
LModeA-nano

Release v1

YT

Aug 14, 2023

GETTING STARTED

1	Recent news and publications	3
1.1	Install LModeA-nano	3
1.2	Install PyMOL	3
1.3	Quickstart Examples	5
1.4	Prerequisite Knowledge	9
1.5	Local Vibrational Mode Theory	9
1.6	Solid-state Modeling Packages	11
1.7	Quantum Chemistry Modeling Packages	39
1.8	Finite Difference Method	42
1.9	Frequently Asked Questions	43
1.10	Need Help	44
1.11	Citing LModeA-nano	44
1.12	Application of LModeA-nano	45
2	Acknowledgement & Funding	47

This manual focuses on the LModeA-nano as a PyMOL plugin designed to perform Local Vibrational Mode Analysis on both solids and molecules. This theoretical tool can quantify the intrinsic strength of chemical bonds in terms of local stretching force constant.

RECENT NEWS AND PUBLICATIONS

[2022-09-26] LModeA-nano was employed by Prof. Peter W. Roesky's research group to investigate the existence of Si-Si bonding within silylene compounds in their work titled **Stimuli Responsive Silylene: Electromerism Induced Reversible Switching Between Mono- and Bis-Silylene** published on *Angew. Chem. Int. Ed.*

[2021-12-20] Our work titled **Capturing Individual Hydrogen Bond Strengths in Ices via Periodic Local Vibrational Mode Theory: Beyond the Lattice Energy Picture** was published on *J. Chem. Theory Comput.*

1.1 Install LModeA-nano

LModeA-nano is a PyMOL plugin, therefore, PyMOL program (version ≥ 2.5) has to be installed first. If PyMOL is not installed on your computer yet, please check [Install PyMOL](#).

The latest code of LModeA-nano can be obtained at GitHub repo <https://github.com/smutao/LModeA-nano>. By clicking **Code** \rightarrow **Download ZIP**, the source package LModeA-nano-main.zip is downloaded to your disk and this zip file needs to be uncompressed as a folder before the next step.

[GitLab repo <https://gitlab.com/smutao/LModeA-nano> Download]

After launching PyMOL program on your computer, navigate to the menu bar **Plugin** \rightarrow **Plugin Manager**. In the pop-up dialog window **Plugin Manager**, navigate to **Install New Plugin** tab, click **Choose file...** button. Then find the LModeA-nano-main folder and choose `__init__.py`, click **OK** twice in the pop-up windows. LModeA-nano is now successfully installed and you should be able to see **LModeA-nano** in the drop-down list of **Plugin** menu of PyMOL.

1.2 Install PyMOL

1.2.1 Binary from official website

Download PyMOL package from the [official website](#) maintained by Schrödinger. This incentive binary version has bundled Python environment and necessary packages.

We recommend using this version of PyMOL due to the easy installation.

1.2.2 Pre-built open-source version

Conda (miniconda/anaconda)[Mac/Linux/Win] users

Open-source pymol has released package on [conda](#).

Please run in the terminal

```
$ conda create -n pymol-opensource
$ conda activate pymol-opensource
$ conda install -c conda-forge pymol-open-source
$ conda install -c anaconda numpy
$ pymol
```

Mac OS users

Latest PyMOL is available in the third-party package managers like Homebrew and MacPorts. See https://pymolwiki.org/index.php/MAC_Install.

For Apple Silicon M1/M2 users, Homebrew is a good choice to install PyMOL via “brew install pymol”.

Linux users

Latest PyMOL can be installed via the native package manager (e.g. yum, pacman) of the Linux distribution. See https://pymolwiki.org/index.php/Linux_Install.

For Ubuntu-based distros (e.g. Ubuntu, Pos OS), PyMOL 2.5+ is available in the software repository for 22.04 LTS and later versions. Therefore, it can be easily installed via “sudo apt install python3-pymol”.

PyMOL from Snap and Flatpak package managers has not been tested and it might have issue when installing LModeA-nano as a plugin.

Windows users

Christoph Gohlke from UC Irvine provides the pre-built open-source version of PyMOL for Windows on [his website](#).

Instructions on how to install this pre-compiled PyMOL can be found at

- https://pymolwiki.org/index.php/Windows_Install#Open-Source_PyMOL
- <https://omicx.cc/posts/2021-04-20-install-pymol-windows/>
- <https://www.youtube.com/watch?v=B5GbuAND3IQ>

For LModeA-nano to function properly, the PyQt5 package should be installed. One may try “pip install pyqt5” if LModeA-nano has problem while launching.

1.2.3 Compile open-source version

For Linux users who are familiar with compiling programs, it is possible to build the open-source PyMOL from [source code](https://pymolwiki.org/index.php/Linux_Install#Install_from_source) on GitHub. Detailed instructions can be found at https://pymolwiki.org/index.php/Linux_Install#Install_from_source.

Note: To make LModeA-nano properly work, the Python environment which compiles the PyMOL source code needs the `numpy` package installed.

1.3 Quickstart Examples

The following two examples may familiarize you with the use of LModeA-nano.

1.3.1 Ethane calculated by Gaussian 16

In the LModeA-nano-main/quickstart/ethane directory, a formatted checkpoint file `ethane-pbe.fchk` was generated by the vibrational analysis (freq) calculation in Gaussian 16 package.

For this type of input data file, it can be directly loaded to LModeA-nano.

Following these steps

- open a new PyMOL window and launch LModeA-nano by clicking **Plugin** → **LModeA-nano**
- click the ... (**Browse**) button and select the `ethane-pbe.fchk` file
- change the program drop-down list from **VASP** to **Gaussian/Q-Chem**
- click **Load**
- change **Dimensions** to 0 and click **Confirm** button

We will see the following

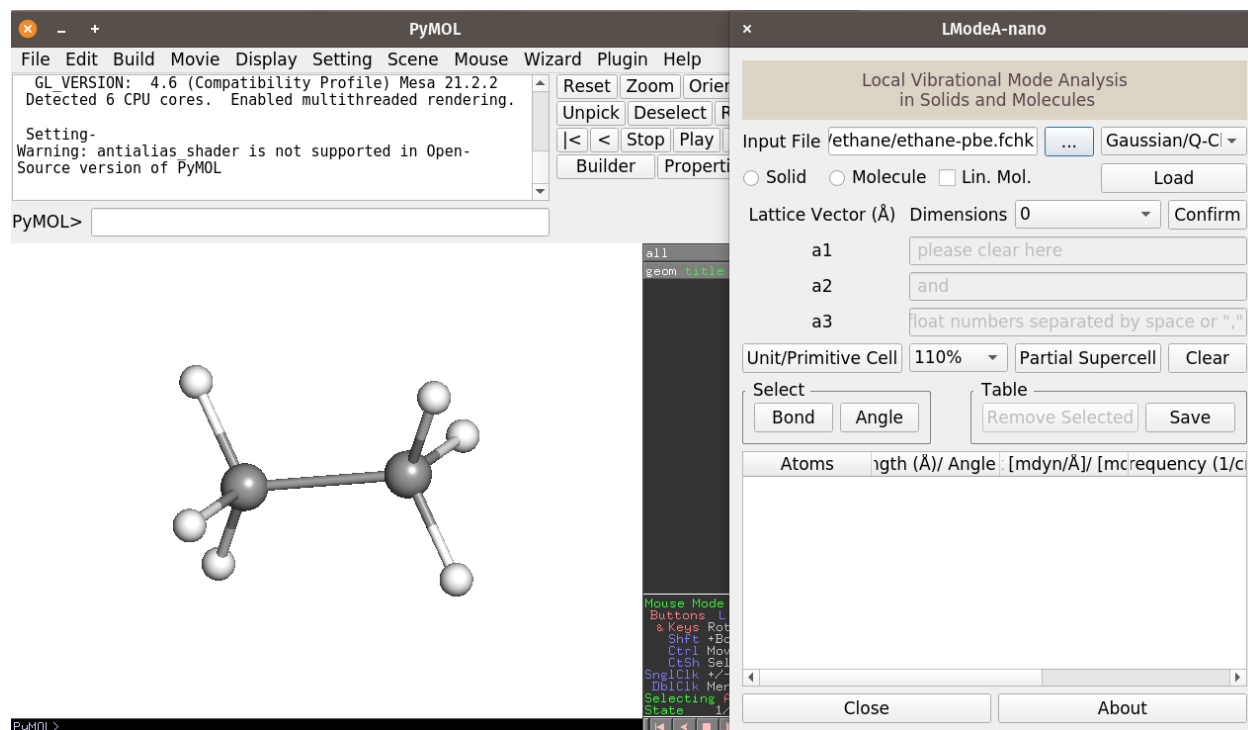
By selecting the CC bond for local mode analysis, the result is

1.3.2 2D Graphene calculated by CRYSTAL17

In the LModeA-nano-main/quickstart/graphene directory, two files were generated by the vibrational frequency calculation with CRYSTAL17 program

- `crystal_21991013.out` - CRYSTAL output file of vibrational analysis
- `HESSFREQ.DAT` - Hessian file generated after vibrational analysis calculation

In the same directory, we compose an input file `crystal.inp` for LModeA-nano



	Atoms	Length (Å)/ Angle (°)	Force constant [mdyn/Å]/ [mdyn·Å/rad^2]	Frequency (1/cm)	Comments
1	C5-C1	1.531	4.075	1073	

@crystal

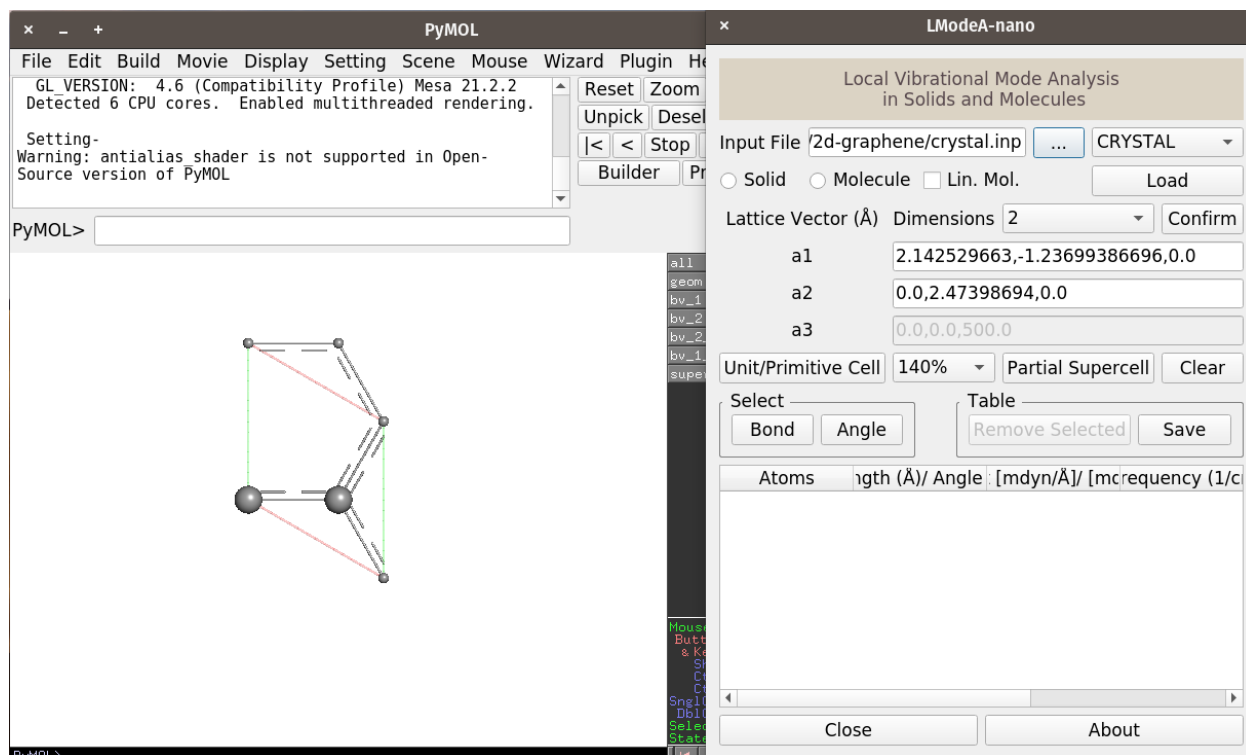
OUTPUT = crystal_21991013.out
HESSFREQ = HESSFREQ.DAT

Following these steps

- open a new PyMOL window and launch LModeA-nano by clicking **Plugin** → **LModeA-nano**
- click the ... (**Browse**) button and select the crystal.inp file
- change the program drop-down list from **VASP** to **CRYSTAL**
- click **Load**
- change **Dimensions** to 2 and click **Confirm** button
- change the supercell size percentage from 110% to 140% and click **Partial Supercell** button

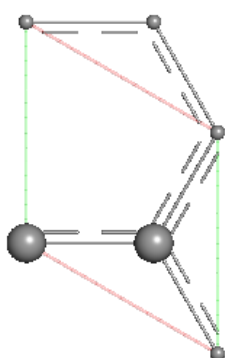
We will see the structure like the following

The PyMOL window shows the graphene primitive cell having two carbon atoms (with larger radii) and other carbon atoms belong to neighboring unit cells.



To do local vibrational mode analysis of CC bonds in this structure, click the **Bond** button in the LModeA-nano window. The interactive wizard may ask for the selection of the first atom of a bond

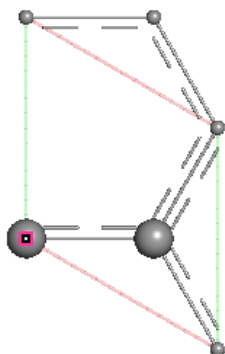
Please click on the first bond atom...



all	A	S	H	L	C
geom title 1/1	A	S	H	L	C
bv_1	A	S	H	L	C
bv_2	A	S	H	L	C
bv_2_b	A	S	H	L	C
bv_1_b	A	S	H	L	C
supercell title	A	S	H	L	C
Bond Selection Wizard					
Reset					
Done					
Close					

Click the left carbon atom in the primitive cell, the wizard prompts the selection of the second atom of a bond

Please click on the second bond atom...



all	A	S	H	L	C		
geom	title	1/1	A	S	H	L	C
bv_1	A	S	H	L	C		
bv_2	A	S	H	L	C		
bv_2_b	A	S	H	L	C		
bv_1_b	A	S	H	L	C		
supercell	title	A	S	H	L	C	
(sele)	A	S	H	L	C		

Bond Selection Wizard

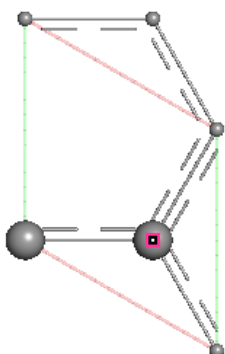
Reset

Done

Close

Click the right carbon atom in the primitive cell, the wizard prompts clicking **Done** button in the wizard menu on the right-hand-side.

Bond selected, please click "Done" button.



all	A	S	H	L	C		
geom	title	1/1	A	S	H	L	C
bv_1	A	S	H	L	C		
bv_2	A	S	H	L	C		
bv_2_b	A	S	H	L	C		
bv_1_b	A	S	H	L	C		
supercell	title	A	S	H	L	C	
(sele)	A	S	H	L	C		

Bond Selection Wizard

Reset

Done

Close

Afterwards, the local mode analysis result for this CC bond is shown in the table region of the LModeA-nano GUI window.

	Atoms	Length (Å)/ Angle (°)	Force constant [mdyn/Å]/ [mdyn·Å/rad^2])	Frequency (1/cm)	Comments
1	C2-C1	1.4284	9.137	1607	

Users are encouraged to try to select a different CC bond in this example for local mode analysis, the result is expected

to be the same as for C1-C2 due to symmetry in graphene structure.

1.4 Prerequisite Knowledge

When using LModeA-nano, you are expected to know

- some basics about PyMOL
- how to conduct structure optimization and vibrational frequency calculations with solid-state modeling packages (e.g. VASP) or/and quantum chemistry modeling packages (e.g. Gaussian)
- general idea of the local vibrational mode theory
- some basics about [Phonopy](#) package

1.5 Local Vibrational Mode Theory

The *local vibrational mode theory* was originally developed by Konkoli and Cremer in 1998 [1-4] and it describes the theoretical *adiabatic internal modes* (later known as *local vibrational modes*) derived from the experimentally measurable *normal vibrational modes* [5] in molecular structures by solving the mass-decoupled Euler–Lagrange equations.

A local vibrational mode associated with an internal coordinate parameter q_n is defined as the infinitesimal displacement of this internal coordinate, followed by the relaxation of all other atoms in this molecular structure.

Like normal vibrational modes having their corresponding frequencies and force constants, a local vibrational mode has its local mode force constant k_n^a (also known as *relaxed force constant*) and frequency ω_n^a . For the local vibrational mode associated with a bond length, the local mode force constant corresponds to the *local stretching force constant* describes the intrinsic strength of this chemical bond of interest. k_n^a as a unique measure of bond strength as an alternative to the bond (dissociation) energy of covalent bond or binding energy of non-covalent interaction has its advantages over the latter bond strength measure.

In 2019, the *local vibrational mode theory* was extended from molecular systems to periodic systems of one through three dimensions (1-3D) in periodicity [6]. The local vibrational mode associated with an internal coordinate q_n is defined as the vibration driven by this internal coordinate in all primitive cells while relaxing all other parts of the periodic system. This theoretical extension enables the quantification of chemical bond strengths in solids and molecules in a head-to-head fashion.

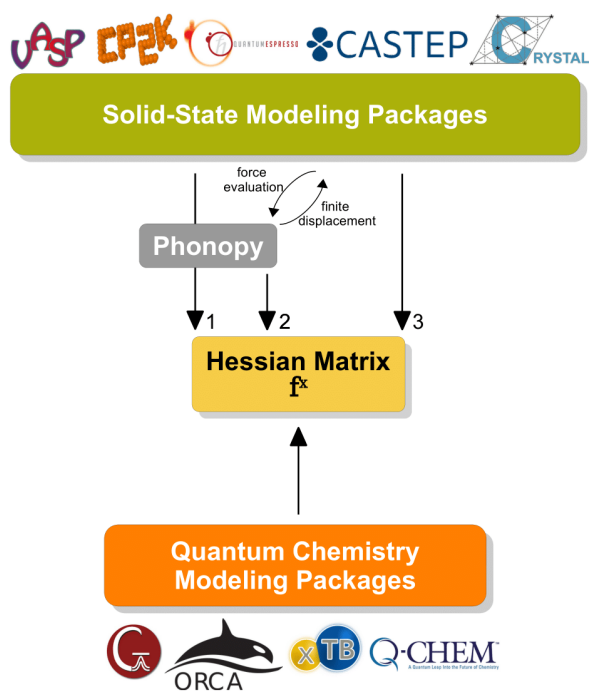
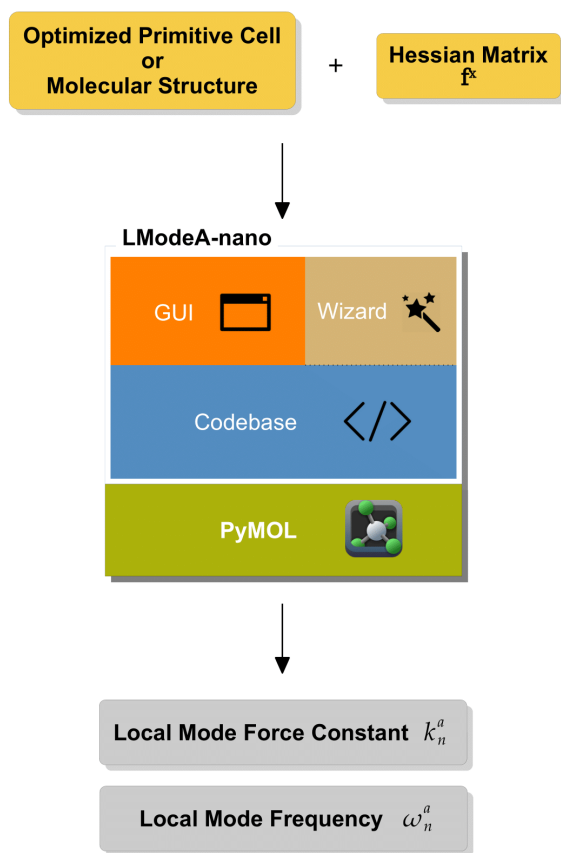
So far, the local vibrational mode theory as a powerful tool has been applied to study both covalent bonds and non-covalent interactions including hydrogen, halogen, pnictogen, chalcogen and tetrel bonding. Interested readers may check a review article by Kraka *et al.* [7].

Necessary input information needed for calculating the local vibrational modes in molecules or solids includes

- optimized molecular structure or primitive cell structure
- Hessian matrix corresponding to the optimized structure

Hessian matrix collecting the second-order energy derivatives with regard to Cartesian coordinates is necessary for vibrational frequency calculations in molecules and solids.

Next, tutorials will be given on how to use LModeA-nano with the input data generated by *solid-state* and *quantum chemistry* modeling packages.



References

1. 26. Konkoli, D. Cremer, *A New Way of Analyzing Vibrational Spectra. I. Derivation of Adiabatic Internal Modes*, Int. J. Quantum Chem. 67, 1 (1998)
2. 26. Konkoli, J. A. Larsson, D. Cremer, *A New Way of Analyzing Vibrational Spectra. II. Comparison of Internal Mode Frequencies*, Int. J. Quantum Chem. 67, 11 (1998)
3. 26. Konkoli, D. Cremer, *A New Way of Analyzing Vibrational Spectra. III. Characterization of Normal Vibrational Modes in terms of Internal Vibrational Modes*, Int. J. Quantum Chem. 67, 29 (1998)
4. 26. Konkoli, J. A. Larsson, D. Cremer, *A New Way of Analyzing Vibrational Spectra. IV. Application and Testing of Adiabatic Modes within the Concept of the Characterization of Normal Modes*, Int. J. Quantum Chem. 67, 41 (1998)
5. E. B. Wilson, J. C. Decius, P. C. Cross, *Molecular Vibrations: The Theory of Infrared and Raman Vibrational Spectra*, Dover Publications: Mineola, NY (2012)
6. Y. Tao, W. Zou, D. Sethio, et al., *In Situ Measure of Intrinsic Bond Strength in Crystalline Structures: Local Vibrational Mode Theory for Periodic Systems*, J. Chem. Theory Comput. 15, 1761 (2019)
7. E. Kraka, W. Zou, Y. Tao, *Decoding Chemical Information from Vibrational Spectroscopy Data: Local Vibrational Mode Theory*, WIREs: Comput. Mol. Sci. 10, e1480 (2020)

1.6 Solid-state Modeling Packages

To obtain the Hessian matrix from some solid-state modeling packages, the latest version of [Phonopy](#) package needs to be installed first.

Note: Although the following examples are on the calculation of solid systems with different solid-state modeling packages, this does not mean that molecular systems calculated by these packages cannot be analyzed by LModeA-nano.

In fact, molecular systems calculated by solid-state modeling packages can be analyzed in the same way as solids except that the **Dimensions** needs to be set to 0 in LModeA-nano.

1.6.1 VASP

Tested with VASP 5.4.4.

Method 1

The first step is cell optimization. For 3D solid system, an example INCAR file is

```
PREC = Accurate

GGA = MK
LUSE_VDW = .TRUE.
PARM1 = 0.1234
PARM2 = 0.711357
```

(continues on next page)

(continued from previous page)

```
Zab_vdW = -1.8867
AGGAC = 0.000
LASPH = .TRUE.

IBRION = 1
NSW = 3000
NELMIN = 5
ISIF = 3
ENCUT = 1000.000000
EDIFF = 1.000000e-08
EDIFFG = -1.000000e-08
ISMear = 0
SIGMA = 0.1
IALGO = 38
LREAL = .FALSE.
ADDGRID = .TRUE.
LWAVE = .FALSE.
LCHARG = .FALSE.
NPAR = 4
```

After the cell optimization (ISIF=3) of a 3D solid structure, the vibrational analysis with VASP needs to be performed.

An example INCAR file for vibrational analysis is

```
PREC = Accurate

GGA = MK
LUSE_VDW = .TRUE.
PARM1 = 0.1234
PARM2 = 0.711357
Zab_vdW = -1.8867
AGGAC = 0.000
LASPH = .TRUE.

IBRION = 5
NFREE=2
POTIM=0.005
NELMIN = 5
ISIF = 3

ENCUT = 1000.000000

EDIFF = 1.000000e-08
EDIFFG = -1.000000e-08
ISMear = 0
SIGMA = 1.000000e-02
IALGO = 38
LREAL = .FALSE.
ADDGRID = .TRUE.
LWAVE = .FALSE.
LCHARG = .FALSE.
```

The most critical parameters are

- IBRION=5 - finite difference method to calculate the Hessian matrix without considering symmetry
- NFREE=2 - use central differences
- POTIM=0.005 - step size for finite difference in Å

When the vibrational analysis calculation is done, check the lowest frequencies by

```
$ grep cm OUTCAR | tail -n 6
115 f = 0.899811 THz 5.653679 2PiTHz 30.014464 cm-1 3.721321 meV
116 f = 0.729527 THz 4.583752 2PiTHz 24.334394 cm-1 3.017081 meV
117 f = 0.483556 THz 3.038272 2PiTHz 16.129693 cm-1 1.999828 meV
118 f/i= 0.013493 THz 0.084780 2PiTHz 0.450086 cm-1 0.055804 meV
119 f/i= 0.036011 THz 0.226262 2PiTHz 1.201188 cm-1 0.148928 meV
120 f/i= 0.056606 THz 0.355667 2PiTHz 1.888177 cm-1 0.234104 meV
```

Such result shows that the optimization is successful because the last three frequencies (close to zero cm^{-1}) correspond to the translation modes of the whole primitive unit cell and the remaining frequencies are all positive.

In the same directory of OUTCAR, another output file `vasprun.xml` is also generated. Run:

```
$ phonopy --fc vasprun.xml
...
FORCE_CONSTANTS has been created from vasprun.xml.
...
```

Two files are needed for later use

- POSCAR - input structure for vibrational analysis calculation
- FORCE_CONSTANTS - Hessian matrix extracted by Phonopy from `.xml` file

Compose an input file `vasp.inp` for LModeA-nano

```
@VASP
POSCON = POSCAR
FCPHONOPY = FORCE_CONSTANTS
```

Following these steps

- open a new PyMOL window and launch LModeA-nano by clicking **Plugin** → **LModeA-nano**
- click the ... (**Browse**) button and select the `vasp.inp` file
- click **Load**
- change **Dimensions** to 3 and click **Confirm** button
- change the supercell size percentage from 110% to 140% or other values, and click **Partial Supercell** button

Then local mode analysis can be performed as described in [quickstart examples](#).

Method 2 - For large cells

For primitive unit cells having more than 70 atoms, the calculation of Hessian matrix by finite difference method can be time-consuming because it requires 6N force evaluation steps (N is number of atoms in the cell). The above *Method 1* performs all force evaluations on a single computational node. If there is a computational cluster, the following method is faster because it may distribute the force evaluations to different nodes.

After the cell optimization (ISIF=3) of a 3D solid structure, save the CONTCAR file as POSCAR in a new directory.

Run Phonopy

```
$ phonopy -d --dim="1 1 1" --pm --amplitude 0.005
...
Creating displacements
Plus Minus displacement: full plus minus directions
Settings:
Supercell: [1 1 1]
...
```

A series of displaced unit cell structure are generated: POSCAR-001, POSCAR-002, . . . Each structure will have its forces evaluated in a single VASP calculation.

The INCAR file for force evaluation can look like

```
PREC = Accurate

GGA = MK
LUSE_VDW = .TRUE.
PARM1 = 0.1234
PARM2 = 0.711357
Zab_vdW = -1.8867
AGGAC = 0.000
LASPH = .TRUE.

IBRION = -1
NELMIN = 5
ISIF = 3

ENCUT = 1000.000000

EDIFF = 1.000000e-08
EDIFFG = -1.000000e-08
ISMear = 0
SIGMA = 1.000000e-02
IALGO = 38
LREAL = .FALSE.
ADDGRID = .TRUE.
LWAVE = .FALSE.
LCHARG = .FALSE.
```

It is recommended to create a separate directory for each structure

```
for id in POSCAR-*
do
  mkdir disp-${id}
```

(continues on next page)

(continued from previous page)

```
cp ${id} disp-${id}/POSCAR
cp POTCAR KPOINTS INCAR disp-${id}/
done
```

When submitting the VASP calculations, each computational node can handle a few structures, like

```
for i in $(seq -f "%03g" 1 100)
do
  echo disp-POSCAR-${i}
  cd disp-POSCAR-${i}
  vasp_std > vasp-${i}.out
  cd ..
done
```

`seq -f "%03g" 1 100` means to loop over 001 to 100.

After the force evaluation of all displaced structures are finished, create `FORCE_SET` using Phonopy

```
$ phonopy -f disp-POSCAR-{001..xxx}/vasprun.xml
```

`xxx` is the total number of displaced structures

Then use Phonopy to create `FORCE_CONSTANTS` file

```
$ phonopy --dim="1 1 1" --writefc
```

With `POSCAR` and `FORCE_CONSTANTS` files, the local mode analysis can be performed with LModeA-nano like in [Method 1](#).

1.6.2 CP2K

Tested with CP2K 8.2. The input file template can be generated by Multiwfn package.

Method 1

The input file for cell optimization of 3D solid is like

```
#Generated by Multiwfn
&GLOBAL
  PROJECT k2reh9-refine
  PRINT_LEVEL LOW
  RUN_TYPE CELL_OPT
&END GLOBAL

&FORCE_EVAL
  METHOD Quickstep
  &SUBSYS
    &CELL
      A    9.691    0.000    0.000
      B   -4.810    8.413    0.000
      C    0.000    0.000    5.706
      PERIODIC XYZ #Direction of applied PBC (geometry aspect)
```

(continues on next page)

(continued from previous page)

&END CELL			
&COORD			
K	-1.2558488724	2.2006561653	2.8528315416
K	3.6056710442	6.2147826995	2.8528315383
K	2.5337691322	0.0019792003	2.8528315571
K	-2.8321552830	4.9560287939	0.0000000311
K	2.0066157818	3.4586295569	-0.0000000269
K	5.7081521839	0.0010789324	0.0000000278
Re	4.8571785554	2.8039248183	2.8528315499
Re	0.0234382187	5.6083829765	2.8528315526
Re	-0.0000638786	-0.0001099606	-0.0000000055
H	6.0270652495	2.5945198607	4.0766192575
H	4.4539534701	3.9229323799	4.0757259845
H	4.0894727226	1.8947602744	4.0751317071
H	-0.3848103609	4.4906671571	1.6305327076
H	4.0894727162	1.8947602632	1.6305314009
H	6.0270652713	2.5945198686	1.6290438503
H	4.4539534568	3.9229323662	1.6299370985
H	-0.7389977868	6.5200724154	4.0766186544
H	1.1950192311	5.8137176574	4.0757278734
H	1.6781788362	-0.0039857278	-0.0000000099
H	4.0388741329	6.9614774107	-0.0000000006
H	-0.8363820802	1.4549103043	-0.0000000041
H	8.4818032881	0.0022916350	1.1978086117
H	0.6071583592	1.0465002108	4.5092456422
H	-4.2077347386	7.3645633109	4.5078562001
H	-4.2077347299	7.3645632719	1.1978068819
H	0.6071583649	1.0465001732	1.1964174406
H	8.4818032732	0.0022916783	4.5078544943
H	-1.6284528162	5.3254766136	2.8528315589
H	0.6034005602	7.1805921593	2.8528315404
H	1.0952417964	4.3204237909	2.8528315625
H	3.2070901557	3.0951493757	2.8528315485
H	5.9342251097	4.0877344629	2.8528315420
H	5.4314516435	1.2294438796	2.8528315643
H	-0.3848103509	4.4906671776	4.0751304240
H	-0.7389977923	6.5200723974	1.6290444364
H	1.1950192243	5.8137176406	1.6299352197
&END COORD			
&KIND K			
ELEMENT K			
BASIS_SET DZVP-MOLOPT-SR-GTH			
POTENTIAL GTH-PBE			
&END KIND			
&KIND Re			
ELEMENT Re			
BASIS_SET DZVP-MOLOPT-SR-GTH			
POTENTIAL GTH-PBE			
&END KIND			
&KIND H			
ELEMENT H			
BASIS_SET DZVP-MOLOPT-SR-GTH			

(continues on next page)

(continued from previous page)

```

    POTENTIAL GTH-PBE
&END KIND
&END SUBSYS

&DFT
    BASIS_SET_FILE_NAME  BASIS_MOLOPT
    POTENTIAL_FILE_NAME  POTENTIAL
#   WFN_RESTART_FILE_NAME k2reh9-opt-RESTART.wfn
    CHARGE      0 #Net charge
    MULTIPLICITY 1 #Spin multiplicity
&KPOINTS
    SCHEME MONKHORST-PACK 5 5 8
    SYMMETRY F #If using symmetry to reduce the number of k-points
&END KPOINTS
&QS
    EPS_DEFAULT 1E-10 #This is default. Set all EPS_xxx to values such that the energy_
↳will be correct up to this value
&END QS
&POISSON
    PERIODIC XYZ #Direction(s) of PBC for calculating electrostatics
    PSOLVER PERIODIC #The way to solve Poisson equation
&END POISSON
&XC
    &XC_FUNCTIONAL PBE
    &PBE
        PARAMETRIZATION REVPBE
    &END PBE
&END XC_FUNCTIONAL
&END XC
&MGRID
    CUTOFF 825
    REL_CUTOFF 75
&END MGRID
&SCF
    MAX_SCF 128
    EPS_SCF 1.0E-06 #Convergence threshold of density matrix during SCF
#   SCF_GUESS RESTART #Use wavefunction from WFN_RESTART_FILE_NAME file as initial_
↳guess
    &DIAGONALIZATION
        ALGORITHM STANDARD #Algorithm for diagonalization. DAVIDSON is faster for large_
↳systems
    &END DIAGONALIZATION
    &MIXING #How to mix old and new density matrices
        METHOD BROYDEN_MIXING #PULAY_MIXING is also a good alternative
        ALPHA 0.4 #Default. Mixing 40% of new density matrix with the old one
        NBROYDEN 8 #Default is 4. Number of previous steps stored for the actual mixing_
↳scheme
    &END MIXING
&PRINT
    &RESTART #Use "&RESTART OFF" can prevent generating wfn file
        BACKUP_COPIES 0 #Maximum number of backup copies of wfn file
    &END RESTART

```

(continues on next page)

(continued from previous page)

```

&END PRINT
&END SCF
&END DFT
STRESS_TENSOR ANALYTICAL
&END FORCE_EVAL

&MOTION
&CELL_OPT
  MAX_ITER 250 #Maximum number of geometry optimization
  EXTERNAL_PRESSURE 1.01325 #External pressure for cell optimization (bar)
  CONSTRAINT NONE #Can be e.g. Z, XY to fix corresponding cell length
  KEEP_ANGLES F #If T, then cell angles will be kepted
  KEEP_SYMMETRY F #If T, crystal symmetry will be kepted, and symmetry should be
  ↪ specified in &CELL
  TYPE DIRECT_CELL_OPT #Geometry and cell are optimized at the same time. Can also be
  ↪ GEO_OPT, MD
  #The following thresholds of optimization convergence are the default ones
  MAX_DR 0.000060 #Maximum geometry change
  RMS_DR 0.000040 #RMS geometry change
  MAX_FORCE 0.000015 #Maximum force
  RMS_FORCE 0.000010 #RMS force
  PRESSURE_TOLERANCE 100 #Pressure tolerance (w.r.t EXTERNAL_PRESSURE)
  OPTIMIZER BFGS #Can also be CG (more robust for difficult cases) or LBFGS
  &BFGS
    TRUST_RADIUS 0.2 #Trust radius (maximum stepsize) in Angstrom
  # RESTART_HESSIAN T #If read initial Hessian, uncomment this line and specify the
  ↪ file in the next line
  # RESTART_FILE_NAME to_be_specified
  &END BFGS
&END CELL_OPT
&PRINT
&RESTART
  BACKUP_COPIES 0 #Maximum number of backing up restart file
&END RESTART
&RESTART_HISTORY
  &EACH
    CELL_OPT 0 #How often a history .restart file is generated, 0 means never
  &END EACH
&END RESTART_HISTORY
&CELL
&END CELL
&END PRINT
&END MOTION

```

With the optimized cell structure, the vibrational analysis is required

```

#Generated by Multiwfn
&GLOBAL
  PROJECT k2-refine-freq
  PRINT_LEVEL MEDIUM

```

(continues on next page)

(continued from previous page)

```

RUN_TYPE VIBRATIONAL_ANALYSIS
&END GLOBAL

&FORCE_EVAL
  METHOD Quickstep
  &SUBSYS
    &CELL
      A    9.7154601050      0.0000000000      0.0000000000
      B   -4.8583032251      8.4134784197      0.0000000000
      C   -0.0000048135     -0.0000078762      5.6945465926
    PERIODIC XYZ #Direction of applied PBC (geometry aspect)
  &END CELL
  &COORD
    K    -1.2717444316      2.2034502965      2.8470104286
    K     3.5860479359      6.2116264326      2.8470414297
    K     2.5441673793      0.0005069845      2.8470105500
    K    -2.8664982103      4.9640103204     -0.0002691372
    K     1.9919809386      3.4504437696     -0.0002481916
    K     5.7322486247     -0.0000606935     -0.0002691208
    Re    4.8578753089      2.8054541385      2.8470302019
    Re    0.0003345425      5.6097335267      2.8470301311
    Re    0.0003406364      0.0005363186     -0.0002407693
    H     6.0318408651      2.6225498479      4.0708725666
    H     4.4291201342      3.9136919383      4.0706909878
    H     4.1125879533      1.8801461820      4.0708571241
    H    -0.4282825066      4.5016125249      1.6232100820
    H     4.1125863706      1.8801513315      1.6232012617
    H     6.0318388962      2.6225540649      1.6231796732
    H     4.4291142103      3.9136997301      1.6233758117
    H    -0.7451026988      6.5349089400      4.0708601931
    H     1.1744732842      5.7926227251      4.0707068918
    H     1.6786597913      0.0005874087     -0.0002624812
    H     4.0184027906      6.9605294583     -0.0002696641
    H    -0.8388731978      1.4539493631     -0.0002624278
    H     8.5073111255      0.0005134857      1.1978276666
    H     0.6045199441      1.0470807705      4.4961739186
    H    -4.2536730257      7.3674658865      4.4962388277
    H    -4.2536811640      7.3674669181      1.1978123072
    H     0.6045200337      1.0470806825      1.1978988291
    H     8.5073057336      0.0005201478      4.4962231782
    H    -1.6564785794      5.3554871447      2.8470329264
    H     0.6084571955      7.1717438292      2.8470366227
    H     1.0487876716      4.3019244622      2.8470389912
    H     3.2010232060      3.0594336945      2.8470390703
    H     5.9064519076      4.1131982559      2.8470367529
    H     5.4661799457      1.2435429741      2.8470328975
    H    -0.4282878791      4.5016116953      4.0708480056
    H    -0.7450971903      6.5349082052      1.6231930653
    H     1.1744841476      5.7926214329      1.6233587569
  &END COORD
  &KIND K
  ELEMENT K

```

(continues on next page)

(continued from previous page)

```

BASIS_SET DZVP-MOLOPT-SR-GTH
POTENTIAL GTH-PBE
&END KIND
&KIND Re
ELEMENT Re
BASIS_SET DZVP-MOLOPT-SR-GTH
POTENTIAL GTH-PBE
&END KIND
&KIND H
ELEMENT H
BASIS_SET DZVP-MOLOPT-SR-GTH
POTENTIAL GTH-PBE
&END KIND
&END SUBSYS

&DFT
BASIS_SET_FILE_NAME BASIS_MOLOPT
POTENTIAL_FILE_NAME POTENTIAL
# WFN_RESTART_FILE_NAME k2-freq-template-RESTART.wfn
CHARGE 0 #Net charge
MULTIPLICITY 1 #Spin multiplicity
&KPOINTS
SCHEME MONKHORST-PACK 5 5 8
SYMMETRY F #If using symmetry to reduce the number of k-points
&END KPOINTS
&QS
EPS_DEFAULT 1E-10 #This is default. Set all EPS_xxx to values such that the energy_
↪will be correct up to this value
&END QS
&POISSON
PERIODIC XYZ #Direction(s) of PBC for calculating electrostatics
PSOLVER PERIODIC #The way to solve Poisson equation
&END POISSON
&XC
&XC_FUNCTIONAL PBE
&PBE
PARAMETRIZATION REVPBE
&END PBE
&END XC_FUNCTIONAL
&END XC
&MGRID
CUTOFF 825
REL_CUTOFF 75
&END MGRID
&SCF
MAX_SCF 128
EPS_SCF 1.0E-07 #Convergence threshold of density matrix during SCF
# SCF_GUESS RESTART #Use wavefunction from WFN_RESTART_FILE_NAME file as initial_
↪guess
&DIAGONALIZATION
ALGORITHM STANDARD #Algorithm for diagonalization. DAVIDSON is faster for large_
↪systems

```

(continues on next page)

(continued from previous page)

```

&END DIAGONALIZATION
&MIXING #How to mix old and new density matrices
  METHOD BROYDEN_MIXING #PULAY_MIXING is also a good alternative
  ALPHA 0.4 #Default. Mixing 40% of new density matrix with the old one
  NBROYDEN 8 #Default is 4. Number of previous steps stored for the actual mixing.
→ scheme
&END MIXING
&END SCF
&PRINT
  &MOMENTS
  &END MOMENTS
&END PRINT
&END DFT
&END FORCE_EVAL

&MOTION
&PRINT
  &RESTART
    BACKUP_COPIES 0 #Maximum number of backing up restart file
  &END RESTART
&END PRINT
&END MOTION
&VIBRATIONAL_ANALYSIS
  DX 0.008 #Step size of finite difference. This is default (Bohr)
  NPROC_REP 1 #Number of processors to be used per replica. This is default
  TC_PRESSURE 101325 #1 atm. Pressure for calculate thermodynamic data (Pa)
  TC_TEMPERATURE 298.15 #Temperature for calculate thermodynamic data (K)
  THERMOCHEMISTRY #Print thermochemistry information (only correct for gas molecule!)
  INTENSITIES F #Calculate IR intensities
&PRINT
  &HESSIAN high
  &END HESSIAN
  &MOLDEN_VIB #Output .mol (Molden file) for visualization vibrational modes
  &END MOLDEN_VIB
&END PRINT
&END VIBRATIONAL_ANALYSIS

```

Critical parameters include

- DX 0.008 - this specifies the step size of finite difference
- HESSIAN high - this prints the Hessian matrix in the CP2K output file

After the vibrational analysis is done, check the vibrational frequency values to make sure the structure is at a local minimum energetically.

Compose an input file cp2k.inp for LModeA-nano

```

@cp2k

mode = native
OUTPUT = k2reh9-refine-freq.log
INPUT = k2reh9-refine-freq.inp

```

The INPUT and OUTPUT specify the input and output files of vibrational analysis calculation.

Follow these steps to perform local mode analysis with LModeA-nano

- open a new PyMOL window and launch LModeA-nano by clicking **Plugin** → **LModeA-nano**
- click the ... (**Browse**) button and select the `cp2k.inp` file
- change the program drop-down list from **VASP** to **CP2K**
- click **Load**
- change **Dimensions** to 3 and click **Confirm** button
- change the supercell size percentage from 110% to 140% or other values, and click **Partial Supercell** button

Then local mode analysis can be performed as described in [quickstart examples](#).

Method 2 - For large cells

Similar to *Method 2 with VASP*, the general idea of this method is ask Phonopy package to generate the CP2K input files for the displaced structures starting from an already optimized structure. Then force evaluations are performed on these displaced structures in order to obtain Hessian.

A template CP2K input file having the optimized cell structure is needed, and the format of this input should look like the following. Otherwise, Phonopy package cannot parse it correctly. More examples of such CP2K input can be found in the Phonopy repo on GitHub: [NaCl](#) and [Si](#).

```
! whether to do kpoints or not
! without kpoints it will be significantly faster,
! but some crossings in the band structure will be missing
@SET WITH_KP yes

! set the following to yes to do a cell optimization instead of energy/force calculation
! do NOT enable this when using this input file as template for phonopy!
@SET DO_CELLOPT no

&GLOBAL
@IF $DO_CELLOPT == yes
  PROJECT solid-cellopt
  RUN_TYPE CELL_OPT
@ENDIF
@IF $DO_CELLOPT == no
  PROJECT na2reh9-phonopy
  RUN_TYPE ENERGY_FORCE
@ENDIF
  PRINT_LEVEL MEDIUM
&END GLOBAL

&FORCE_EVAL
  METHOD Quickstep
  &DFT
    BASIS_SET_FILE_NAME BASIS_MOLOPT
    POTENTIAL_FILE_NAME POTENTIAL

    &QS
      EPS_DEFAULT 1e-10

    ! We are using GAPW instead of the default GPW to prevent numerical
```

(continues on next page)

(continued from previous page)

```

! noise due to the finite grid size in the integration schemes in CP2K
! together with the small displacements used to determine the forces.
! Alternatively one could increase the CUTOFF to >5000 (depends on basis set)
! to ensure that the higher exponents in the basis sets are well represented on
↳the grid.
&END QS

&POISSON
  PERIODIC XYZ
  PSOLVER PERIODIC
&END POISSON

&SCF
  EPS_SCF 1e-07
  MAX_SCF 80
  &DIAGONALIZATION
    ALGORITHM STANDARD
  &END DIAGONALIZATION
  &MIXING #How to mix old and new density matrices
    METHOD BROYDEN_MIXING #PULAY_MIXING is also a good alternative
    ALPHA 0.4 #Default. Mixing 40% of new density matrix with the old one
    NBROYDEN 8 #Default is 4. Number of previous steps stored for the actual
↳mixing scheme
  &END MIXING
&END SCF

@if $WITH_KP == yes
  &KPOINTS
    SCHEME MONKHORST-PACK 5 5 8
    SYMMETRY .FALSE.
  &END KPOINTS
@endif

&XC
  &XC_FUNCTIONAL PBE
  &PBE
    PARAMETRIZATION REVPBE
  &END PBE
&END XC_FUNCTIONAL
&END XC

&MGRID
  REL_CUTOFF 80
  CUTOFF 900
&END MGRID
&END DFT

&SUBSYS
  &CELL
  A 8.8901326513 0.0000000000 0.0000000000
  B -4.4739104629 7.6824375325 0.0000000000
  C 0.0000001000 0.0000003943 5.2483716421

```

(continues on next page)

(continued from previous page)

```

      PERIODIC XYZ
&END CELL
&COORD
Na      -1.1859069865      2.0379623261      2.6241869010
Na      3.2449119045      5.6448521398      2.6241867646
Na      2.3578700048      0.0007892838      2.6241868726
Na      -2.6256486318      4.5068346666      0.0000001680
Na      1.8251693178      3.1750730554      0.0000001576
Na      5.2158593380     -0.0009231392      0.0000001673
Re      4.4364564557      2.5642868258      2.6242178317
Re     -0.0167453884      5.1242663051      2.6242177424
Re      0.0017170475      0.0030111053      0.0000094252
H       5.6222385887      2.4929720119      3.8444638123
H       3.9052360919      3.6232613245      3.8472931155
H       3.7846319842      1.5754147357      3.8473354516
H      -0.5432325121      4.0633761049      1.4010781033
H       3.7846337784      1.5754183851      1.4010976360
H       5.6222353787      2.4929729080      1.4039700228
H       3.9052378288      3.6232592016      1.4011411061
H      -0.6751167153      6.1130464337      3.8444723455
H       1.1657297010      5.1981428489      3.8472647704
H       1.6763457738      0.0076515368      0.0000010890
H       3.5835080375      6.2338251923      0.0000016665
H      -0.8370053879      1.4524719986      0.0000010868
H       7.6815358956      0.0008872303      1.1958253823
H       0.6038026414      1.0503904668      4.0506195218
H      -3.8649471720      6.6385181149      4.0525569576
H      -3.8649468269      6.6385180090      1.1958373681
H       0.6038029996      1.0503908521      1.1977742281
H       7.6815360048      0.0008870488      4.0525690768
H      -1.6822864070      5.0230528750      2.6242153250
H       0.7327983274      6.6152712210      2.6242163261
H       0.9014162231      3.7301543776      2.6242158401
H       2.7696859375      2.6561823231      2.6242158698
H       5.3476415856      3.9623777672      2.6242164032
H       5.1871788616      1.0740919399      2.6242154475
H      -0.5432348504      4.0633727533      3.8473548145
H      -0.6751143293      6.1130441536      1.4039612322
H       1.1657270304      5.1981432445      1.4011693676
&END COORD
&KIND Na
  ELEMENT Na
    BASIS_SET DZVP-MOLOPT-SR-GTH # use an AE basis optimized for solids, any other
↳(matching the pseudo) will work too, though
    POTENTIAL GTH-PBE # one could also use a pseudopotentials (non-AE)
    ! while not strictly required, this should make the initial guess more accurate.
↳(Na+ Cl-)
&END KIND
&KIND Re
  ELEMENT Re
    BASIS_SET DZVP-MOLOPT-SR-GTH # use an AE basis optimized for solids, any other
↳(matching the pseudo) will work too, though

```

(continues on next page)

(continued from previous page)

```

        POTENTIAL GTH-PBE # one could also use a pseudopotentials (non-AE)
        ! while not strictly required, this should make the initial guess more accurate.
↪(Na+ Cl-)
        &END KIND
        &KIND H
        ELEMENT H
        BASIS_SET DZVP-MOLOPT-SR-GTH # use an AE basis optimized for solids, any other
↪(matching the pseudo) will work too, though
        POTENTIAL GTH-PBE # one could also use a pseudopotentials (non-AE)
        ! while not strictly required, this should make the initial guess more accurate.
↪(Na+ Cl-)
        &END KIND

&END SUBSYS

STRESS_TENSOR ANALYTICAL

@if $DO_CELLOPT == no
    &PRINT
        &STRESS_TENSOR
            ADD_LAST NUMERIC
            FILENAME stress_tensor
        &END STRESS_TENSOR
        &FORCES
            ADD_LAST NUMERIC
            FILENAME forces
        &END FORCES
    &END PRINT
@endif
&END FORCE_EVAL

@if $DO_CELLOPT == yes
&MOTION
    &CELL_OPT
        KEEP_ANGLES .TRUE.
        MAX_FORCE 1.0E-10
    &END CELL_OPT
&END MOTION
@endif

```

Then run

```
$ phonopy --cp2k -d --dim="1 1 1" --pm --amplitude 0.005 -c na2reh9-template.inp
```

The parameter after `-c` is the above-mentioned CP2K template input file containing the optimized structure. Phonopy generates a series of CP2K input files for force evaluation: `na2reh9-template-supercell-001.inp`, `na2reh9-template-supercell-002.inp`, This step will also generate a `phonopy_disp.yaml` file.

The generated CP2K input can look like

```

# Generated by Phonopy, based on na2reh9-template.inp
# Merged configuration with displacements
&FORCE_EVAL

```

(continues on next page)

(continued from previous page)

```

METHOD QUICKSTEP
STRESS_TENSOR ANALYTICAL
&PRINT
  &FORCES
    ADD_LAST NUMERIC
    FILENAME forces
  &END FORCES
  &STRESS_TENSOR
    ADD_LAST NUMERIC
    FILENAME stress_tensor
  &END STRESS_TENSOR
&END PRINT
&SUBSYS
  &KIND H
    ELEMENT H
    BASIS_SET DZVP-MOLOPT-SR-GTH
    POTENTIAL GTH-PBE
  &END KIND
  &KIND Re
    ELEMENT Re
    BASIS_SET DZVP-MOLOPT-SR-GTH
    POTENTIAL GTH-PBE
  &END KIND
  &KIND Na
    ELEMENT Na
    BASIS_SET DZVP-MOLOPT-SR-GTH
    POTENTIAL GTH-PBE
  &END KIND
  &COORD
    Na -1.1809069865 2.0379623261 2.624186901
    Na 3.2449119044999986 5.6448521398 2.6241867646
    Na 2.3578700047999996 0.0007892838 2.6241868725999993
    Na 6.2644840195 4.5068346666 1.6799999999999997e-07
    Na 1.8251693178000001 3.1750730554 1.576e-07
    Na 0.7419488750999992 7.6815143933 1.673e-07
    Re 4.4364564557000002 2.5642868258 2.6242178317
    Re -0.01674538839999951 5.1242663051000002 2.6242177424
    Re 0.0017170474999999995 0.0030111053000000001 9.425199999999999e-06
    H 5.6222385887 2.4929720119 3.844463812299999
    H 3.9052360918999995 3.6232613245 3.847293115499999
    H 3.7846319841999994 1.5754147357000003 3.847335451599999
    H -0.5432325121 4.0633761049 1.4010781033
    H 3.7846337784 1.5754183851 1.401097636
    H 5.6222353787 2.492972907999999 1.4039700228
    H 3.9052378287999994 3.6232592016 1.4011411060999996
    H -0.6751167152999991 6.1130464337000002 3.8444723455
    H 1.1657297010000001 5.198142848899998 3.8472647704
    H 1.6763457738 0.00765153680000000025 1.089e-06
    H 3.5835080375000002 6.2338251923 1.6665e-06
    H -0.8370053878999996 1.4524719986 1.0867999999999997e-06
    H 7.6815358956 0.00088723030000000002 1.1958253823
    H 0.6038026414 1.0503904668 4.0506195218

```

(continues on next page)

(continued from previous page)

```

H -3.864947171999999 6.6385181149 4.0525569576
H -3.864946826899999 6.638518009 1.1958373681
H 0.6038029995999996 1.0503908521 1.1977742281
H 7.6815360048 0.0008870488 4.052569076799999
H -1.682286407 5.023052875 2.6242153249999998
H 0.7327983273999987 6.615271221 2.624216326099999
H 0.9014162231000011 3.730154377600001 2.6242158401
H 2.7696859375000003 2.6561823231 2.6242158698
H 5.3476415856 3.9623777672 2.6242164032
H 5.1871788616 1.0740919399 2.6242154474999997
H -0.5432348503999994 4.0633727533 3.8473548145
H -0.6751143292999994 6.1130441536 1.4039612321999997
H 1.1657270304000014 5.1981432445 1.4011693676
&END COORD
&CELL
  A 8.8901326513 0.0 0.0
  B -4.4739104629 7.6824375325 0.0
  C 1e-07 3.943e-07 5.2483716421
  PERIODIC XYZ
&END CELL
&END SUBSYS
&DFT
  BASIS_SET_FILE_NAME BASIS_MOLOPT
  POTENTIAL_FILE_NAME POTENTIAL
  &MGRID
    REL_CUTOFF 80.0
    CUTOFF 900.0
  &END MGRID
  &XC
    &XC_FUNCTIONAL PBE
    &PBE
      PARAMETRIZATION REVPBE
    &END PBE
  &END XC_FUNCTIONAL
&END XC
&KPOINTS
  SCHEME MONKHORST-PACK 5 5 8
  SYMMETRY .FALSE.
&END KPOINTS
&SCF
  EPS_SCF 1e-07
  MAX_SCF 80
  &MIXING
    METHOD BROYDEN_MIXING
    ALPHA 0.4
    NBUFFER 8
  &END MIXING
  &DIAGONALIZATION
    ALGORITHM STANDARD
  &END DIAGONALIZATION
&END SCF
&POISSON

```

(continues on next page)

(continued from previous page)

```

        PERIODIC XYZ
        POISSON_SOLVER PERIODIC
    &END POISSON
    &QS
        EPS_DEFAULT 1e-10
    &END QS
    &END DFT
&END FORCE_EVAL
&GLOBAL
    PROJECT_NAME na2reh9-phonopy-supercell-001
    RUN_TYPE ENERGY_FORCE
    PRINT_LEVEL MEDIUM
&END GLOBAL

```

After the force evaluation for these generated input files is completed, it creates a series files containing the force information: na2reh9-phonopy-supercell-001-forces-1_0.xyz, na2reh9-phonopy-supercell-002-forces-1_0.xyz, ...

Make sure that these .xyz files are collected in one directory and this directory should contain phonopy_disp.yaml file.

Run Phonopy to create the FORCE_SETS file

```
$ phonopy --cp2k -f na2reh9-phonopy-supercell-{001..xxx}-forces-1_0.xyz
```

xxx is the total number of displaced structures generated by Phonopy

Then use Phonopy to generate the FORCE_CONSTANTS file

```
$ phonopy --dim="1 1 1" --writefc
```

Compose the input file cp2k.inp for LModeA-nano

```

@cp2k

mode = phonopy
FCPHONOPY = FORCE_CONSTANTS
INPUT = na2reh9-template.inp

```

The INPUT parameter specifies the template CP2K input file containing the optimized structure.

1.6.3 Quantum ESPRESSO

Tested with QE 6.4.1.

There are two ways to do local mode analysis on the results from QE. The first one is to utilize the density functional perturbation theory (DFPT) implemented in QE to obtain the force constant matrix at Gamma point ($q=0$). The other method known as “frozen phonon” is to use finite difference approach to generate a series of force sets in order to construct the Hessian matrix by using Phonopy package.

These two approaches are equivalent in principle. While the DFPT method in QE has limited support for levels of theory (e.g. hybrid density functional), one has to use the finite difference approach if DFPT is not applicable. Therefore, it is recommended to employ the frozen phonon method.

Suppose that users have obtained the optimized primitive cell structure of a solid, the following tutorial skips this cell optimization step.

DFPT

This method includes three steps

1. SCF calculation with `pw.x`
2. DFPT calculation with `ph.x`
3. generate force constant matrix with `q2r.x`

Based on an optimized primitive unit cell structure, prepare an SCF input file `diamond-scf.inp`

```
&CONTROL
  pseudo_dir = '.'
  outdir = './outp'
  calculation = 'scf'
  prefix='carbon'
/

&SYSTEM
 ibrav = 0
  nat = 2
  ntyp = 1
  ecutwfc = 40.0
  ecutrho = 400.0
/

&ELECTRONS
  scf_must_converge = .true.
/

ATOMIC_SPECIES
C 12.0107 C.pbe-n-rrkjus_psl.1.0.0.UPF

CELL_PARAMETERS (angstrom)
 2.522557697  0.0000000000 -0.0000000000
 1.261278849  2.184599048 -0.0000000000
 1.261278849  0.728199683  2.060840365

ATOMIC_POSITIONS (angstrom)
C      3.783836546  2.184599048  1.545801941
C      2.522557698  1.456399366  1.030248516

K_POINTS automatic
 5 5 5 1 1 1
```

The CELL_PARAMETERS and ATOMIC_POSITIONS have to be in the unit of angstrom.

Run DFPT calculations. The program to run this job is PHonon (`ph.x`). The input file needs to specify the same outdir and prefix as the SCF job did. In addition, specify the Gamma point ($q=[0,0,0]$) in the end of the input file.

```
phonons of diamond at Gamma
&inputph
  tr2_ph=1.0d-14,
  epsil=.true.,
```

(continues on next page)

(continued from previous page)

```
prefix='carbon',
fildyn='dynmat.dyn1',
amass(1)=12.0107,
outdir='./outp'
/
0.0 0.0 0.0
```

The `fildyn` specifies the name of the dynamical matrix file as an output.

Create a text file named `dynmat.dyn0`

```
1 1 1
1
```

Run

```
$ cp dynmat.dyn1 dynmat.dyn
```

Prepare the input file for `q2r.x`

```
&input
fildyn='dynmat.dyn',
zasr='simple',
flfrc='diamond.fc'
/
```

In this input file, the `zasr` controls the *acoustic sum rules*. For solid systems, we need to set `zasr` to 'simple' or 'crystal'. For one-dimensional polymer systems, we need to set `zasr` to 'one-dim'. For isolated molecular systems, we need to set `zasr` to 'zero-dim'.

The parameter `flfrc` specifies the name of the interatomic force constant file generated from the calculation of `q2r.x`.

So far, we have obtained necessary data files for local mode analysis

- `diamond-scf.inp` - input file for SCF calculation
- `diamond.fc` - the force constant file specified by `flfrc`

Compose an input file `qe.inp` for local mode analysis

```
@QE
mode = dfpt
GeomIn = diamond-scf.in
q2rFC = diamond.fc
```

Follow these steps to perform local mode analysis with LModeA-nano

- open a new PyMOL window and launch LModeA-nano by clicking **Plugin** → **LModeA-nano**
- click the ... (**Browse**) button and select the `qe.inp` file
- change the program drop-down list from **VASP** to **Quantum ESPRESSO**
- click **Load**
- change **Dimensions** to 3 and click **Confirm** button
- change the supercell size percentage from 110% to 140% or other values, and click **Partial Supercell** button

Then local mode analysis can be performed as described in [quickstart examples](#).

Frozon Phonon

Given an already optimized cell structure, prepare a template QE input file `diamond-temp.in`

```
&CONTROL
  pseudo_dir = '.'
  calculation = 'scf'
  tprnfor = .true.
  tstress = .true.
/

&SYSTEM
  ibrav = 0

  nat = 2
  ntyp = 1
  ecutwfc = 40.0
  ecutrho = 400.0
/

&ELECTRONS
scf_must_converge = .true.
/

ATOMIC_SPECIES
C 12.0107 C.pbe-n-rrkjus_psl.1.0.0.UPF

CELL_PARAMETERS (angstrom)
  2.522557697  0.000000000 -0.000000000
  1.261278849  2.184599048 -0.000000000
  1.261278849  0.728199683  2.060840365

ATOMIC_POSITIONS (crystal)
C  0.749972224  0.749972284  0.750083327
C  0.500027776  0.500027716  0.499916703

K_POINTS automatic
  5 5 5 1 1 1
```

It is important to set `tprnfor` and `tstress` as force evaluation is desired to construct Hessian. In addition, the atomic positions have to be in the crystal format using the fractional coordinates as required by Phonopy later.

Then save upper and lower part as two separate files

`diamond-temp-upper.in` :

```
&CONTROL
  pseudo_dir = '.'
  calculation = 'scf'
  tprnfor = .true.
  tstress = .true.
/
```

(continues on next page)

(continued from previous page)

```
&SYSTEM
 ibrav = 0

  nat = 2
  ntyp = 1
  ecutwfc = 40.0
  ecutrho = 400.0
/

&ELECTRONS
scf_must_converge = .true.
/
```

diamond-temp-lower.in :

```
K_POINTS automatic
5 5 5 1 1 1
```

Use Phonopy to generate displaced structures

```
$ phonopy --qe -c diamond-temp.in -d --dim="1 1 1"
```

This will generate a series of files containing the displaced structures `supercell-001.in`, `supercell-002.in`, ... But these files only contain the structure information. Create complete QE input files with the following script

```
prefix="supercell-"
for a in supercell-*.in
do
  b=${a#$prefix}
  cat diamond-temp-upper.in $a diamond-temp-lower.in > diamond-${b}
done
```

This script can generate QE input files `diamond-001.in`, `diamond-002.in`, ... These input files differ from `diamond-temp.in` only in the structure information.

Run QE `pw.x` calculations with these generated input files, the output files are expected to be `diamond-001.out`, `diamond-002.out`, ...

Use Phonopy to generate the `FORCE_SETS` file

```
$ phonopy --qe -f diamond-{001..xxx}.out
```

`xxx` is the total number of displaced structures.

Generate `FORCE_CONSTANTS` file with Phonopy

```
$ phonopy --writefc
```

Compose a LModeA-nano input file `qe.inp`

```
@QE
mode = phonopy
GeomIn = diamond-temp.in
FCPHONOPY = FORCE_CONSTANTS
```

The remaining step to perform local mode analysis is the same as in *DFPT method*.

1.6.4 CASTEP

Tested with CASTEP 20.11. CASTEP is able to perform phonon (vibrational analysis) calculations with either *DFPT* or *finite difference* method.

Given the already optimized solid structure, prepare the `.cell` input for CASTEP

```
%BLOCK lattice_cart
Ang
2.5333173      0.0000000      0.0000000
1.2666587      2.1939171      0.0000000
1.2666587      0.7313057      2.0684936
%ENDBLOCK lattice_cart

%BLOCK positions_frac
C              0.749996      0.749996      0.750013
C              0.500004      0.500004      0.499987
%ENDBLOCK positions_frac
!
! Choose which pseudopotentials to use
! Either specify external files, or omit to generate a pseudopotential
!
%BLOCK species_pot
C NCP
%ENDBLOCK species_pot
!
! Analyse structure to determine symmetry
!
!symmetry_generate
!
! Specify M-P grid dimensions for electron wavevectors (K-points)
!
kpoint_mp_grid 5 5 5
!
! Offset the grid in the xy plane to include (0,0,0)

%block PHONON_KPOINT_LIST
      0.0   0.0   0.0   1.0           ! Wavevector of phonon(s) to compute ( qx qy qz,
↪weight)
%endblock PHONON_KPOINT_LIST
```

Then prepare the `.param` file for either DFPT

```
IPRINT 3
!
! Task keyword specifies type of calculation
task      : phonon
!
! Treat the system as an insulator
!
fix_occupancy : true
```

(continues on next page)

(continued from previous page)

```
!
! Tune for speed rather than memory conservation.
!
opt_strategy : speed
!
! Don't write unnecessary wavefunction files
!
num_dump_cycles : 0
!
! Choose exchange-correlation functional
!
xc_functional : PBE
!
! Give plane-wave cutoff. Also allowed keywords (COARSE,MEDIUM,FINE,PRECISE)
!
BASIS_PRECISION : FINE
!
! Choose size of FFT grid. 2.0 is fully converged. 1.75 is good enough for LDA
!
grid_scale 2.0
!
! Choose which SCF solver to use ("none" for allbands, "dm" for Density Mixing)
!
elec_method : dm
mixing_scheme : Pulay
mix_charge_amp : 0.6
!
! Convergence tolerance criterion - energy per atom.
!
elec_energy_tol : 1.0e-9 eV
!
! Turn on calculation of stress (off by default)
!
calculate_stress : true

phonon_sum_rule : true
```

or finite difference calculation

```
IPRINT 2
!
! Task keyword specifies type of calculation
task : phonon
PHONON_METHOD : FINITEDIPLACEMENT

!
! Treat the system as an insulator
!
fix_occupancy : true
!
! Tune for speed rather than memory conservation.
```

(continues on next page)

(continued from previous page)

```

!
opt_strategy : speed
!
! Don't write unnecessary wavefunction files
!
num_dump_cycles : 0
!
! Choose exchange-correlation functional
!
xc_functional : PBE
!
! Give plane-wave cutoff. Also allowed keywords (COARSE,MEDIUM,FINE,PRECISE)
!
BASIS_PRECISION : FINE
!
! Choose size of FFT grid. 2.0 is fully converged. 1.75 is good enough for LDA
!
grid_scale 2.0
!
! Choose which SCF solver to use ("none" for allbands, "dm" for Density Mixing)
!
elec_method : dm
mixing_scheme : Pulay
mix_charge_amp : 0.6
!
! Convergence tolerance criterion - energy per atom.
!
elec_energy_tol : 1.0e-9 eV
!
! Turn on calculation of stress (off by default)
!
calculate_stress : true

phonon_sum_rule : true

```

The major output is .castep file. Then compose an input file castep.inp for LModeA-nano.

```

@castep

output = xxxx.castep

```

Follow these steps

- open a new PyMOL window and launch LModeA-nano by clicking **Plugin** → **LModeA-nano**
- click the ... (**Browse**) button and select the qe.inp file
- change the program drop-down list from **VASP** to **CASTEP**
- click **Load**
- change **Dimensions** to 3 and click **Confirm** button
- change the supercell size percentage from 110% to 140% or other values, and click **Partial Supercell** button

Then local mode analysis can be performed as described in [quickstart examples](#).

1.6.5 CRYSTAL

Tested with CRYSTAL17.

To optimize the primitive cell structure, the input file can be

```

graphene
SLAB
1
2.456 2.456 120.00
2
  6  0.0000000000      0.0000000000      0.0000000000
  6  0.3333333333      0.6666666666      0.0000000000
OPTGEOM
TOLDEG
0.0000050
TOLDEX
0.000010
TOLDEE
1E-9
END
END
6 6
0 0 5 2 1
  1238.40169380      0.0054568832082
  186.290049920      0.0406384092110
  42.2511763460      0.1802559388800
  11.6765579320      0.4631512175500
  3.59305064820      0.4408717331400
0 0 1 2 1
  0.46866670000      1.000000000000000
0 0 1 0 1
  0.23547146000      1.000000000000000
0 2 3 2 1
  9.46809706210      0.0383878717280
  2.01035451420      0.2111702511200
  0.54771004707      0.5132817211400
0 2 1 0 1
  0.21046214000      1.000000000000000
0 3 1 0 1
  0.52857792000      1.000000000000000
8 5
0 0 6 2.0 1.0
  0.5484671660D+04  0.1831074430D-02
  0.8252349460D+03  0.1395017220D-01
  0.1880469580D+03  0.6844507810D-01
  0.5296450000D+02  0.2327143360D+00
  0.1689757040D+02  0.4701928980D+00
  0.5799635340D+01  0.3585208530D+00
0 1 3 6.0 1.0
  0.1553961625D+02 -0.1107775495D+00  0.7087426823D-01
  0.3599933586D+01 -0.1480262627D+00  0.3397528391D+00
  0.1013761750D+01  0.1130767015D+01  0.7271585773D+00
0 1 1 0.0 1.0

```

(continues on next page)

(continued from previous page)

```

0.2700058226D+00 0.1000000000D+01 0.1000000000D+01
0 1 1 0.0 1.0
0.8450000000D-01 0.1000000000D+01 0.1000000000D+01
0 3 1 0.0 1.0
0.8000000000D+00 0.1000000000D+01
1 4
0 0 3 1.0 1.0
0.1873113696D+02 0.3349460434D-01
0.2825394365D+01 0.2347269535D+00
0.6401216923D+00 0.8137573261D+00
0 0 1 0.0 1.0
0.1612777588D+00 0.1000000000D+01
0 2 1 0.0 1.0
0.2200000000D+01 0.1000000000D+01
0 2 1 0.0 1.0
0.5500000000D+00 0.1000000000D+01
99 0
END
DFT
EXCHANGE
PBE
CORRELAT
PBE
CHUNKS
200
XXLGRID
END
TOLDEE
9
TOLINTEG
11 11 11 11 20
SHRINK
20 20
END

```

With the optimized cell, perform the vibrational analysis

```

graphene
SLAB
1
2.47398205      2.47398694  120.000076
2
6      3.55109989926E-07  1.528176553945E-07  1.149651240873E-06
6      3.333329778900E-01 -3.333334868177E-01 -1.149651240873E-06
FREQCALC
NUMDERIV
2
STEPsize
0.003
END
END
6 6

```

(continues on next page)

(continued from previous page)

```

0 0 5 2 1
  1238.40169380      0.0054568832082
  186.290049920      0.0406384092110
  42.2511763460      0.1802559388800
  11.6765579320      0.4631512175500
  3.59305064820      0.4408717331400
0 0 1 2 1
  0.46866670000      1.000000000000000
0 0 1 0 1
  0.23547146000      1.000000000000000
0 2 3 2 1
  9.46809706210      0.0383878717280
  2.01035451420      0.2111702511200
  0.54771004707      0.5132817211400
0 2 1 0 1
  0.21046214000      1.000000000000000
0 3 1 0 1
  0.52857792000      1.000000000000000
8 5
0 0 6 2.0 1.0
  0.5484671660D+04  0.1831074430D-02
  0.8252349460D+03  0.1395017220D-01
  0.1880469580D+03  0.6844507810D-01
  0.5296450000D+02  0.2327143360D+00
  0.1689757040D+02  0.4701928980D+00
  0.5799635340D+01  0.3585208530D+00
0 1 3 6.0 1.0
  0.1553961625D+02 -0.1107775495D+00  0.7087426823D-01
  0.3599933586D+01 -0.1480262627D+00  0.3397528391D+00
  0.1013761750D+01  0.1130767015D+01  0.7271585773D+00
0 1 1 0.0 1.0
  0.2700058226D+00  0.1000000000D+01  0.1000000000D+01
0 1 1 0.0 1.0
  0.8450000000D-01  0.1000000000D+01  0.1000000000D+01
0 3 1 0.0 1.0
  0.8000000000D+00  0.1000000000D+01
1 4
0 0 3 1.0 1.0
  0.1873113696D+02  0.3349460434D-01
  0.2825394365D+01  0.2347269535D+00
  0.6401216923D+00  0.8137573261D+00
0 0 1 0.0 1.0
  0.1612777588D+00  0.1000000000D+01
0 2 1 0.0 1.0
  0.2200000000D+01  0.1000000000D+01
0 2 1 0.0 1.0
  0.5500000000D+00  0.1000000000D+01
99 0
END
DFT
EXCHANGE
PBE

```

(continues on next page)

(continued from previous page)

CORRELAT

PBE

CHUNKS

200

XXLGRID

END

TOLDEE

9

TOLINTEG

11 11 11 11 20

SHRINK

20 20

END

NUMDERIV set to 2 means double-sided finite difference while calculating the Hessian. STEPSIZE sets the step size for finite displacements in Å.

The calculation produces the main output file `xxxx.out` and the Hessian file `HESSFREQ.DAT`.

Then follow the instructions in [quickstart example](#) to do local mode analysis with LModeA-nano.

1.7 Quantum Chemistry Modeling Packages

LModeA-nano can directly read the formatted checkpoint file generated by Gaussian and Q-Chem. Other quantum chemistry modeling packages (e.g. ORCA, xtb) requires the formatted data file generated by UniMoVib.

Note: LModeA-nano can do basic local mode analysis for bond length and bond angles. More advanced analysis including decomposition of normal modes, adiabatic connection scheme and other internal coordinates are supported only by LModeA code.

1.7.1 Gaussian

Tested with Gaussian 16.

Perform an OPT+Freq calculation to do geometry optimization and then vibrational analysis.

```
%chk=ben.chk
```

```
%nproc=36
```

```
#p hf/6-31g opt freq geom=connectivity
```

Title Card Required

```
0 1
```

C	-0.54491017	1.16167665	0.00000000
C	0.85024983	1.16167665	0.00000000
C	1.54778783	2.36942765	0.00000000
C	0.85013383	3.57793665	-0.00119900
C	-0.54469117	3.57785865	-0.00167800
C	-1.24229217	2.36965265	-0.00068200
H	-1.09466917	0.20935965	0.00045000

(continues on next page)

(continued from previous page)

```
H          1.39975783    0.20916365    0.00131500
H          2.64746783    2.36950765    0.00063400
H          1.40033383    4.53007965   -0.00125800
H         -1.09481317    4.53013965   -0.00263100
H         -2.34189617    2.36983565   -0.00086200
```

```
1 2 1.5 6 1.5 7 1.0
2 3 1.5 8 1.0
3 4 1.5 9 1.0
4 5 1.5 10 1.0
5 6 1.5 11 1.0
6 12 1.0
7
8
9
10
11
12
```

Please check the vibrational frequencies to make sure that the molecular structure is at a local minimum point.

The %chk is a must to first produce checkpoint file. Convert it to formatted one with

```
$ formchk ben.chk
```

The corresponding .fchk file is then generated. Then follow the instructions in [quickstart example](#) to do local mode analysis with LModeA-nano.

1.7.2 Q-Chem

Tested with Q-Chem 5.1.0

Given an optimized molecular structure, perform vibrational analysis

```
$comment
  water freq
$end

$molecule
0 1
O   -0.000001  -0.000001  -0.408455
H    0.758758  -0.000004   0.204225
H   -0.758756   0.000004   0.204230
$end

$rem
MEM_TOTAL 100000
BASIS cc-pVDZ
METHOD b3lyp
```

(continues on next page)

(continued from previous page)

```
xc_grid 75000302
UNRESTRICTED FALSE
JOBTYPE freq
THRESH 14
SCF_CONVERGENCE 10
GUI 2
$end
```

GUI 2 asks Q-Chem to produce the formatted checkpoint file `.fchk`. Always check the vibrational frequencies before doing local mode analysis. Then follow the instructions in [quickstart example](#) to do local mode analysis with LModeA-nano.

1.7.3 Other Packages

To perform local mode analysis based on the calculation results by other quantum chemistry packages, the UniMoVib package should be installed first.

The workflow is

- Perform vibrational analysis on an optimized molecular structure with a quantum chemistry package (e.g. ORCA, xtb)
- Run UniMoVib to generate a formatted data file `.umv` by parsing the output file in the last step
- Load the `.umv` file to LModeA-nano for local mode analysis

A complete list of supported quantum chemistry packages by UniMoVib can be found at <https://github.com/zorkzou/UniMoVib>.

In this tutorial, ORCA 4 is used as an example.

The vibrational analysis of an optimized molecular structure with ORCA 4 produces an `.hess` file. Compose an input file `job.inp` for UniMoVib

```
a test job

$contrl
  qcprog="orca"
  ifsave=.true.
$end

$qcdata
  fchk="h2o.hess"
$end
```

`ifsave=.true.` saves a formatted data file `.umv`. Running UniMoVib with above input, it produces `job.umv`.

Compose an input file `unimovib.inp` for LModeA-nano

```
@unimovib

UMV = job.umv
```

Follow these steps to perform local mode analysis with LModeA-nano

- open a new PyMOL window and launch LModeA-nano by clicking **Plugin** → **LModeA-nano**

- click the ... (**Browse**) button and select the `unimovib.inp` file
- change the program drop-down list from **VASP** to **UniMoVib**
- click **Load**
- change **Dimensions** to 0 and click **Confirm** button

Then local mode analysis can be performed as described in *quickstart examples*.

1.8 Finite Difference Method

Above tutorials on performing local mode analysis with LModeA-nano all require the calculation of the full Hessian matrix, then the local stretching force constant can be calculated via analytical formula under the hood. In the following, a finite difference method to obtain the same local stretching force constant is introduced for the following situations/purposes

- The chemical structure is too large to have its full Hessian matrix calculated in an efficient manner. In addition, there is only one or two chemical bonds of interest to be analyzed in this system.
- In order to understand local mode analysis in a straightforward manner.

Follow these steps

- optimize the molecular or primitive cell structure to a local minimum point, record the total energy as $E_{(r_0)}$
- find the chemical bond of interest, its bond length is r_0
- lengthen this bond by a tiny amount Δr (e.g. 0.005 Å), fix this bond length or fix the Cartesian coordinates of these two bonded atoms, optimize the remaining structure (including cell parameters in the case of solid), record the final total energy as $E_{(r_0+\Delta r)}$
- shorten this bond by a tiny amount Δr (e.g. 0.005 Å), fix this bond length or fix the Cartesian coordinates of these two bonded atoms, optimize the remaining structure (including cell parameters in the case of solid), record the final total energy as $E_{(r_0-\Delta r)}$
- the approximate local stretching force constant $k_n^a \approx (E_{(r_0+\Delta r)} - 2E_{(r_0)} + E_{(r_0-\Delta r)})/\Delta r^2$

Conversion factors to mdyn/Å are

- 1 Hartree/ \AA^2 = 4.3597 mdyn/Å
- 1 eV/ \AA^2 = 0.1602 mdyn/Å

1.9 Frequently Asked Questions

Q: I have never heard about nor used this method about local vibrational mode theory. Is this method reliable in measuring the chemical bond strength? Will I get trouble if I apply this method in my research project?

The force constant defined in the local vibrational mode theory is a reliable quantitative measure for chemical bond strength (especially for bonds of similar types).

After years of exploration, testing and application, this method has been extensively used by us and peer researchers to study various types of covalent bonds and non-covalent interactions (e.g. hydrogen/halogen bond). Our recent review article has listed the applications of the local vibrational mode theory as of April 2020.

Noteworthy is that Prof. Gernot Frenking who is an expert on energy decomposition analysis (EDA) recently published an article titled *The Strength of A Chemical Bond* on *Int. J. Quantum Chem.* with co-workers. He pointed out that "... the force constant is the **most general measure** for determining the strength of a chemical bond in molecules" and considered the local vibrational mode theory as "... the **best broadly applicable method** for estimating the strength of chemical bonds that is presently available."

Q: Could the local vibrational modes be experimentally measured in labs?

Local vibrational modes are generally not measurable in experiments except for situations like diatomic molecules, isotope substitution and overtone. This is because vibrational spectroscopy measures normal vibrational modes. However, one may consider one normal vibrational mode as a mixture of local vibrational modes.

Q: Is structure optimization a must before local mode analysis? Can I proceed even if there are some imaginary frequencies?

It is **required** to optimize the structure to a local minimum point.

If imaginary frequencies are found in vibrational analysis, one should not proceed to do local mode analysis. Try to eliminate the imaginary frequencies. This implies that current implementation of local mode analysis in LModeA-nano cannot be applied to analyze transition-state (TS) structure.

However, be cautious that some translational or rotational modes might have minor imaginary frequency values.

Q: Is cell optimization a must before local mode analysis for solids? Can I only optimize the Cartesian coordinates of atoms in a solid?

Yes, the complete cell optimization should be conducted. For example, in VASP program the `ISIF=3` should be specified.

Q: Can I apply the local mode analysis to a system whose structure is partially optimized?

For a molecular system where some atoms are fixed during geometry optimization, one can first apply the generalized subsystem vibrational analysis (GSVA) implemented in the UniMoVib package to extract the effective Hessian matrix for the relaxed portion of the molecule (excluding the fixed atoms). Then one can use LModeA-nano to analyze the bond strength for the relaxed portion of this structure with the effective Hessian matrix.

For solid system (or periodic systems in general), GSVA is not extended yet.

Q: Does the consistency in the level of theory matter? Can I calculate one bond force constant at level A and calculate the force constant of a similar bond at level B, then directly compare their values?

While performing local mode analysis, the consistency in the level of theory should be guaranteed.

In other words, it is preferred to calculate a series of structures with the same level of theory in one program.

Q: I got some force constant value by selecting the H...H bond in a water molecule, is this result correct?

One should always be aware of what he/she is calculating in computational chemistry. The local stretching force constant should **ONLY** be calculated for two atoms between which there exists a chemical bond or non-covalent interaction. One should **NEVER** select two random atoms and interpret corresponding force constant as bond strength.

Q: Can I apply the local vibrational mode theory to measure the bond strength between two metal atoms?

If these two metal atoms exist in elemental metals (metallic solids consisting of only one element), it is not possible to apply the local vibrational mode theory because the phonon frequencies of the primitive cell at Gamma point are all zeros.

For other systems like metal clusters and metal oxides, one may apply the local vibrational mode theory to measure the bond strength between two metal atoms.

Q: As only the primitive cell structure is required for local mode analysis in solids, does the k-point sampling matter?

Yes, in order to get reasonable value, the k-point sampling of the first Brillouin zone should be sufficient.

Q: LModeA-nano gives force constant and frequency as output, which one should I used for chemical bond strength measure?

In most situations, one should choose force constant because it is independent of atomic masses.

Q: I got some error when selecting atoms for an angle in a solid structure? How can I handle this?

For an angle definition, it needs three atoms to be picked. Try pick the first atom within the primitive cell. :-)

Q: I got some error message when I load in the second structure after I analyze the first one. How to handle this issue?

One needs to close the current PyMOL window after analyzing one structure and reopen a new PyMOL window for another structure.

1.10 Need Help

If you encounter any issue, please first check the [Frequently Asked Questions \(FAQ\)](#) page.

If the issue cannot be solved, please email to [ywtao.smu\[at\]gmail.com](mailto:ywtao.smu[at]gmail.com) or open an issue on GitHub.

1.11 Citing LModeA-nano

If you find LModeA-nano useful in your research, please support this project by citing the following paper in your publications/reports:

Y. Tao, W. Zou, S. Nanayakkara, et al., [LModeA-nano: A PyMOL Plugin for Calculating Bond Strength in Solids, Surfaces, and Molecules via Local Vibrational Mode Analysis](#), J. Chem. Theory Comput. 18, 1821 (2022)

```
@article{Tao2022,
  year = {2022},
  volume = {18},
  number = {3},
  pages = {1821--1837},
  author = {Yunwen Tao and Wenli Zou and Sadisha Nanayakkara and Elfi Kraka},
  title = {{LModeA}-nano: A {PyMOL} Plugin for Calculating Bond Strength in Solids, Surfaces, and Molecules via Local Vibrational Mode Analysis},
  journal = {J. Chem. Theory Comput.}
}
```

Additional references include:

Y. Tao, W. Zou, D Sethio, et al., [In Situ Measure of Intrinsic Bond Strength in Crystalline Structures: Local Vibrational Mode Theory for Periodic Systems](#), J. Chem. Theory Comput. 15, 1761 (2019)

E. Kraka, W. Zou, Y. Tao, Decoding Chemical Information from Vibrational Spectroscopy Data: Local Vibrational Mode Theory, WIREs: Comput. Mol. Sci. 10, e1480 (2020)

1.12 Application of LModeA-nano

Bonding in Silylene



Communication | Open Access |

Stimuli Responsive Silylene: Electromerism Induced Reversible Switching Between Mono- and Bis-Silylene

Dr. Ravi Yadav, Dr. Xiaofei Sun, Dr. Ralf Köppe, Dr. Michael T. Gamer, Prof. Dr. Florian Weigend, Prof. Dr. Peter W. Roesky

Dedicated to Professor V. Chandrasekhar

First published: 26 September 2022 | <https://doi.org/10.1002/anie.202211115>

R. Yadav, X. Sun, R. Köppe, M. T. Gamer, F. Weigend, P. W. Roesky, Stimuli Responsive Silylene: Electromerism Induced Reversible Switching Between Mono- and Bis-Silylene, Angew. Chem. Int. Ed. 61, e202211115 (2022)

Hydrogen Bonding in Ice Polymorphs

S. Nanayakkara, Y. Tao, E. Kraka, Capturing Individual Hydrogen Bond Strengths in Ices via Periodic Local Vibrational Mode Theory: Beyond the Lattice Energy Picture, J. Chem. Theory. Comput. 18, 562 (2022)

Capturing Individual Hydrogen Bond Strengths in Ices via Periodic Local Vibrational Mode Theory: Beyond the Lattice Energy Picture

Sadisha Nanayakkara,[†] Yunwen Tao,[†] and Elfi Kraka*

Cite This: *J. Chem. Theory Comput.* 2022, 18, 562–579

Read Online

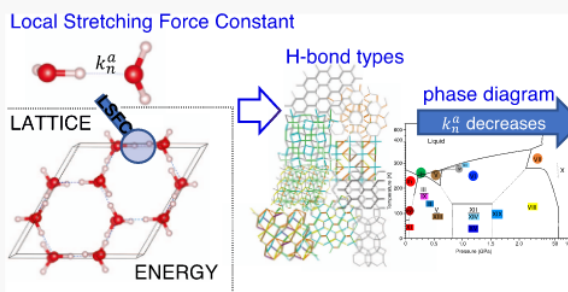
ACCESS |

Metrics & More

Article Recommendations

Supporting Information

ABSTRACT: Local stretching force constants derived from periodic local vibrational modes at the vdW-DF2 density functional level have been employed to quantify the intrinsic hydrogen bond strength of 16 ice polymorphs, ices *Ih*, II, III, IV, V, VI, VII, VIII, IX, XI, XII, XIII, XIV, XV, XVII, and XIX, that are stable under ambient to elevated pressures. Based on this characterization on 1820 hydrogen bonds, relationships between local stretching force constants and structural parameters such as hydrogen bond length and angle were identified. Moreover, different bond strength distributions, from uniform to inhomogeneous, were observed for the 16 ices and could be explained in relation to different local structural elements within ices, that is, rings, that consist of different hydrogen bond types. In addition, criteria for the classification of hydrogen bonds as strong, intermediate, and weak were introduced. The latter was used to explore a different dimension of the water–ice phase diagram. These findings will provide important guidelines for assessing the credibility of new ice structures.



ACKNOWLEDGEMENT & FUNDING

The LModeA-nano project is sponsored by National Science Foundation.