIF = 6`	✗	✓	✓
`ISIF = 7`	✗	✗	✓

https://cms.mpi.univie.ac.at/wiki/index.php/ISIF

Important `INCAR` tags

- The break criterion for the optimisation loop is set by the `EDIFFG` tag:
    - `EDIFFG > 0`: break when change in energy between successive ionic steps falls below $E_0(N) - E_0(N-1)$ < `EDIFFG`
    - `EDIFF = 0`: no break criterion - run for `NSW` steps
    - `EDIFFG < 0`: break when maximum force F_{max} < |`EDIFFG`|

- The number of steps in the optimisation is controlled by `NSW`:
    - Should be set to a relatively small value (e.g. `NSW = 10-25`) for `ISIF > 2`

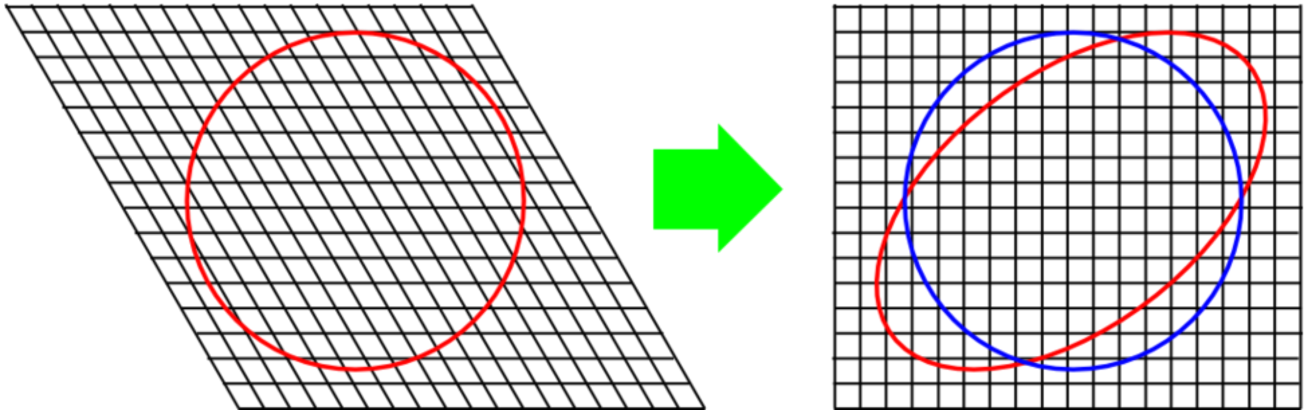
- `POTIM` controls how far the atoms move between ionic steps:
    - For relaxations (`IBRION = 1-3`), specifies a scaling constant (default: `POTIM = 0.5`)
    - For MD simulations (`IBRION = 0`), `POTIM` sets the MD timestep (no default - must be specified)

Important `INCAR` tags

- `ISYM` can be used to control symmetry handling
  - For optimisations, usually helpful to leave `ISYM = 2/3` (on - default)
  - For MD simulations, almost certainly want to set `ISYM = 0` (off)

Pulay stress

- The plane-wave basis depends on the reciprocal lattice vectors and hence the cell shape/volume: during a variable-cell optimisation the basis set becomes incomplete and leads to artificial Pulay forces



- To mitigate this:
  - Increase `ENCUT` to 1.3-2 × the max(`ENMAX`) in the `POTCAR` files
  - Perform variable-cell optimisations as a series of shorter steps (`NSW = 10-25`)

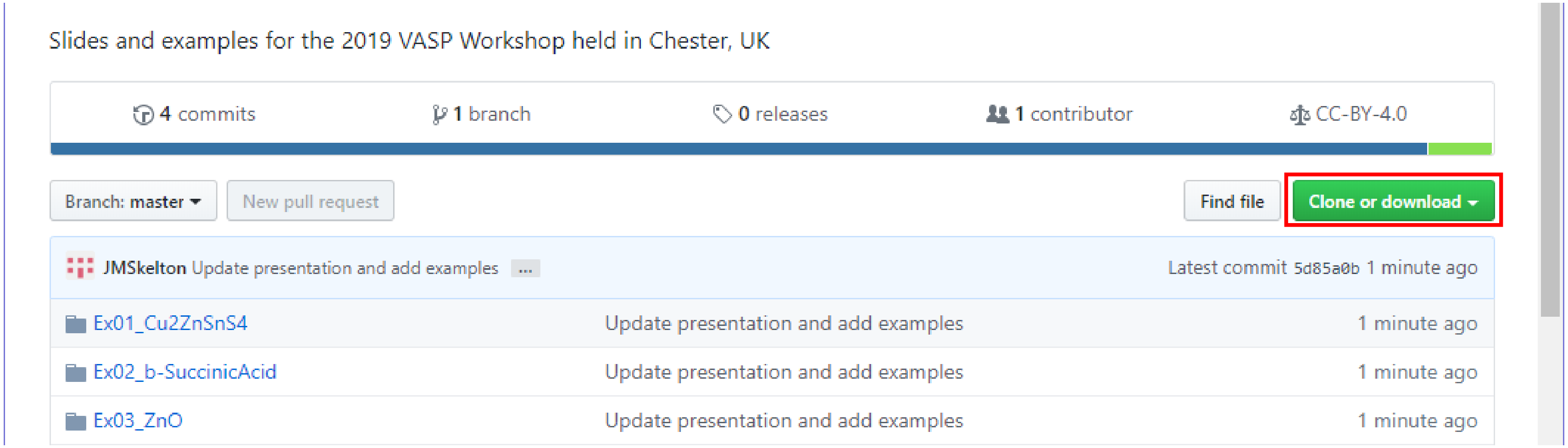
https://howlingpixel.com/i-en/Pulay_stress

Examples: https://tinyurl.com/y37naf92

- Download directly...:

```
git clone https://github.com/skelton-group/VASP-Workshop-Chester-
2019
```

- ... or download manually...:



Slides and examples for the 2019 VASP Workshop held in Chester, UK

4 commits | 1 branch | 0 releases | 1 contributor | CC-BY-4.0

JMSkelton Update presentation and add examples ... Latest commit 5d85a0b 1 minute ago

Ex01_Cu2ZnSnS4	Update presentation and add examples	1 minute ago
Ex02_b-SuccinicAcid	Update presentation and add examples	1 minute ago
Ex03_ZnO	Update presentation and add examples	1 minute ago

- ... or download from the VASP Wiki

Workshop examples

- Most of the examples include job files and Python scripts to automate some tasks:
  - `Opt.py`

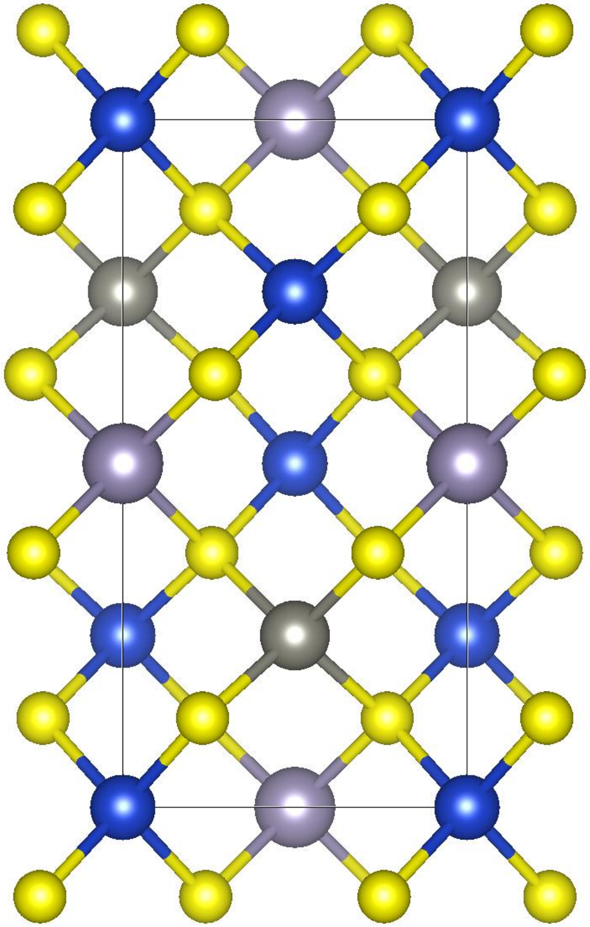
    Repeats a VASP calculation until a run completes in N < `NSW` steps - useful for variable-cell optimisations
  - `Opt-EV.py`

    Performs a series of geometry optimisations, as in `Opt.py`, starting from a `POSCAR` file and scaling the volume, and outputs an E/V curve.
  - `ModeMap.py`

    Reads phonon eigenvectors from an `OUTCAR` file, sets up and runs a series of single-point energy calculations on displacements of a reference `POSCAR` along a selected mode, and generates a mode PES
- If you find these useful, feel free to take them and adapt them to your own materials.
- (Disclaimer: The Python scripts should work for the workshop examples - I hope! - but they are not "bulletproof", so check they will work for what you need to do and note any warnings in the files.)

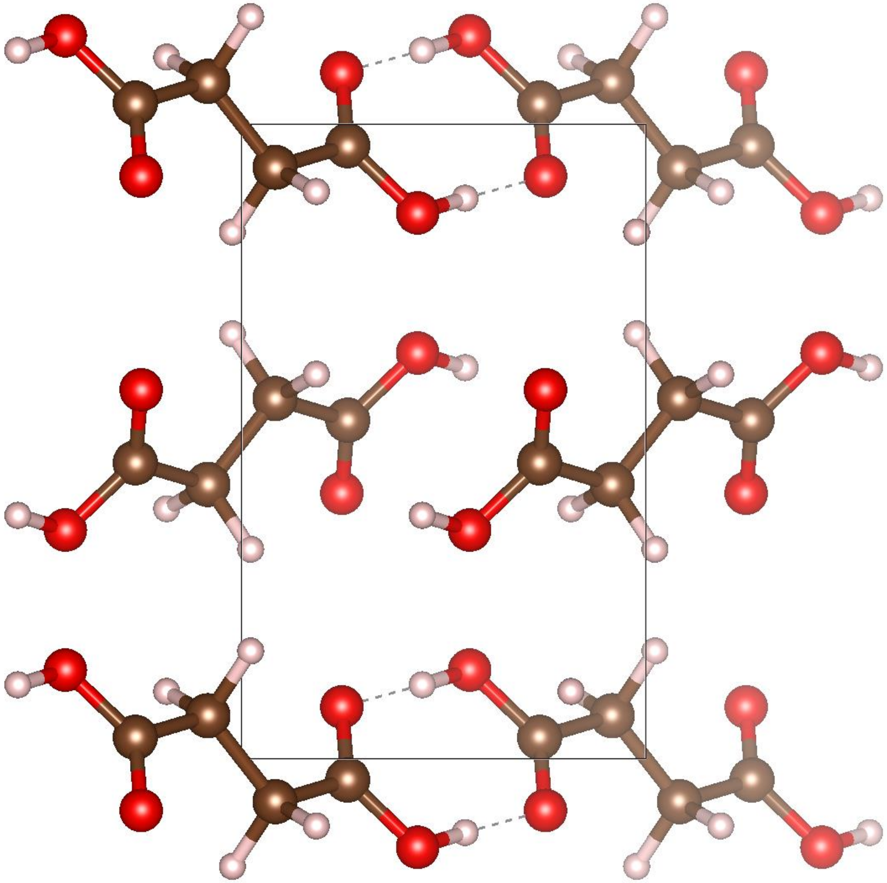
Example #1: Cu_2ZnSnS_4 (CZTS)

```
! Technical.

ENCUT = 600
GGA = PS
LASPH = .TRUE.
LREAL = .FALSE.
PREC = Accurate

! Electronic SCF.

ALGO = Normal
EDIFF = 1e-8
ISMEAR = 0
SIGMA = 0.01

! Geometry optimisation.

EDIFFG = -1e-2
IBRION = 1
ISIF = 3
NSW = 25
```



Example #2: β-succinic acid

```
! Technical.

ENCUT = 800
GGA = PE
IVDW = 11
LASPH = .TRUE.
LREAL = .FALSE.
PREC = Accurate

! Electronic SCF.

ALGO = Normal
EDIFF = 1e-8
ISMEAR = 0
SIGMA = 0.01

! Geometry optimisation.

EDIFFG = -1e-2
IBRION = 2
ISIF = 4
NSW = 25
```

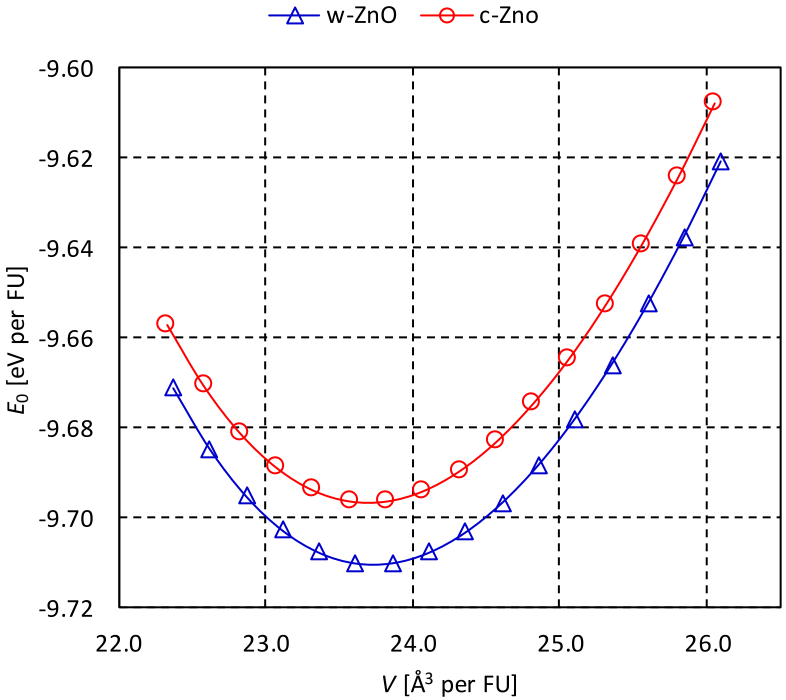


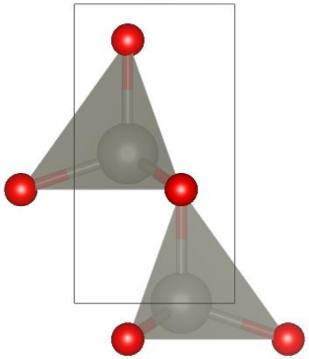
Energy/volume curves

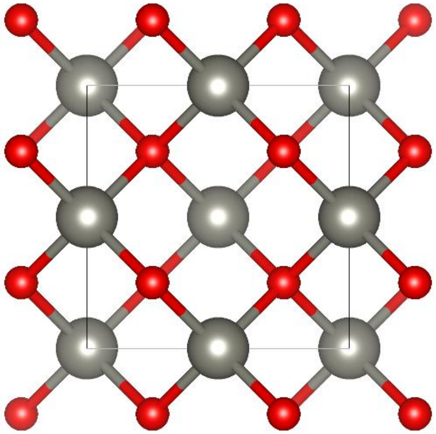
- An alternative way to obtain the equilibrium volume is to calculate an $E(V)$ curve:
    - Perform a series of constant-volume relaxations
    - Locate minimum by fitting to an equation of state (EoS) such as Birch-Murnaghan EoS - gives equilibrium volume V_0, energy E_0 and also bulk modulus B_0:

$$E(V) = E_0 + \frac{9V_0B_0}{16}\left\{\left[\left(\frac{V_0}{V}\right)^{\frac{2}{3}} - 1\right]^3 B_0' + \left[\left(\frac{V_0}{V}\right)^{\frac{2}{3}} - 1\right]^2 \left[6 - 4\left(\frac{V_0}{V}\right)^{\frac{2}{3}}\right]\right\}$$

Example #3: *c*-ZnO vs. *w*-ZnO







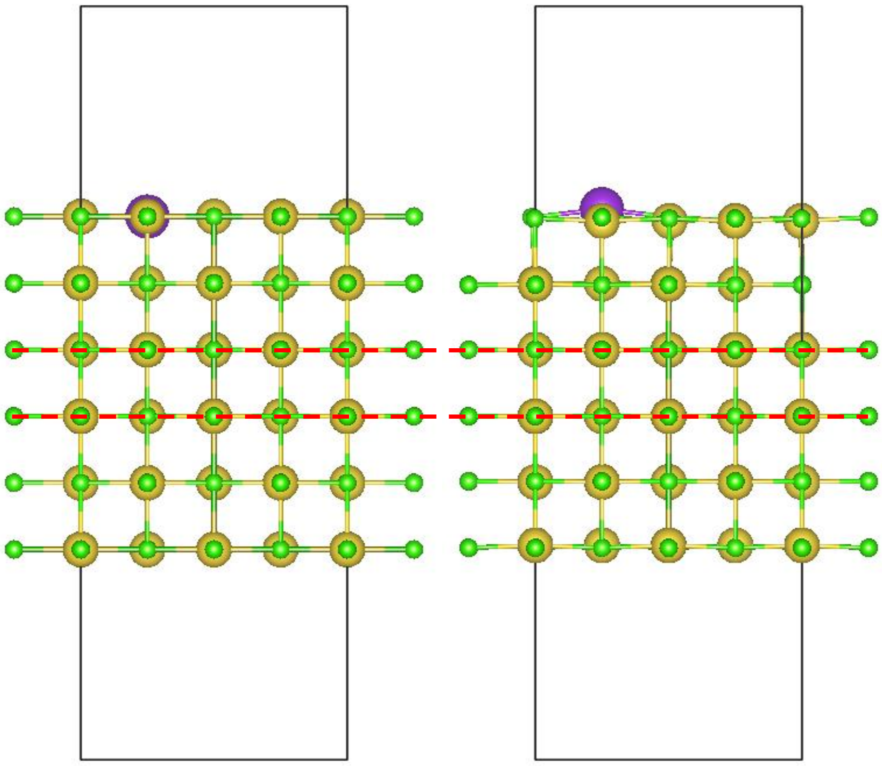
Selective dynamics

- The `Selective Dynamics` keyword in the `POSCAR` file can be used to freeze atomic positions during geometry optimisations (or MD simulations)

```
NaCl <100> (Na -> K)
   1.0000000000000000
    11.2804000000000002     0.0000000000000000     0.0000000000000000
     0.0000000000000000    11.2804000000000002     0.0000000000000000
     0.0000000000000000     0.0000000000000000    31.9206000000000003
   Na    Cl     K
   47    48     1
Selective Dynamics
Direct
     0.0000000000000000     0.0000000000000000     0.2791316579262295     T  T  T
     0.0000000000000000     0.0000000000000000     0.4558263315852459     F  F  F
     0.0000000000000000     0.0000000000000000     0.6325210052442624     T  T  T
     0.0000000000000000     0.2500000000000000     0.3674789947557377     T  T  T
     0.0000000000000000     0.2500000000000000     0.5441736684147541     F  F  F
     0.0000000000000000     0.2500000000000000     0.7208683420737706     T  T  T
     0.0000000000000000     0.4999999999999999     0.2791316579262295     T  T  T
     0.0000000000000000     0.4999999999999999     0.4558263315852459     F  F  F
     0.0000000000000000     0.4999999999999999     0.6325210052442624     T  T  T
     0.0000000000000000     0.7500000000000000     0.3674789947557377     T  T  T
     0.0000000000000000     0.7500000000000000     0.5441736684147541     F  F  F
```

Example #4: NaCl <100> with K defect

```
NaCl <100> (Na -> K)
   1.0000000000000000
    11.2804000000000002     0.0000000000000000     0.0000000000000000
     0.0000000000000000    11.2804000000000002     0.0000000000000000
     0.0000000000000000     0.0000000000000000    31.9206000000000003
   Na    Cl     K
   47    48     1
Selective Dynamics
Direct
     0.00000...     0.00000...     0.27913...     T  T  T
     0.00000...     0.00000...     0.45582...     F  F  F
     0.00000...     0.00000...     0.63252...     T  T  T
     0.00000...     0.25000...     0.36747...     T  T  T
     0.00000...     0.25000...     0.54417...     F  F  F
     0.00000...     0.25000...     0.72086...     T  T  T
     0.00000...     0.49999...     0.27913...     T  T  T
     0.00000...     0.49999...     0.45582...     F  F  F
     0.00000...     0.49999...     0.63252...     T  T  T
     0.00000...     0.75000...     0.36747...     T  T  T
     0.00000...     0.75000...     0.54417...     F  F  F
     0.00000...     0.75000...     0.72086...     T  T  T
     0.25000...     0.00000...     0.36747...     T  T  T
     0.25000...     0.00000...     0.54417...     F  F  F
     0.25000...     0.00000...     0.72086...     T  T  T
     0.25000...     0.25000...     0.27913...     T  T  T
     0.25000...     0.25000...     0.45582...     F  F  F
     0.25000...     0.25000...     0.63252...     T  T  T
     0.25000...     0.49999...     0.36747...     T  T  T
     0.25000...     0.49999...     0.54417...     F  F  F
     0.25000...     0.49999...     0.72086...     T  T  T
     0.25000...     0.75000...     0.27913...     T  T  T
     0.25000...     0.75000...     0.45582...     F  F  F
     0.25000...     0.75000...     0.63252...     T  T  T
     0.49999...     0.00000...     0.27913...     T  T  T
     0.49999...     0.00000...     0.45582...     F  F  F
     0.49999...     0.00000...     0.63252...     T  T  T
```



Choosing an XC functional

- Provided other technical parameters - `ENCUT`, `KPOINTS`, `PREC` - are converged, the choice of XC functional is the most important factor for getting good results

- Although LDA and PBE are very widely used in the literature, they are not always the best for geometry optimisation (!)

- It is fairly common practice to optimise the geometry with one XC functional and model the electronic structure with another (e.g. PBEsol → HSE 06)

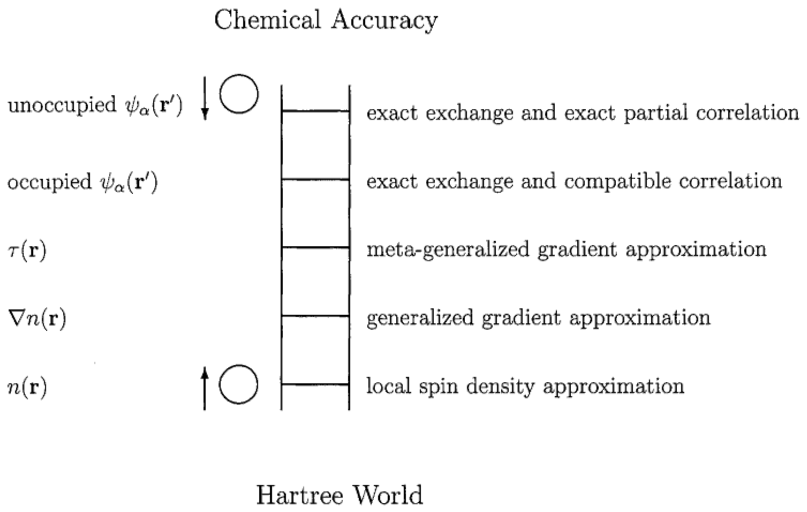

Chemical Accuracy
unoccupied $\psi_\alpha(\mathbf{r}')$
exact exchange and exact partial correlation
occupied $\psi_\alpha(\mathbf{r}')$
exact exchange and compatible correlation
$\tau(\mathbf{r})$
meta-generalized gradient approximation
$\nabla n(\mathbf{r})$
generalized gradient approximation
$n(\mathbf{r})$
local spin density approximation
Hartree World

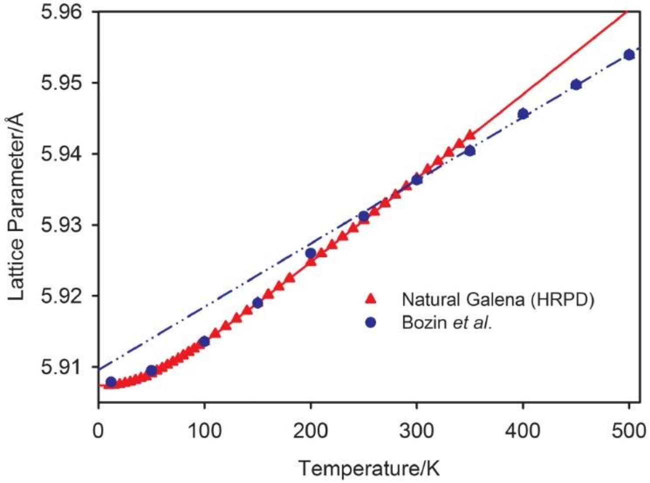
FIGURE 1. Jacob's ladder of density functional approximations. Any resemblance to the Tower of Babel is purely coincidental. Also shown are angels in the spherical approximation, ascending and descending. Users are free to choose the rungs appropriate to their accuracy requirements and computational resources. However, at present their safety can be guaranteed only on the two lowest rungs.

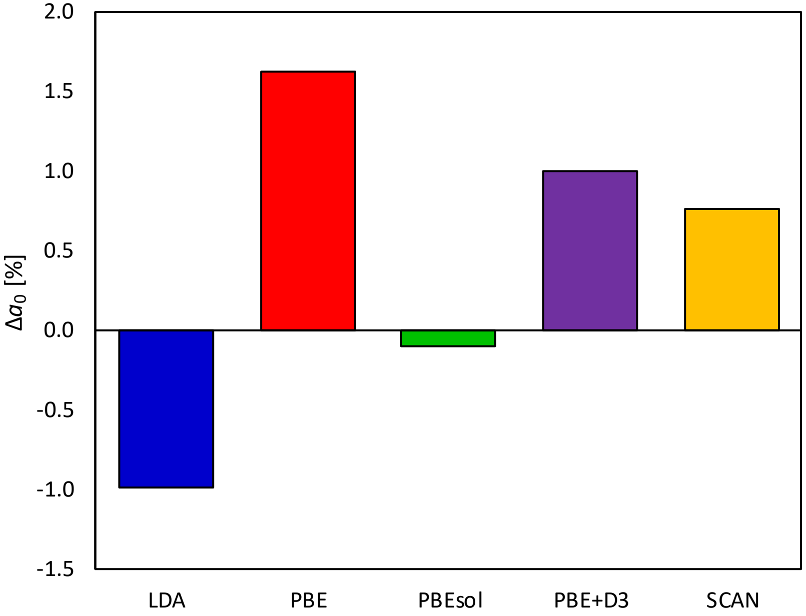
Perdew and Schmidt, *AIP Conf. Proc.* **577**, 1 (**2001**), DOI: 10.1063/1.1390175

Some useful XC functionals

- LDA:
  - Selected automatically when using `POTCAR` files from the `potpaw_LDA.*` set
- PBE (standard GGA):
  - Set `GGA = PE` or use the `potpaw_PBE.*` set.
- PBEsol (improved GGA for solids):
  - Set `GGA = PS`
- PBE + D3 (PBE GGA with the D3 dispersion correction):
  - Add `IVDW = 11` to a PBE calculation
- SCAN (modern meta-GGA):
  - Add `METAGGA = SCAN` to a PBE calculation
  - Usually best to use `ALGO = Normal` and `LMIXTAU = .TRUE.`

Example #5: PbS





$$a(T) = a_0 + k\frac{1}{[\exp(E/T) - 1]}$$

Knight, *J. Phys.: Condens. Matter* **26**, 385403 (**2014**), DOI: 10.1088/0953-8984/26/38/385403

Geometry optimisation: summary

1. Obtain a good starting structure:
   - Preferably choose a high-quality single-crystal X-ray or neutron structure collected at low T
   - Be careful with structures containing disorder/partial occupancies
   - The Materials Project website (https://materialsproject.org) is a good alternative to the ICSD for downloading initial structures

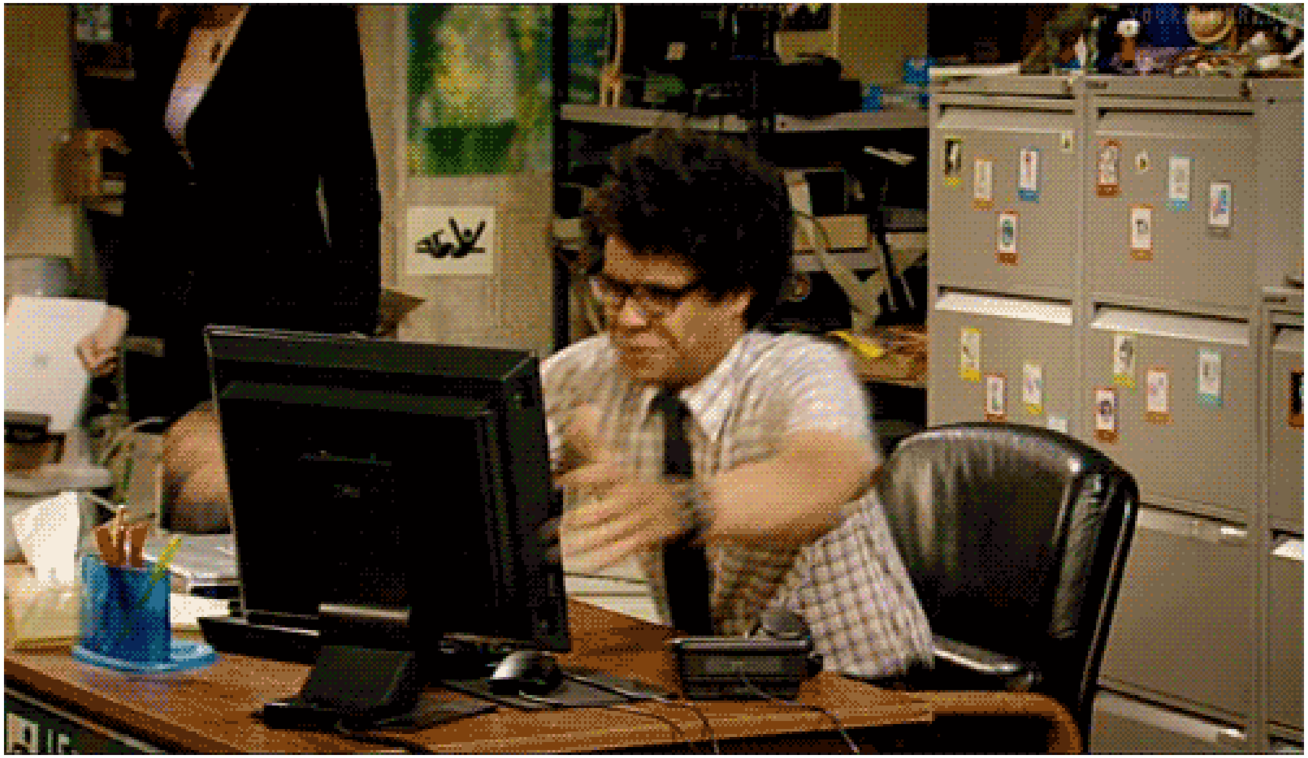
2. Choose a good XC functional:
   - Hard bulk materials (e.g. oxides): try PBEsol (`GGA = PS`)
   - Layered or molecular materials: try PBE + D3 (`GGA = PE` with `IVDW = 11`)

3. Choose your `ISIF`:
   - `ISIF = 2`: internal coordinates (atom positions) only
   - `ISIF = 4`: cell shape with fixed volume
   - `ISIF = 3`: full variable-cell optimisation

Geometry optimisation: summary

4. Adjust your `ENCUT` if required:
   - 1.3-2 × max(`ENMAX`) for accurate forces and stresses - especially important for variable-cell optimisations
5. Choose your `IBRION`:
   - Start with `IBRION = 2` and change to `IBRION = 1` as the ionic energy begins to converge
6. Make sure your electronic structure is well converged:
   - If the electronic SCF fails to converge, the forces will probably not be accurate -> try `ALGO = Normal/All` and increase `NELM` if required
   - If using a loose `EDIFF`, the forces might not be accurate enough to converge to a tight `EDIFFG` (e.g. `EDIFF = 1e-4` with `EDIFFG = -1e-2`)
   - A failed SCF step in the middle of an optimisation could easily undo several prior successful minimisation steps

What to do when things go wrong?



Source: MovieGif.com *via* http://gph.is/1xzJdX3

Troubleshooting

Problem:

My optimisation is taking a long time to converge.

Things to check:

1. Are your electronic SCF steps converging consistently? One failed SCF can undo several optimisation steps.
2. Is your `EDIFFG` sane? Reaching `EDIFFG = -1e-3` in a large, low-symmetry structure is hard, and rarely actually necessary.
3. Are your `EDIFF` and `EDIFFG` compatible? (See summary slides.)

Things to try:

1. Change `IBRION` (`IBRION = 1/2`).
2. For systems with H atoms or a relatively flat PES, try `POTIM = 0.25`).

Troubleshooting

Problem:

My optimisation is diverging.

Things to check:

1. Are your electronic SCF steps converging?
2. Is your input structure reasonable? (Experimental structures are definitely *not* infallible - check for guesstimated H positions, partial occupancies, and dodgy CIF files.)
3. Is your `POTCAR` file ordered correctly?
4. If using spin polarisation, are your `MAGMOM`/`NUPDOWN` set correctly?

Things to try:

1. Start with `ISIF = 2` to constrain volume/shape and/or do a few ionic steps with a very small `POTIM` (e.g. `POTIM = 0.05`) to constrain initial explosion.

Troubleshooting

Problem:

My optimised structure doesn't match experiment/previous calculations.

Things to check:

1. If comparing to experiment:
    - Is the experimental structure good quality?
    - Have you considered thermal expansion?
    - Ideally want single-crystal X-ray or neutron data recorded close to 0 K.
2. If comparing to other calculations:
    - Are you using a comparable setup? (XC functional, cutoff, *k*-point sampling, ...)
    - Are your chosen parameters more appropriate than those used in previous work?

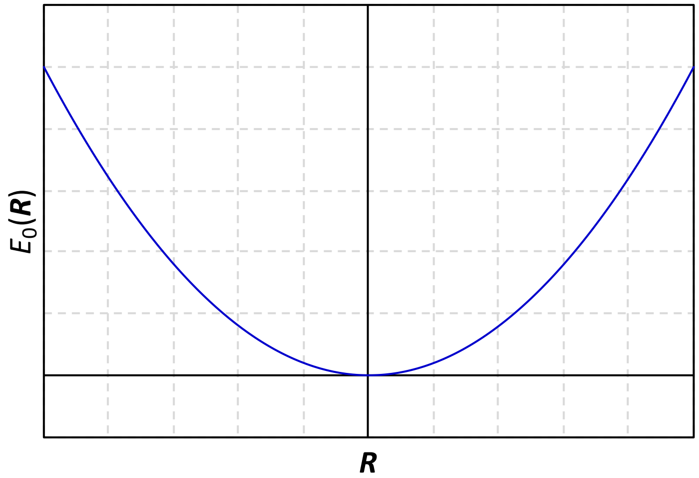
Things to try:

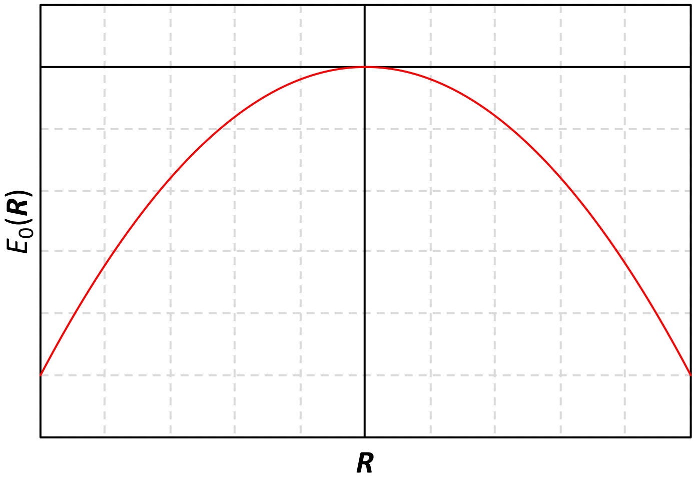
1. Try different XC functionals - PBEsol or PBE-D3 instead of PBE.

Not all stationary points are equal...

Problem:

Find the set of atomic positions $\boldsymbol{R} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \dots, \boldsymbol{r}_N\}$ and lattice vectors $\boldsymbol{a} = \{\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3\}$ that define the minimum (a stationary point) on the potential-energy surface.





Minimum: $\frac{\partial^2 E}{\partial \boldsymbol{R}^2} > 0$

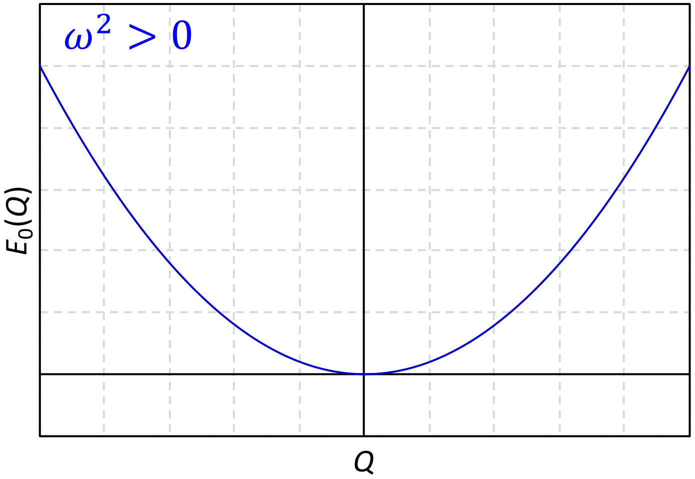
Maximum: $\frac{\partial^2 E}{\partial \boldsymbol{R}^2} < 0$

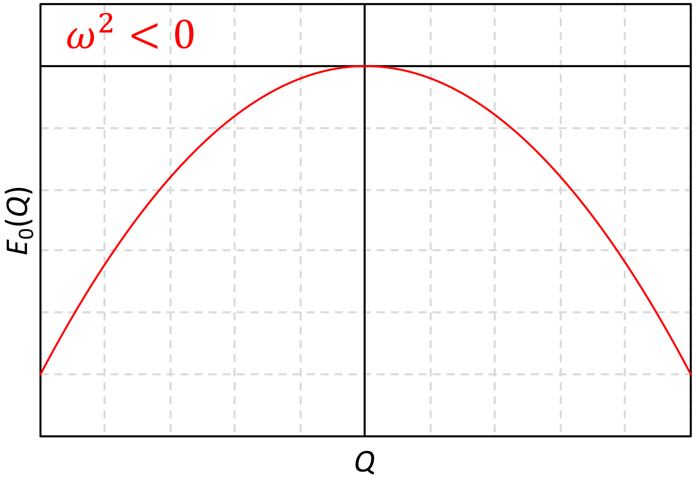
Phonon calculations

Solution:

Computing the force constants $\mathbf{\Phi} = \partial^2 E / \partial \boldsymbol{R}^2$ and obtaining the harmonic phonon frequencies from the dynamical matrix $\boldsymbol{D}(\mathbf{q})$ allows us to confirm we have located a minimum

$$D_{\alpha\beta}\,(jj',\mathbf{q}) = \frac{1}{\sqrt{m_j m_{j'}}} \sum_{l'} \Phi_{\alpha\beta}(j0, j'l') \exp[i\mathbf{q} \cdot \{\boldsymbol{r}(j'l') - \boldsymbol{r}(j'l')\}]$$



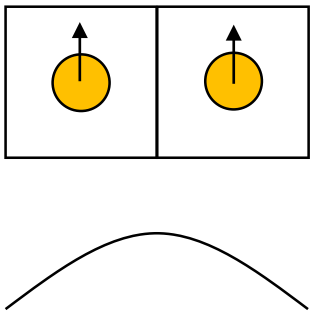


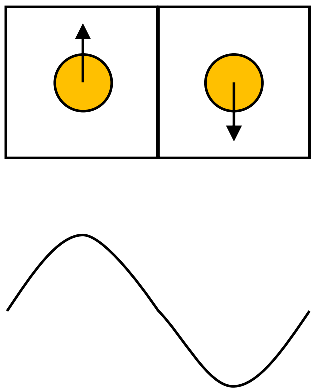
Phonon calculations in VASP

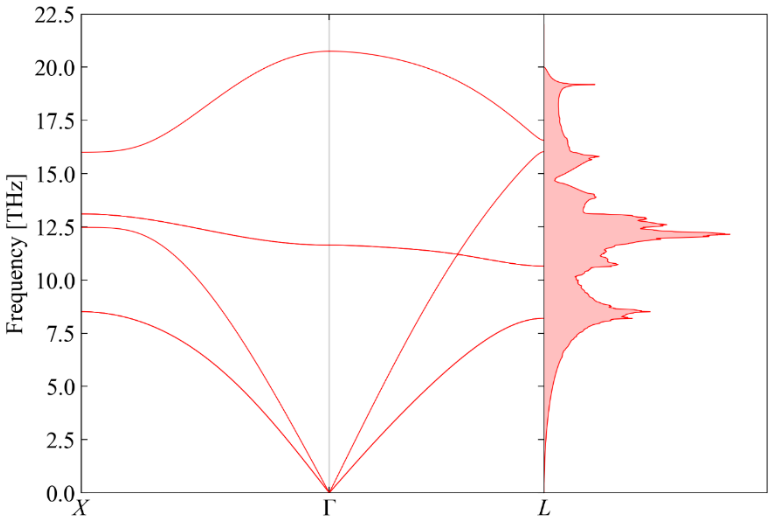
- Phonon calculations are activated by setting `IBRION = 5-8`:
  - `IBRION = 5/6`: calculate force constants and determine $\boldsymbol{D}(\mathbf{q} = \Gamma)$ using finite differences
  - `IBRION = 7/8`: evaluate $\boldsymbol{D}(\mathbf{q} = \Gamma)$ using DFPT (not available for all functionals)
  - `IBRION = 6/8` take into account symmetry, whereas `IBRION = 5/7` do not
- For finite differences calculations (`IBRION = 5/6`):
  - `POTIM` sets the finite difference step (default: 1.5 × 10^{-2} Å)
  - `NFREE` sets the number of steps in the central difference scheme (default: 2)
- For at least some versions of VASP it is also necessary to specify `NSW = 1`
- In a finite-differences calculation (`IBRION = 5/6`), adding `ISIF = 3` determines the elastic constant matrix and the ionic contribution to the dielectric constant ε
- In either calculation, Born charges and the polarisation contribution to ε can be obtained using `LEPSILON` (DFPT) or `LCALCEPS` (finite field)

Phonon calculations in VASP

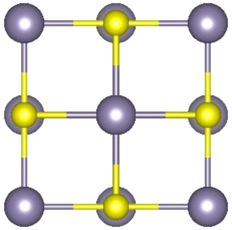
- The major drawback is that at present the VASP routines only calculate $\boldsymbol{D}(\mathbf{q} = \Gamma)$...
- ... but the Phonopy package provides a full set of functionality and works very well with VASP as a force calculator: https://atztogo.github.io/phonopy/

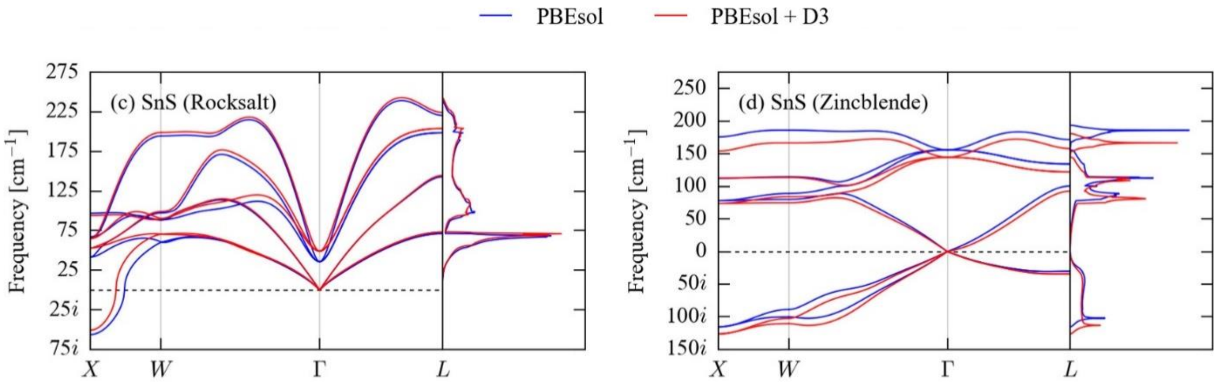

Zone centre (Γ)

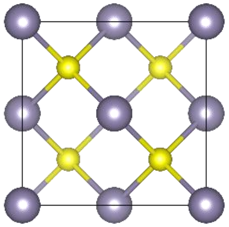

Zone boundary

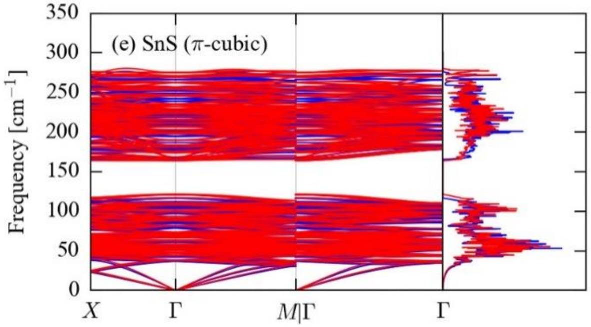


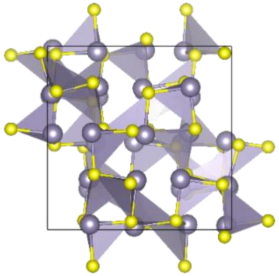
Structural instabilities: “cubic” SnS







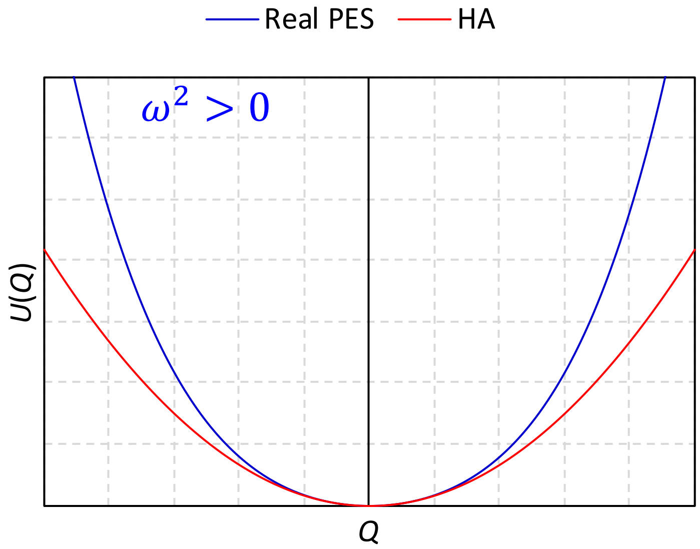


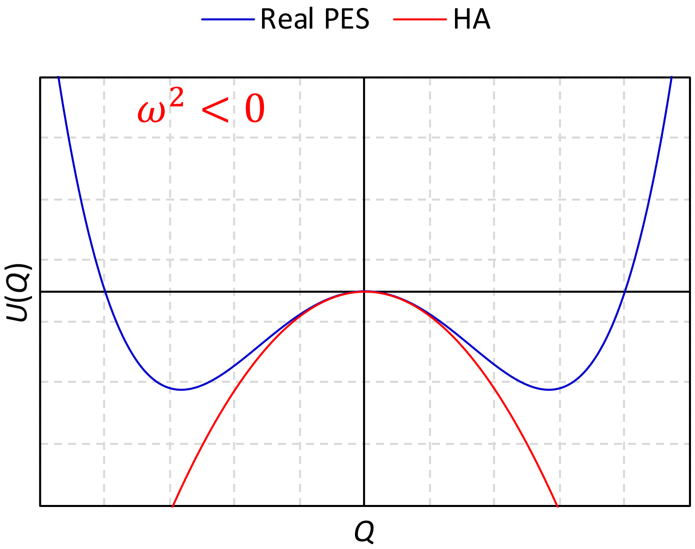


Skelton *et al.*, *J. Phys. Chem. C* **121** (*12*), 6446 (**2017**), DOI: 10.1021/acs.jpcc.6b12581

Are imaginary modes always bad?

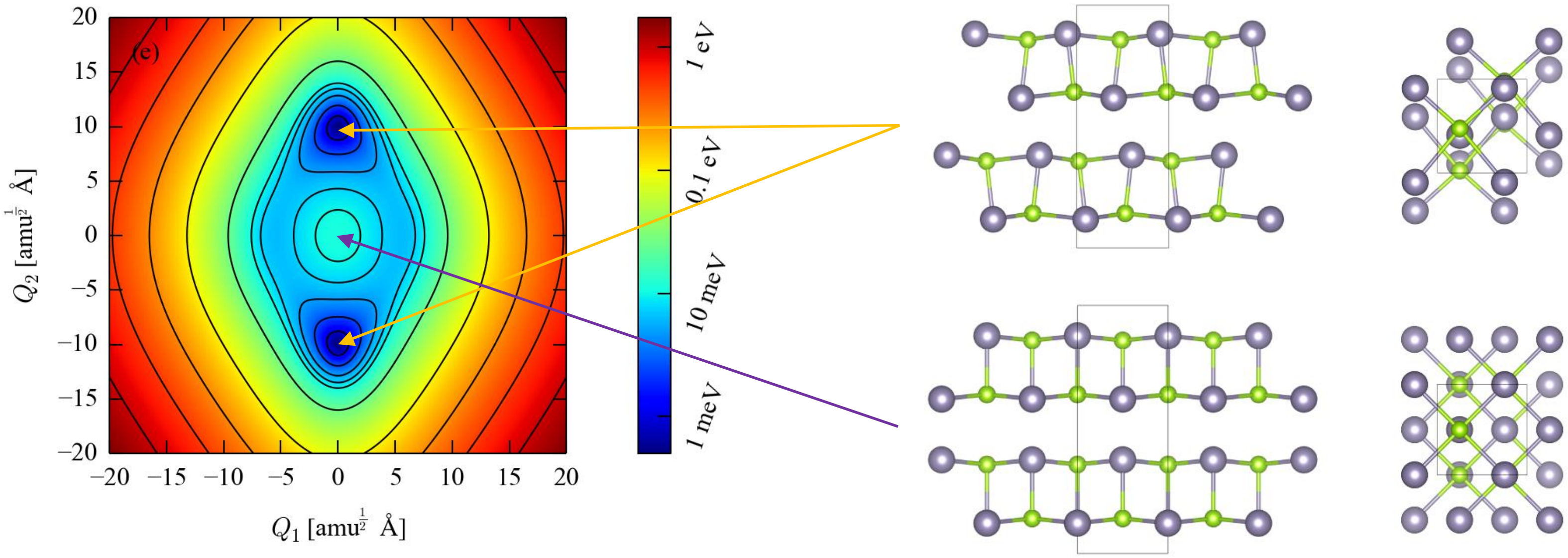
- In some systems, equivalent energetic minima are separated by a shallow energy barriers - at high temperatures, sufficient thermal energy is available to hop between minima on a short timescale, and crystallography measures the average structure





Are imaginary modes always bad?

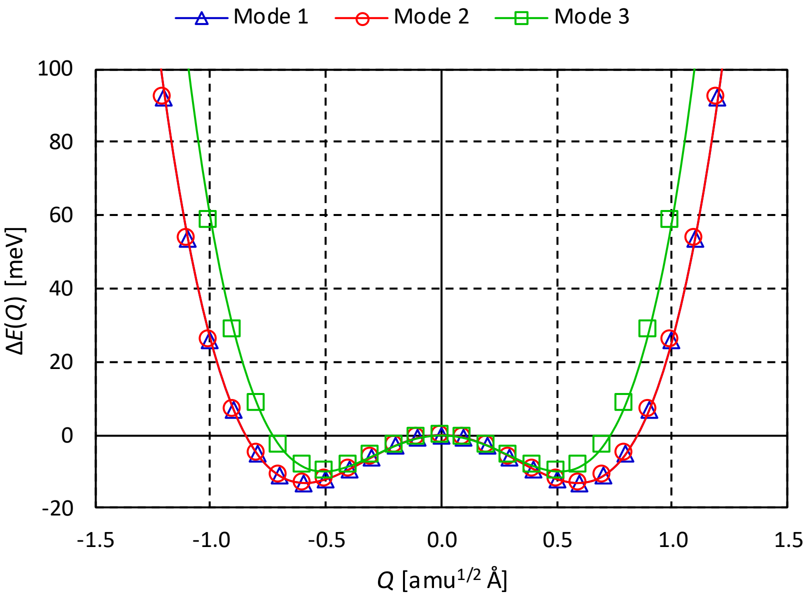
- Many examples of this phenomenon in real materials:
  - Octahedral rotation and cation off-centering in cubic perovskites
  - *Pnma* ↔ *Cmcm* phase transition in SnS/SnSe
  - The paraelectric ↔ ferroelectric phase transition in ferroelectrics

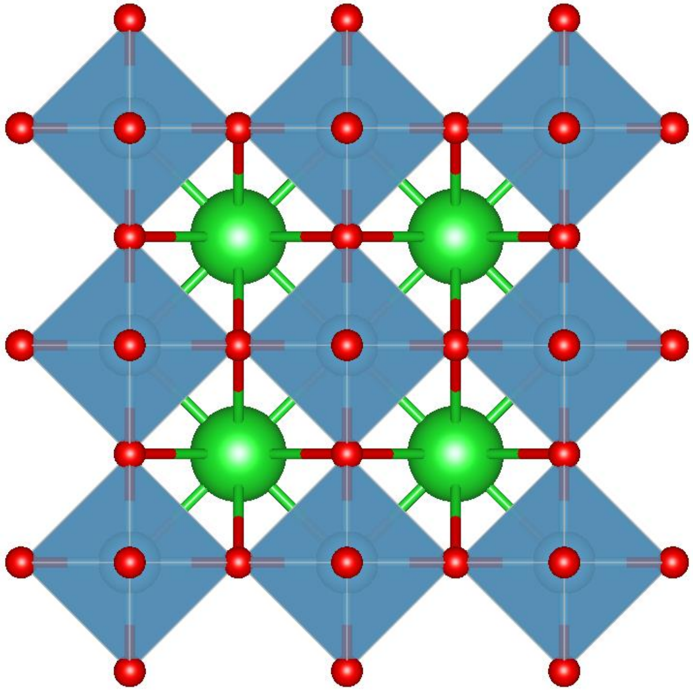


Skelton *et al.*, *Phys. Rev. Lett.* **117**, 075502 (**2016**), DOI: 10.1103/PhysRevLett.117.075502

Example #6: Cubic $BaTiO_3$

- *c*-$BaTiO_3$ displays a spontaneous off-centring of the Ti^{4+} cation within the TiO_6 octahedral cage - manifests as a triply-degenerate imaginary mode with $\nu \approx 200i$ cm^{-1}





Thankyou for listening!

Any questions?