Journal of Molecular Modeling (2020) 26:290
https://doi.org/10.1007/s00894-020-04508-z

SOFTWARE REPORT

PyVibMS: a PyMOL plugin for visualizing vibrations in molecules and solids

Yunwen Tao[1] · Wenli Zou[2] · Sadisha Nanayakkara[1] · Elfi Kraka[1]

Received: 30 June 2020 / Accepted: 4 August 2020

Abstract

Visualizing vibrational motions calculated with different ab initio packages requires dedicated post-processing tools. Here, we present a PyMOL plugin called PyVibMS for visualizing the vibrational motions for both molecular and solid systems calculated by mainstream quantum chemical computer programs including Gaussian, Q–Chem, VASP, and CRYSTAL. Benefiting from the continuing development of the PyMOL platform, PyVibMS provides powerful functionalities and user-friendly interface. PyVibMS was written in Python and its open-source nature makes it flexible and sustainable. As an example, the motions of the Konkoli-Cremer local vibrational modes are shown in this work for the first time. PyVibMS is freely available at https://github.com/smutao/PyVibMS.

Keywords Vibrational spectroscopy · PyMOL · Phonon · Normal mode · Harmonic approximation · Local vibrational mode theory

Introduction

Vibrational spectroscopy is a fundamental analytical tool to study structural properties and dynamics of molecules and solids [1–4]. Infrared (IR) and Raman spectroscopy can be employed to determine the microscopic structure by assigning observed frequency peaks to the vibrations of specific functional groups [5, 6] or to characterize intermolecular forces, e.g., hydrogen bonding [7–9], in both molecular and solid systems [10–14].

With the rapid development in the field of computational chemistry, the theoretical prediction of vibrational spectra of molecular systems within the harmonic approximation based on analytic Hessian matrices is nowadays a routine procedure, which has been implemented in various quantum chemical packages [15, 16]. Extracting vibrational spectra for liquid and solution systems with time-correlation functions of velocities from molecular dynamics simulation trajectories is also on the rise [17–20]. Theoretical modeling of solids, e.g., crystals, although lagging behind the modeling of isolated molecules, has witnessed the development of several first-principles calculation packages (e.g., VASP, CASTEP, CP2K, Quantum ESPRESSO, WIEN2k, DMol3, and CRYSTAL) that allow the calculation of vibrational frequencies of solids [21, 22].

After the lattice vibrational modes and associated frequencies have been calculated for a crystal or other periodic system, it is necessary to visualize the atomic vibrational motions if assigning a vibrational frequency to a certain functional group is desired. When modeling the heterogeneous catalysis on metal surfaces, it would be helpful to visualize the vibrational mode with imaginary frequency in order to check whether the atomic movement in this mode connects the reactant and product geometries.

However, only a few software packages have the functionality to visualize lattice vibrations in solids. One is the wxDragon package [23] which was designed mainly for visualizing the geometries calculated by various ab initio programs. The other is the XCrySDen program [24, 25], which supports the animation of vibration with given trajectory files as a workaround.

There exist also a few online tools that can display the vibration motions of solids. The *Phonon website* created by Miranda [26] takes one step further. It allows the user to visualize the vibrational motions besides Γ

✉ Elfi Kraka
ekraka@gmail.com

[1] Department of Chemistry, Southern Methodist University, 3215 Daniel Ave, Dallas, TX, 75275-0314, USA

[2] Shaanxi Key Laboratory for Theoretical Physics Frontiers, Institute of Modern Physics, Northwest University, Xi'an, Shaanxi, 710127, People's Republic of China

Published online: 28 September 2020

($\mathbf{q} = \mathbf{0}$) point by clicking at the points in a phonon dispersion plot. This online applet has been embedded into two material databases: Materials Project (https://materialsproject.org/) and Materials Cloud (https://www.materialscloud.org/). Recently, Noël designed a webpage [27] as part of the CRYSPLOT [28] suit for visualizing the solid vibrations calculated by CRYSTAL program. However, these online tools require the user to upload the output files and their closed-source nature prohibits the flexibility to meet different needs of the users.

In this work, we present a novel easy-to-use graphic user interface (GUI) named PyVibMS as a plugin for the PyMOL visulization system to visualize vibrations for both molecular and solid systems. PyVibMS is designed to visualize the geometry and vibrational modes for a number of ab initio packages, including Gaussian, Q-Chem, VASP, and CRYSTAL. It is written in Python language and is released as an open-source software, which makes PyVibMS easy to maintain and extend for more possibilities.

Methods

Overall architecture

PyMOL is written in the Python programming language [29, 30]. It offers an easy-to-use Python interface which allows users being familiar with Python to import PyMOL as a module, and to directly manipulate structural objects in the viewer window with the `cmd` functions in an external script. Based on the Python platform, PyMOL has access to a variety of modules and libraries that perform scientific computing, and it fosters many plugins developed by the scientific user community [31–68]. Starting from version 2.0 released in 2017, PyMOL replaces Tkinter with PyQt as its GUI engine and plugin developers could then harness PyQt's more abundant widgets and design the plugin's GUI in a straightforward way.

The overall architecture of the PyVibMS plugin created in this work is shown in Fig. 1. It contains a PyQt GUI window as is shown in Fig. 2. The plugin is compatible with PyMOL 2.x and works on all operating systems including Windows, Linux, and MacOS.

In the following subsections, the functionalities of PyVibMS are described in more detail.

Functionalities

View 3D structures of molecules and solids

PyVibMS reads the input geometry from up to five different file formats:

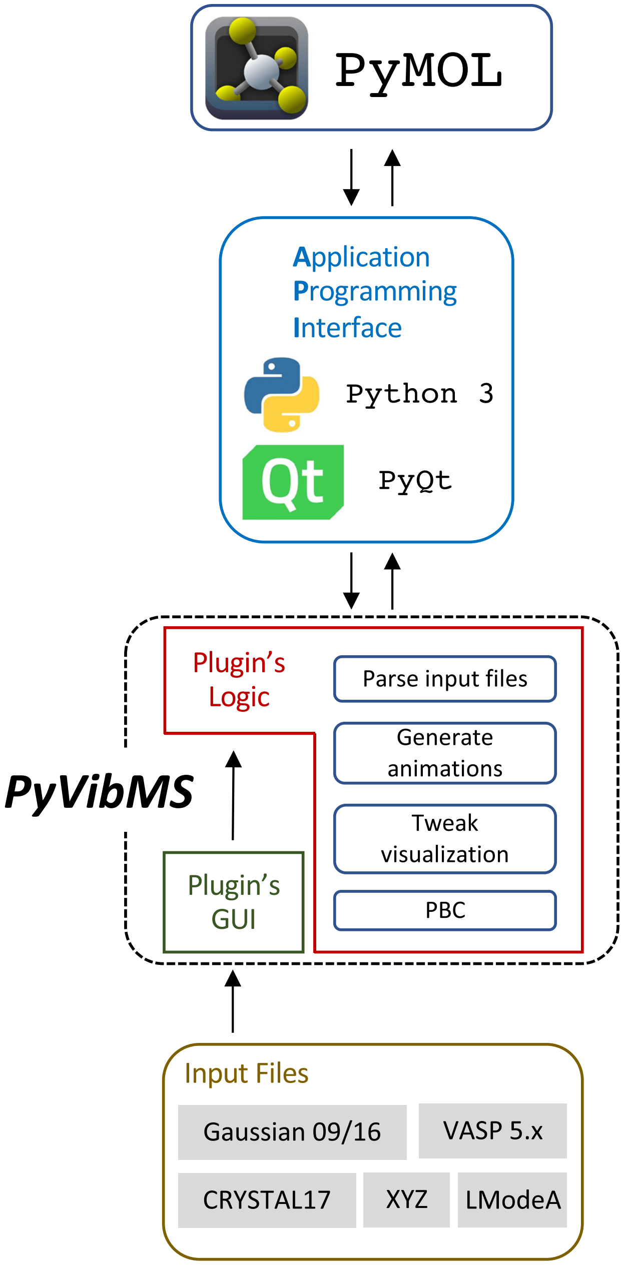
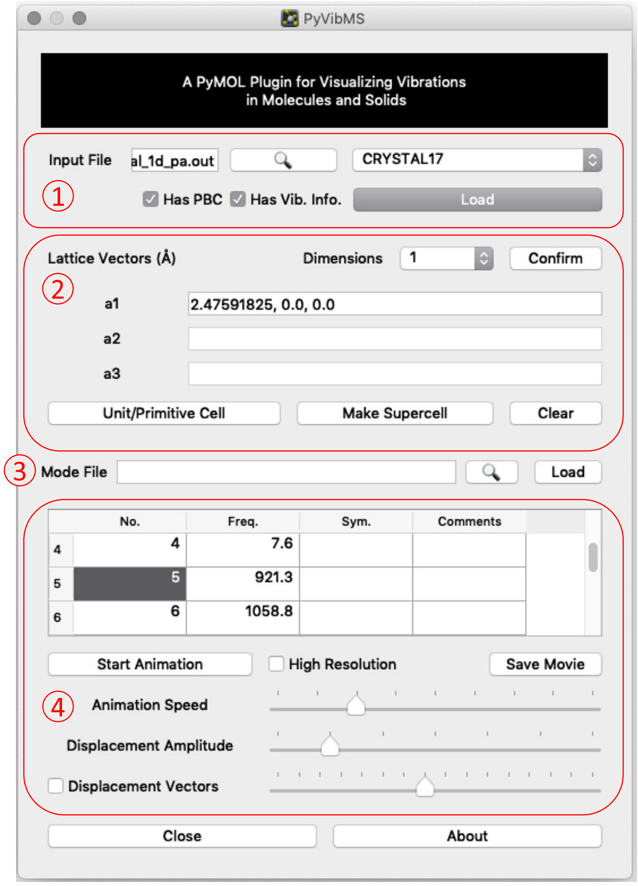


Fig. 1 Flowchart of the communication between PyMOL and the PyVibMS plugin through application programming interface (API)

- XYZ file
- Gaussian 09/16 output file
- Q-Chem output file
- VASP 5.x output with POSCAR/CONTCAR file
- CRYSTAL17 output file

The user can choose the desired file format from a drop-down list located in Section "Introduction" from the GUI window of PyVibMS. If the geometry has been obtained from the calculation with the program currently not supported by PyVibMS, one can extract the Cartesian



PyVibMS

A PyMOL Plugin for Visualizing Vibrations in Molecules and Solids

Input File: al_1d_pa.out — CRYSTAL17

☑ Has PBC ☑ Has Vib. Info.

Lattice Vectors (Å) — Dimensions: 1

a1: 2.47591825, 0.0, 0.0

a2:

a3:

Mode File:

	No.	Freq.	Sym.	Comments
4	4	7.6		
5	5	921.3		
6	6	1058.8		

□ High Resolution

□ Displacement Vectors

Fig. 2 The graphical user interface (GUI) window of the PyVibMS plugin. It is divided into four major sections according to functionalities, including (**1**) loading input geometry, (**2**) setting up periodic boundary conditions (PBC), (**3**) loading extra vibrational mode file, and (**4**) tweaking the visualization of vibrations

coordinates and convert them into an XYZ file, which can be read by PyVibMS.

The geometry is shown with a ball-and-stick model in PyMOL's viewer window if properly imported. PyMOL excels at visualizing biological macromolecules, e.g., protein and nucleic acids; however, it has not been optimized for displaying the structure of small molecules and solids. The default representation of chemical bonding between two bonded atoms is simply a single bond. We implemented an automatic functionality determining the bond order of a particular bond based on the distance between the two atoms in question and adjusting the representation using PyMOL's `valence` command accordingly. In the current version of PyVibMS, a library that contains the mapping between bond length and bond order for CC, CO, and CN bonds has been incorporated. If other types of multiple covalent bonds are desired to be shown, the user can easily add additional distance criteria for multiple bonds into the `judge_valence` function in the PyVibMS code.

Besides molecular structures, the PyVibMS plugin can also be used to show the 3D structure of periodic systems of one, two, and three dimensions. The structure of any periodic system can be described by the unit cell or primitive cell with translational symmetry [69], which is specified by lattice vectors $\{\mathbf{a}_i\}(i = 1, 2, 3)$. If the unit cell is translated along a lattice vector, it will completely overlap with the neighboring unit cell.

Importing solid structure into PyVibMS is a two-step process. (i) The geometry for the unit cell needs to be loaded as it can be done for molecular geometry. (ii) The user needs to enter the information about the lattice vectors in Section "Methods" of the GUI window (see Fig. 2). After selecting and confirming the dimensionality of the imported periodic system, the lattice vector information can be entered. A lattice vector has three components in x, y, and z directions separated by comma, and each component takes the unit of angstrom (Å) in PyVibMS. However, for VASP and CRYSTAL output files, the lattice information is automatically recognized by PyVibMS. Then if the *Unit/Primitive Cell* button is clicked, the lattice vectors will be shown in red, green, and blue colors for $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$ respectively. The structure of the supercell can be shown by clicking the *Make Supercell* button. Atoms within supercell adjacent to the central unit cell will be shown with smaller atomic radii as demonstrated in Fig. 3.

Animate vibrations

Molecular vibrations or lattice vibrations in solids are frequently treated as harmonic oscillations about the equilibrium geometry. In PyVibMS, the animation of a vibration is realized by showing a series of interpolated displacements as a movie, being played with a constant frame rate. The trigonometric sine function is used within the interpolation in order to simulate the movement of the harmonic oscillator.

A vibrational mode displacement vector specifies the direction and amplitude of the movement for each atom. However, it can have different names or even definitions in the output files of various calculation packages, which can be confusing. In this work, we provide a short summary for clarification.

The harmonic normal vibrational modes and frequencies of a molecular system being composed of N atoms are obtained by solving the Wilson equation of vibrational spectroscopy [10]:

$$\mathbf{f}^x\mathbf{L} = \mathbf{M}\mathbf{L}\mathbf{\Lambda} \tag{1}$$

where $\mathbf{f}^x$ is the Hessian matrix in $3N \times 3N$ dimension. $\mathbf{M}$ is the mass matrix collecting the atomic masses for N atoms in three directions as its diagonal elements. The diagonal matrix $\mathbf{\Lambda}$ with the dimension of $N_{vib} \times N_{vib}$ collects N_{vib}

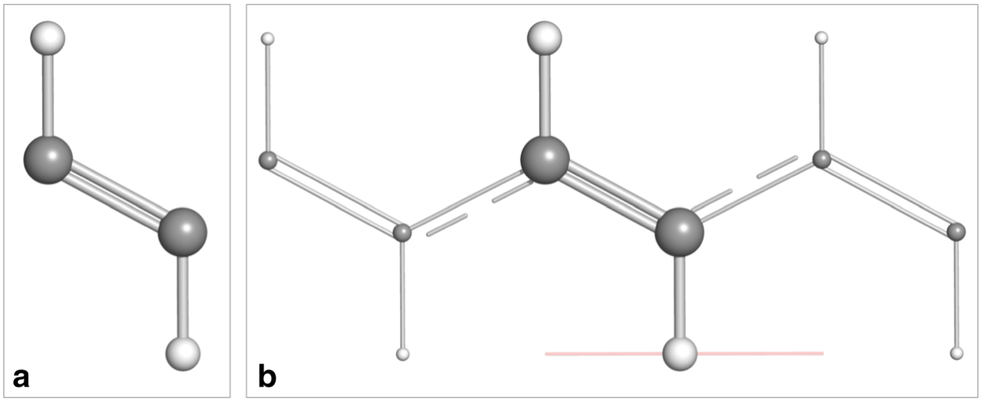


Fig. 3 Imported geometry for 1D polyacetylene (PA). The geometry of unit cell (**a**) is firstly imported, then the supercell (**b**) can be constructed. The line segment in red color represents the lattice vector $\mathbf{a}_1$

vibrational eigenvalues λ_μ ($\mu = 1, ..., N_{vib}$ with $N_{vib} = 3N - K$) where N_{vib} is the total number of molecular vibrations and K equals 5/6 for linear/non-linear molecules. The harmonic vibrational frequencies ω_μ (normally in the unit of cm^{-1}) are related to the eigenvalue λ_μ by:

$$\lambda_\mu = 4\pi^2 c^2 \omega_\mu^2 \tag{2}$$

where c is the speed of light. Matrix $\mathbf{L}$ with the dimension of $3N \times N_{vib}$ collects N_{vib} column vectors $\mathbf{l}_\mu$ which are renormalized and orthogonal to each other. As a result, one obtains $\mathbf{L}^T\mathbf{M}\mathbf{L} = \mathbf{M}^R$ and $\mathbf{L}^T\mathbf{f}^x\mathbf{L} = \mathbf{K}$, which defines the reduced mass matrix $\mathbf{M}^R$ and the diagonal normal force constant matrix $\mathbf{K}$ in normal coordinates, respectively. In Gaussian 09/16 and Q-Chem, the $\mathbf{l}_\mu$ vectors collected in $\mathbf{L}$, termed *normal coordinates*, are printed out as the atomic displacement for each vibration [16].

In a periodic system, i.e., solid, the vibrational analysis usually takes the ansatz of phonon spectroscopy. After the Hessian matrix $\mathbf{f}^x$ of the unit cell has been obtained, a dynamical matrix (mass-weighted Hessian matrix) $\mathbf{D}$ at Γ point ($\mathbf{q} = \mathbf{0}$) is calculated by mass-weighting [70, 71].

$$D_{ai,bj}(\mathbf{q} = \mathbf{0}) = \frac{f_{ij}^x}{\sqrt{m_a m_b}} \tag{3}$$

where m_a and m_b are the masses for atoms a and b associated with the i-th and j-th Cartesian coordinates respectively. Noteworthy is that the term "unit cell" mentioned above arises from periodic boundary conditions (PBC) for calculation. This cell can be either a primitive cell in order to calculate the vibrational motions leading to infrared (IR) and Raman spectra or a supercell used for calculating phonon dispersion relations [72, 73].

By diagonalizing matrix $\mathbf{D}$ for the unit cell containing N atoms, $3N$ eigenvalues λ and their eigenvectors $\mathbf{c}$ are obtained (collected in $\mathbf{\Lambda}$ and $\mathbf{C}$ respectively).

$$\mathbf{DC} = \mathbf{C\Lambda} \tag{4}$$

For a 2D/3D periodic system, it is expected to have three zero eigenvalues, which correspond to the translation of the whole system in three directions. For a 1D periodic system, four zero eigenvalues are produced from diagonalization of $\mathbf{D}$. The fourth zero eigenvalue is linked to the overall rotation of the system around the principal axis parallel to the lattice vector. The remaining $3N - 3/4$ vibrational frequencies ω can be calculated via Eq. 2. In VASP 5.x, the normalized eigenvectors $\mathbf{c}$ are printed in the output file as the atomic displacements for both vibrations and translations/rotations. However, the CRYSTAL09/17 package takes a different path to obtain the atomic displacements for each vibration, while the vibrational frequencies are calculated in the same way as VASP 5.x.

In CRYSTAL09/17, the Hessian matrix $\mathbf{f}^x$ is first diagonalized:

$$\mathbf{f}^x\mathbf{C}' = \mathbf{C}'\mathbf{\Lambda}' \tag{5}$$

and each eigenvector $\mathbf{c}'$ is then mass weighted by:

$$c_{ai}^{\prime m.w.} = \frac{c'_{ai}}{m_a} \tag{6}$$

where m_a is the mass for atom a with i-th Cartesian coordinates leading to a new vector $\mathbf{c}^{\prime m.w.}$ whose norm (vector length) is denoted as u_0. Then, vector $\mathbf{c}^{\prime m.w.}$ is normalized according to the equation of the time average kinetic energy of a phonon mode [74]:

$$\frac{1}{8}\rho V \omega^2 u_0^2 = \frac{1}{2}\left(\frac{1}{2}\hbar\omega\right) \tag{7}$$

which can be re-written as:

$$\rho V \omega u_0^2 = 2\hbar \tag{8}$$

where ω is the vibrational frequency, ρ is the mass density, V is the crystal volume, and ρV does mass-weighting (with m_a) on the square of each element in $\mathbf{c}^{\prime m.w.}$. As the right-hand side of Eq. 8 is two times the Planck constant leading to a constant, vector $\mathbf{c}^{\prime m.w.}$ can be then normalized according to a determined value of u_0 as the mode amplitude in Eq. 8 and thereby leading to a new vector as $\mathbf{c}_0^{\prime m.w.}$. In the CRYSTAL09/17 output file, vectors $\mathbf{c}_0^{\prime m.w.}$ are printed as *normal modes normalized to classical amplitudes* in the unit of Bohr [75].

Although different packages use different flavors for the definition of vibrational mode displacements, these differences are obscured when the displacement of the atoms is translated into the animation on the screen. Therefore, PyVibMS directly takes the displacement information from output files. There is no need for conversion into a consistent definition as explained above.

The user needs to select the *Has Vib. Info.* checkbox in the GUI window before the input geometry file is loaded if it also contains vibrational analysis information. After the vibration information is properly loaded, the vibrational frequency and symmetry (if available) for each vibration will be listed in the table region of the GUI window. Clicking a row in the table selects a target vibration. If the *Start Animation* button is then clicked, the movement of the selected vibration is shown in PyMOL's viewer window.

In addition, PyVibMS provides two slider bars in the GUI window for the user to adjust the animation speed and displacement amplitude of vibrations.

Show displacement vectors

PyVibMS allows the user to visualize the vibrational displacement vectors as arrows by selecting the *Displacement Vectors* checkbox in the GUI window. The slider bar on the right-side scales the length of these arrows. We have used in PyVibMS a script written by Holder to show the arrow object in PyMOL [76].

If an arrow of the displacement vector overlaps with a chemical bond, the user can choose to unselect and then reselect the *Displacement Vectors* checkbox so that all displacement vector arrows will be shown in an opposite direction for better visualization.

Visualize local vibrational modes

In the past few years, our research group has made many contributions to the theory extension [77–82] and application [83–93] of the the local vibrational mode theory originally proposed by Konkoli and Cremer in 1998 [94–100]. We have been employing extensively the local mode force constant k_n^a and local mode frequency ω_n^a in our previous work, but we have not visualized so far how the local vibrational modes look like, although it has been proved that the Konkoli-Cremer local modes are the only local counterpart of normal vibrational modes [77, 88]. With PyVibMS, it is now possible to visualize how the atoms in molecules and solids move during a local vibration.

A local vibrational mode associated with an internal coordinate q_n (e.g., bond length) in a molecular system is defined based on the *leading parameter principle* [94] which describes a process that this internal coordinate is first displaced infinitesimally followed by the relaxation of other parts of this molecule.

The solution of the Wilson equation (Eq. 1) leads to a diagonal Hessian matrix in normal coordinates Q:

$$\mathbf{f}^Q = \mathbf{K} = \mathbf{L}^T \mathbf{f}^x \mathbf{L} \tag{9}$$

where $\mathbf{L}^T$ is the transpose of $\mathbf{L}$.

The internal coordinate q_n leading a local vibrational mode can be defined via the Wilson $\mathbf{B}$-matrix [10], connecting internal coordinate q_n with Cartesian coordinates $\mathbf{x}$ via partial derivatives:

$$\mathbf{b}_n = \frac{\partial q_n}{\partial \mathbf{x}} \tag{10}$$

Row vector $\mathbf{b}_n$ of length of $3N$ converts the N_{vib} vibrational modes collected in $\mathbf{L}$ from Cartesian coordinates into internal coordinates via:

$$\mathbf{d}_n = \mathbf{b}_n \mathbf{L} \tag{11}$$

Row vector $\mathbf{d}_n$ of the length of N_{vib} is then used to derive the local vibrational mode vector $\mathbf{a}_n$ led by the internal coordinate q_n:

$$\mathbf{a}_n = \frac{\mathbf{K}^{-1}\mathbf{d}_n^T}{\mathbf{d}_n \mathbf{K}^{-1}\mathbf{d}_n^T} \tag{12}$$

However, before visualizing local mode $\mathbf{a}_n$ of dimension N_{vib} with PyVibMS, it has to be transformed into Cartesian coordinates via:

$$\mathbf{a}_n^x = \mathbf{L}\mathbf{a}_n \tag{13}$$

so that $\mathbf{a}_n^x$ has the proper length of $3N$. Besides, it can be easily proved that $\mathbf{b}_n \mathbf{a}_n^x = 1$.

In order to visualize a local vibrational mode, the file containing the mode vector $\mathbf{a}_n^x$ needs to be first loaded with Section "Computational details" of the plugin's GUI window. Then, the corresponding local vibrational modes will be added into the table region with corresponding local mode frequencies ω_n^a. The format of the mode file storing the local vibrational mode vectors is elaborated in Appendix 2.

Produce publication-quality images and movies

In order to make publication ready figures with PyVibMS, e.g., the displacement vectors of a vibration, one can benefit from PyMOL's powerful ray tracing function, which renders quality images for publication use.

To export the animation of a vibration as a movie file, it is recommended to use PyMOL's *Export Movie* utility as long as necessary encoders are properly installed. Before recording the movie, the target vibration needs to be selected in the table region. According to our tests, a movie file in (1) animated GIF or (2) QuickTime MOV format has satisfactory publication quality.

Computational details

The geometry of the carbon dioxide (CO_2) molecule was optimized using the B3LYP density functional [101] with Dunning's aug-cc-pVTZ basis set [102, 103] in the Gaussian 16 package [104]. The density functional theory (DFT) calculation was conducted on a pruned (75, 302) grid [105]. The Hessian matrix was evaluated at the same level of theory. The local mode analysis was carried out with the program package COLOGNE2020 [106].

The monocrystalline silicon was modeled in the VASP 5.4.4 package [21, 107–110] using the Perdew-Burke-Ernzerhof (PBE) functional [111, 112] with projector-augmented wave (PAW) potential [113, 114]. The kinetic energy cutoff for basis set is 900 eV. A 12×12×12 **k**-point grid according to the Monkhorst-Pack scheme was employed to sample the Brillouin zone [115] as a primitive cell model was used in this work. A tight convergence criterion ($< 10^{-8}$ eV/Å) was used for cell relaxation. The analytic Hessian matrix for the primitive cell was calculated with density-functional perturbation theory (DFPT) [116].

The two-dimensional (2D) water layer was calculated with the CRYSTAL17 program [117, 118] at the M06-2X/6-31+G(d,2p) level [119–122]. In order for sufficient sampling of **k**-points in reciprocal space, a 12×12 grid was employed according to the Pack-Monkhorst method [115]. Besides, a pruned XXLGRID (99,1454) integration grid was used for DFT calculation. The Hessian matrix was evaluated in a semi-analytical approach by taking numerical derivatives of analytic gradients according to a central-difference formula with the stepsize of 0.001 Å [118].

Results

Carbon dioxide molecule

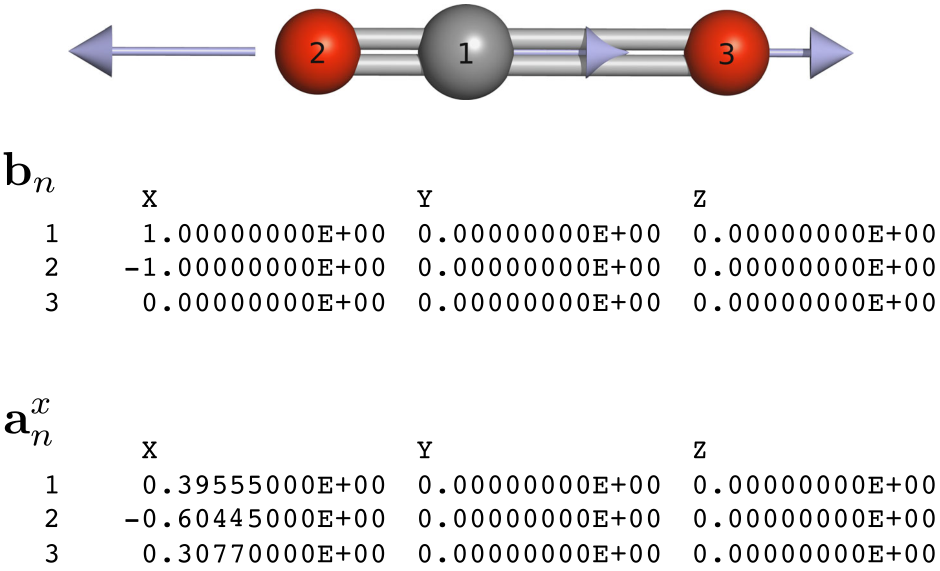
As a demonstration how a local vibrational mode looks like, the local vibration of a CO bond stretching in the carbon dioxide (CO_2) molecule is shown in Fig. 4.

According to the definition of internal coordinates in Eq. 10, the C1=O2 bond length is defined as the partial derivatives of the Cartesian coordinates for these two atoms in the X direction as shown in the components of the $\mathbf{b}_n$ vector. However, the component for the third atom of the local vibrational mode vector $\mathbf{a}_n^x$ is not zero. Instead, with the lengthening of the C1=O2 bond, the O3 atom has a displacement in the +X direction. This is because the local vibrational mode as defined with the leading parameter principle starts with an infinitesimal displacement of the C1=O2 bond and the O3 atom follows effortlessly.

Monocrystalline silicon

The monocrystalline silicon is an important player in materials science. It shares a similar lattice structure with diamond. In the primitive cell of monocrystalline silicon, there are two atoms connected with a Si–Si single bond and in total $3 \times 2 - 3 = 3$ vibrations, which are triply degenerate due to the unique diamond cubic lattice structure as shown in Fig. 5.

All three lattice vibrational modes in the monocrystalline silicon are orthogonal to each other. However, if the perspective is changed into the direction along the Si–Si



$\mathbf{b}_n$
```
          X                Y                Z
  1    1.00000000E+00   0.00000000E+00   0.00000000E+00
  2   -1.00000000E+00   0.00000000E+00   0.00000000E+00
  3    0.00000000E+00   0.00000000E+00   0.00000000E+00
```

$\mathbf{a}_n^x$
```
          X                Y                Z
  1    0.39555000E+00   0.00000000E+00   0.00000000E+00
  2   -0.60445000E+00   0.00000000E+00   0.00000000E+00
  3    0.30770000E+00   0.00000000E+00   0.00000000E+00
```

Fig. 4 The local vibrational mode $\mathbf{a}_n^x$ led by the C(1)O(2) bond stretching in CO_2 molecule. The whole molecules lies in the X-axis direction. The Wilson **B**-matrix vector $\mathbf{b}_n$ and associated local vibrational mode vector $\mathbf{a}_n^x$ of the CO bond are printed in Cartesian coordinates

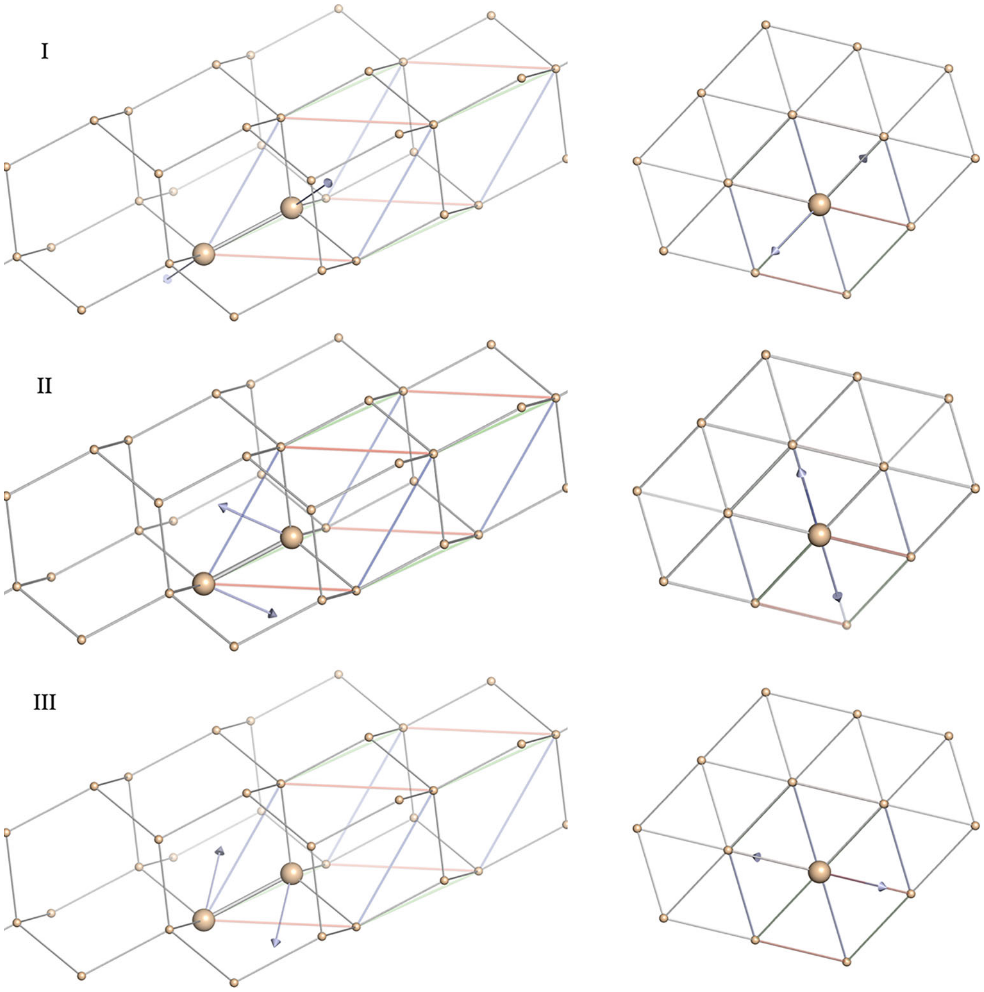


Fig. 5 Three lattice vibrations in a primitive cell of monocrystalline silicon denoted as (I), (II), and (III) with the same vibrational frequency of 502 cm^{-1}. The cell with red, green, and blue edges is a primitive cell containing two silicon atoms with relatively larger size. For each vibration, the structure in two different perspectives are shown

bond axis, one can see that the vibrational mode vectors can be projected onto a planar hexagon and the displacement vectors of two atoms point to a pair of opposite vertices.

Water layer

The 2D water layer is a model system which has been investigated in our previous work to determine the intrinsic strength of its hydrogen bonds [82]. As shown in Fig. 6, all water molecules in this layer have the same orientation and each water molecule donates and accepts two hydrogen bonds simultaneously.

The primitive cell in the 2D water layer can be chosen to contain one water molecule with three atoms. Therefore, in total, six lattice vibrations of the primitive cell are expected as shown in Fig. 7.

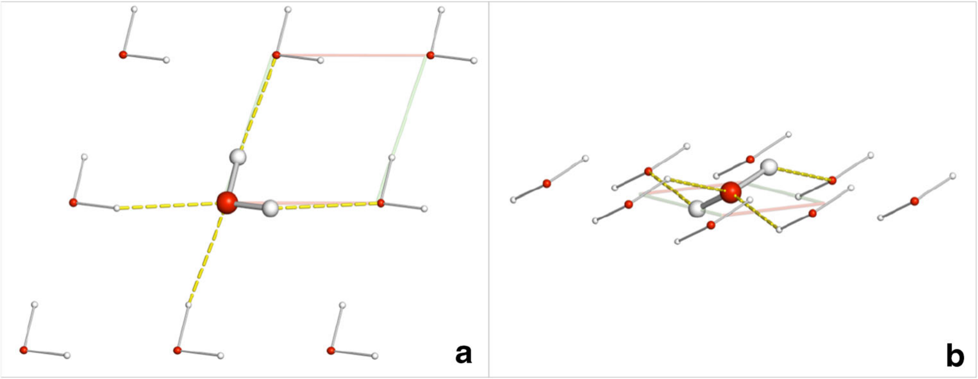


Fig. 6 Structure of the 2D layer of water molecules connected with hydrogen bonding in two different perspectives shown as two panels of (**a**) and (**b**). The cell with red and green edges denotes a primitive cell containing one water molecule. The yellow dashed lines represent four hydrogen bonds each water could form with surrounding water molecules

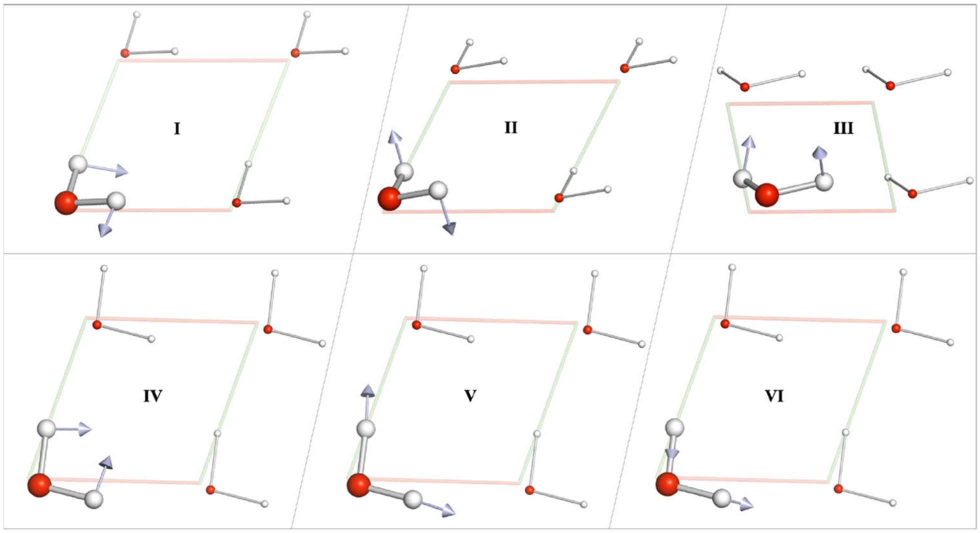


Fig. 7 Six lattice vibrations denoted as (I)∼(VI) in the 2D water layer with the vibrational frequencies as 435, 667, 750, 1642, 3718, and 3799 cm^{-1}, respectively

These six lattice vibrations can be divided into two groups. The first group (I∼III) is basically the overall rotations of the water molecule. Different from molecular water in gas phase, each water molecule in the 2D layer forms hydrogen bonds with neighboring water molecules. Therefore, its rotation causes a change in the total energy, leading to non-zero vibrational frequencies. The three vibrations of the second group (IV∼VI) are the same as the internal vibrations of an isolated water molecule including the angle bending, symmetric O-H bond stretching, and asymmetric stretching. Noteworthy is that the asymmetric O-H stretching (VI) has also higher vibrational frequency than the symmetric O-H stretching (V) as in the case of an isolated water molecule.

Conclusions

PyVibMS is a freely available PyMOL plugin that can be obtained from GitHub (https://github.com/smutao/PyVibMS) and it works on major operating system platforms including Linux, Mac OS, and Windows.

With PyVibMS, one can visualize the vibrational modes of molecular and solid systems calculated by standard computational packages including Gaussian 09/16, VASP 5.x, and CRYSTAL09/17 in the PyMOL system. Benefiting from PyMOL's powerful visualization capability, high-quality figures and movies can be easily generated.

We expect that PyVibMS will be a useful tool for all colleagues who are interested in the vibrational motions in molecules and solids.

Outlook

For future versions of PyVibMS, we plan to implement a functionality of identifying the symmetry (point group/space group and related irreducible representations) of the input geometry and the vibrational modes.

Besides, an interface to the *phonopy* package [123] is under development to visualize the phonons for other wave vectors $\mathbf{q}$ besides the Γ point ($\mathbf{q} = \mathbf{0}$).

PyVibMS will also be interfaced to more ab initio/first-principles modeling packages. With the release of our local mode program LMODEA [100], users will be able to visualize local vibrational mode in both molecules and solids.

Acknowledgments We thank SMU for providing supercomputing resources. Y.T. thanks Yue Qiu and Xin Chen for valuable comments.

Funding This work was financially supported by National Science Foundation Grants CHE 1464906.

Appendix 1: Installation guide

The following instructions describe the installation of the latest version of PyVibMS:

1. Install the latest version of PyMOL 2.x from pre-compiled binaries or source code;
2. Download the latest version of PyVibMS from GitHub repository;
3. Open PyMOL and navigate to the Plugin Manager menu under the Plugin button;

4. On the *Install New Plugin* tab, select the `__init__.py` file in the PyVibMS folder after clicking the *Choose file* button. This step loads PyVibMS into PyMOL;
5. The *PyVibMS* item will be added to the Plugin menu if properly installed. Clicking *PyVibMS* opens its GUI window.

Appendix 2: Format of the user-provided mode file

If the user wants to visualize the local vibrational modes calculated by LMODEA program or molecular/lattice vibrations calculated by a different package outside the supported ones listed in this work, an additional text file can be read like the following.

```
 1 4 2
 2
 3 L 3000.82834 0 CH_stretch
 4 -0.003504561823
 5 -0.071010202331
 6 -0.000001904593
 7 -0.001568317096
 8  0.928990134727
 9  0.000008762757
10  0.003646002972
11 -0.005846446492
12  0.000001286935
13 -0.000206076488
14 -0.013893423928
15 -0.000001632069
16
17 L 1397.92606 0 C=C_stretch
18 -0.457530268453
19  0.205636472277
20 -0.000002593490
21 -0.343583794574
22  0.170823871256
23  0.000002255224
24  0.457151985749
25 -0.205628300954
26  0.000003236451
27  0.348053247013
28 -0.170838415821
29 -0.000006654680
30 END
```

The 1st line contains two integer numbers: the first is the number of atoms N in a molecule or in a primitive/unit cell for solid systems, while the second number specifies the number of additional vibrations this text file has.

The 2nd and 16th lines are the blank lines before the information of each vibration.

The 3rd and 17th lines are the header of each vibration, and each line has four fields. The first field takes either `L` or `N`, representing local vibration and normal vibration respectively. The second field is the vibrational frequency in wavenumbers. The third field gives the irreducible representation of current vibration. Noteworthy is that local vibrations have no symmetry; therefore, we provide `0` here. The last field takes a string which will show up in the comment column in the table section of the GUI window.

Lines 4–15 and 18–29 list $3N$ atomic displacements of vibrations in Cartesian coordinates for the N atoms.

The 30th line of `END` following the displacement information of the last vibration denotes the end of this text file.

References

1. Chalmers J (2002) Handbook of vibrational spectroscopy. Wiley, New York
2. Larkin P (2018) Infrared and Raman spectroscopy: principles and spectral interpretation. Elsevier, Amsterdam
3. Meier RJ (2005) Vibrational spectroscopy: a 'vanishing' discipline? Chem Soc Rev 34(9):743
4. Krimm S, Bandekar J (1986) Vibrational spectroscopy and conformation of peptides, polypeptides, and proteins. In: Advances in protein chemistry, vol 38. Elsevier, pp 181–364
5. Pouchert C (1997) The Aldrich library of FT-IR spectra. Aldrich, Milwaukee
6. Paschoal VH, Faria LFO, Ribeiro MCC (2017) Vibrational spectroscopy of ionic liquids. Chem Rev 117(10):7053–7112
7. Bakker HJ, Skinner JL (2010) Vibrational spectroscopy as a probe of structure and dynamics in liquid water. Chem Rev 110(3):1498–1517
8. Skinner JL, Pieniazek PA, Gruenbaum SM (2011) Vibrational spectroscopy of water at interfaces. Acc Chem Res 45(1):93–100
9. Perakis F, Marco LD, Shalit A, Tang F, Kann ZR, Kühne TD, Torre R, Bonn M, Nagata Y (2016) Vibrational spectroscopy and dynamics of water. Chem Rev 116(13):7590–7607
10. Wilson EB, Decius JC, Cross PC (2012) Molecular vibrations: the theory of infrared and Raman vibrational spectra. Dover Publications, Mineola
11. Bratos S, Pick RM (1980) Vibrational spectroscopy of molecular liquids and solids. Springer, Boston
12. Mitra SS (1962) Vibration spectra of solids. In: Solid state physics. Elsevier, pp 1–80
13. Tarte P (1990) Vibrational spectroscopy and solid state chemistry. Solid State Ion 42(3-4):177–196
14. Sherwood PMA (2011) Vibrational spectroscopy of solids. Cambridge University Press, Cambridge
15. Zou W (2020) UniMoVib: A unified interface for molecular harmonic vibrational frequency calculations. https://github.com/zorkzou/UniMoVib
16. Ochterski JW (1999) White paper: vibrational analysis in gaussian. Gaussian
17. Martinez M, Gaigeot M-P, Borgis D, Vuilleumier R (2006) Extracting effective normal modes from equilibrium dynamics at finite temperature. J Chem Phys 125(14):144106
18. Brehm M, Kirchner B (2011) TRAVIS - a free analyzer and visualizer for Monte Carlo and molecular dynamics trajectories. J Chem Inf Model 51(8):2007–2023
19. Thomas M, Brehm M, Fligg R, Vöhringer P, Kirchner B (2013) Computing vibrational spectra from ab initio molecular dynamics. Phys Chem Chem Phys 15(18):6608–6622
20. Gastegger M, Behler J, Marquetand P (2017) Machine learning molecular dynamics for the simulation of infrared spectra. Chem Sci 8(10):6924–6935

21. Hafner J (2008) Ab-initio simulations of materials using vasp: density-functional theory and beyond. J Comput Chem 29(13):2044–2078
22. Dovesi R, Pascale F, Civalleri B, Doll K, Harrison NM, Bush I, D'Arco P, Noël Y, Rérat M, Carbonnière P, Causà M, Salustro S, Lacivita V, Kirtman B, Ferrari AM, Gentile FS, Baima J, Ferrero M, Demichelis R, Pierre MDL (2020) The CRYSTAL code, 1976-2020 and beyond, a long story. J Chem Phys 152(20):204111
23. Eck B (2019) wxDragon Version 2.1.7. http://www.wxdragon.de/
24. Kokalj A (1999) XCrySDen-a new program for displaying crystalline structures and electron densities. J Mol Graph Model 17(3-4):176–179
25. Kokalj A (2003) Computer graphics and graphical user interfaces as tools in simulations of matter at the atomic scale. Computat Mater Sci 28(2):155–168
26. Miranda H (2019) Phonon website - visualize phonon vibrational modes, GitHub. http://henriquemiranda.github.io/phononwebsite/
27. Noël Y (2018) Animation of vibrational modes and Simulated IR/Raman spectra with CRYSTAL. http://crysplot.crystalsolutions.eu/web_pages_yves3/vibration.html
28. Beata G, Perego G, Civalleri B (2019) CRYSPLOT: a new tool to visualize physical and chemical properties of molecules, polymers, surfaces, and crystalline solids. J Comput Chem 40(26):2329–2338
29. DeLano WL (2002) The PyMOL molecular graphics system. DeLano Scientific, San Carlos, CA, USA
30. Schrödinger LLC (2017) The PyMOL molecular graphics system. Version 2.0
31. Atilgan AR, Durell SR, Jernigan RL, Demirel MC, Keskin O, Bahar I (2001) Anisotropy of fluctuation dynamics of proteins with an elastic network model. Biophys J 80(1):505–515
32. Grell L, Parkin C, Slatest L, Craig PA (2006) EZ-Viz, a tool for simplifying molecular viewing in PyMOL. Biochem Mol Biol Educ 34(6):402–407
33. Medek P, Beneš P, Sochor J (2007) Computation of tunnels in protein molecules using Delaunay triangulation. J WSCG 15:107–114
34. Hodis E, Schreiber G, Rother K, Sussman JL (2007) eMovie: a storyboard-based tool for making molecular movies. Trends in Biochem Sci 32(5):199–204
35. Ordog R (2008) PyDeT, A PyMOL plug-in for visualizing geometric concepts around proteins. Bioinformation 2(8):346–347
36. Steinkellner G, Rader R, Thallinger GG, Kratky C, Gruber K (2009) VASCo: computation and visualization of annotated protein surface contacts. BMC Bioinform 10(1):32
37. Lua RC, Lichtarge O (2010) PyETV: a PyMOL evolutionary trace viewer to analyze functional site predictions in protein complexes. Bioinformatics 26(23):2981–2982
38. Seeliger D, de Groot BL (2010) Ligand docking and binding site analysis with PyMOL and AutoDock/Vina. J Comput Aided Mol Des 24(5):417–422
39. Cabrera AC, Gil-Redondo R, Perona A, Gago F, Morreale A (2011) VSDMIP 1.5: an automated structure- and ligand-based virtual screening platform with a PyMOL graphical user interface. J Comput Aided Mol Des 25(9):813–824
40. Lua RC (2012) PyKnot: a PyMOL tool for the discovery and analysis of knots in proteins. Bioinformatics 28(15):2069–2071
41. Martin OA, Vila JA, Scheraga HA (2012) CheShift-2: graphic validation of protein structures. Bioinformatics 28(11):1538–1539
42. Faure G, Andreani J, Guerois R (2012) InterEvol database: exploring the structure and evolution of protein complex interfaces. Nucleic Acids Res 40(D1):D847–D856
43. Hagelueken G, Ward R, Naismith JH, Schiemann O (2012) MtsslWizard: in silico spin-labeling and generation of distance distributions in pymol. Appl Magn Reson 42(3):377–391
44. Bramucci E, Paiardini A, Bossa F, Pascarella S (2012) PyMod: sequence similarity searches, multiple sequence-structure alignments, and homology modeling within PyMOL. BMC Bioinform 13(Suppl 4):S2
45. Pasi M, Tiberti M, Arrigoni A, Papaleo E (2012) xPyder: a PyMOL plugin to analyze coupled residues and their networks in protein structures. J Chem Inf Model 52(7):1865–1874
46. Sehnal D, Vařeková RS, Berka K, Pravda L, Navrátilová V, Banáš P, Ionescu C-M, Otyepka M, Koča J (2013) MOLE 2.0: advanced approach for analysis of biomacromolecular channels. J Cheminform 5(1):39
47. Bachega JFR, Timmers LFSM, Assirati L, Bachega LR, Field MJ, Wymore T (2013) GTKDynamo: a PyMOL plug-in for QC/MM hybrid potential simulations. J Comput Chem 34(25):2190–2196
48. Makarewicz T, Kaźmierkiewicz R (2013) Molecular dynamics simulation by GROMACS using GUI plugin for PyMOL. J Chem Inf Model 53(5):1229–1234
49. Oberhauser N, Nurisso A, Carrupt P-A (2014) MLP tools: a PyMOL plugin for using the molecular lipophilicity potential in computer-aided drug design. J Comput Aided Mol Des 28(5):587–596
50. Hu B, Lill MA (2014) WATsite: hydration site prediction program with PyMOL interface. J Comput Chem 35(16):1255–1260
51. Patel H, Gruning BA, Gunther S, Merfort I (2014) PyWATER: a PyMOL plug-in to find conserved water molecules in proteins by clustering. Bioinformatics 30(20):2978–2980
52. Oliveira SauloHP, Ferraz FelipeAN, Honorato RV, Xavier-Neto J, Sobreira TiagoJP, de Oliveira PauloSL (2014) KVFinder: steered identification of protein cavities as a pymol plugin. BMC Bioinform 15(1):197
53. Warnecke A, Sandalova T, Achour A, Harris RA (2014) PyTMs: a useful PyMOL plugin for modeling common post-translational modifications. BMC Bioinform 15(1):370
54. Gaudreault F, Morency L-P, Najmanovich RJ (2015) NRGsuite: a PyMOL plugin to perform docking simulations in real time using flexaid. Bioinformatics 31(23):3856–3858
55. Chaudhari R, Li Z (2015) PyMine: a PyMOL plugin to integrate and visualize data for drug discovery. BMC Res Notes 8(1):517
56. Baumgartner M (2016) Improving rational drug design by incorporating novel biophysical insight. Ph.D. Thesis, University of Pittsburgh
57. Panjkovich A, Svergun DI (2016) SASpy: a PyMOL plugin for manipulation and refinement of hybrid models against small angle x-ray scattering data. Bioinformatics 32(13):2062–2064
58. Makarewicz T, Kaźmierkiewicz R (2016) Improvements in GROMACS plugin for pymol including implicit solvent simulations and displaying results of PCA analysis. J Mol Model 22(5):109
59. Arroyuelo A, Vila JA, Martin OA (2016) Azahar: a PyMOL plugin for construction, visualization and analysis of glycan molecules. J Comput Aided Mol Des 30(8):619–624
60. Sridhar A, Ross GA, Biggin PC (2017) Waterdock 2.0: water placement prediction for holo-structures with a PyMOL plugin. PLOS ONE 12(2):e0172743
61. Masand VH, Rastija V (2017) PyDescriptor : a new PyMOL plugin for calculating thousands of easily understandable molecular descriptors. Chemom Intell Lab Syst 169:12–18
62. Janson G, Zhang C, Prado MG, Paiardini A (2017) PyMod 2.0: improvements in protein sequence-structure analysis and homology modeling within PyMOL. Bioinformatics 33(3):444–446

63. Gierut AM, Niemyska W, Dabrowski-Tumanski P, Sułkowski P, Sulkowska JI (2017) PyLasso: a PyMOL plugin to identify lassos. Bioinformatics 33(23):3819–3821
64. Yuan S, Chan HS, Hu Z (2017) Using PyMOL as a platform for computational drug design. Wiley Interdiscip Rev Comput Mol Sci 7(2):e1298
65. Jurrus E, Engel D, Star K, Monson K, Brandi J, Felberg LE, Brookes DH, Wilson L, Chen J, Liles K, Chun M, Li P, Gohara DW, Dolinsky T, Konecny R, Koes DR, Nielsen JE, Head-Gordon T, Geng W, Krasny R, Wei G-W, Holst MJ, McCammon JA, Baker NA (2018) Improvements to the APBS biomolecular solvation software suite. Protein Sci 27(1):112–128
66. Jarmolinska AI, Zhou Q, Sulkowska JI, Morcos F (2019) DCA-MOL: a PyMOL plugin to analyze direct evolutionary couplings. J Chem Inf Model 59(2):625–629
67. Gierut AM, Dabrowski-Tumanski P, Niemyska W, Millett KC, Sulkowska JI, Valencia A (2019) PyLink: a PyMOL plugin to identify links. Bioinformatics 35(17):3166–3168
68. Lu X-J (2020) DSSR-enabled innovative schematics of 3d nucleic acid structures with PyMOL. Nucleic Acids Res 48(13):e74. https://doi.org/10.1093/nar/gkaa426
69. Tilley RJD (2006) Crystals and crystal structures. Wiley, New York
70. Born M, Huang K (1954) Dynamical theory of crystal lattices. Clarendon Press, Oxford
71. Fang T, Li Y, Li S (2017) Generalized energy-based fragmentation approach for modeling condensed phase systems. Wiley Interdiscip Rev Comput Mol Sci 7(2):e1297
72. Civalleri B, Pascale F, Noel Y (2017) Vibrational frequencies calculation. CRYSTAL 2017
73. Ferrabone M, Baima J (2017) Phonon dispersion with CRYSTAL. CRYSTAL 2017
74. Kittel C (2004) Introduction to solid state physics. In: Crystal Vibrations. Wiley, p 99
75. Pascale F, Zicovich-Wilson CM, Gejo FL, Civalleri B, Orlando R, Dovesi R (2004) The calculation of the vibrational frequencies of crystalline compounds and its implementation in the CRYSTAL code. J Comput Chem 25(6):888–897
76. Holder T (2013) Cgo arrow. Accessed: 2019-06-19. https://pymolwiki.org/index.php/Cgo_arrow
77. Zou W, Kalescky R, Kraka E, Cremer D (2012) Relating normal vibrational modes to local vibrational modes with the help of an adiabatic connection scheme. J Chem Phys 137(8):084114
78. Zou W, Cremer D (2014) Properties of local vibrational modes: the infrared intensity. Theor Chem Acc 133:1451–1466
79. Tao Y, Zou W, Cremer D, Kraka E (2017) Correlating the vibrational spectra of structurally related molecules: a spectroscopic measure of similarity. J Comput Chem 39(6):293–306
80. Tao Y, Tian C, Verma N, Zou W, Wang C, Cremer D, Kraka E (2018) Recovering intrinsic fragmental vibrations using the generalized subsystem vibrational analysis. J Chem Theory Comput 14(5):2558–2569
81. Tao Y (2018) Advances in local vibrational mode theory and Unified Reaction Valley Approach (URVA). Ph.D. Thesis, Southern Methodist University. Chemistry theses and dissertations. 2. https://scholar.smu.edu/hum_sci_chemistry_etds/2
82. Tao Y, Zou W, Sethio D, Verma N, Qiu Y, Tian C, Cremer D, Kraka E (2019) In situ measure of intrinsic bond strength in crystalline structures: local vibrational mode theory for periodic systems. J Chem Theory Comput 15(3):1761–1776
83. Cremer D, Wu A, Larsson JA, Kraka E (2000) Some thoughts about bond energies, bond lengths, and force constants. J Mol Model 6(4):396–412
84. Kalescky R, Kraka E, Cremer D (2013) Identification of the strongest bonds in chemistry. J Phys Chem A 117(36):8981–8995
85. Zou W, Cremer D (2016) C_2 in a box: determining its intrinsic bond strength for the $X^1 \ \Sigma^+{}_g$ Ground State. Chem Eur J 22(12):4087–4097
86. Zhang X, Dai H, Yan H, Zou W, Cremer D (2016) B-H$\cdot\cdot\cdot\pi$ interaction: a new type of nonclassical hydrogen bonding. J Am Chem Soc 138(13):4334–4337
87. Tao Y, Zou W, Jia J, Li W, Cremer D (2017) Different ways of hydrogen bonding in water - why does warm water freeze faster than cold water? J Chem Theory Comput 13(1):55–76
88. Cremer D, Kraka E (2017) Generalization of the tolman electronic parameter: the metal-ligand electronic parameter and the intrinsic strength of the metal-ligand bond. Dalton Trans 46(26):8323–8338
89. Tao Y, Zou W, Kraka E (2017) Strengthening of hydrogen bonding with the push-pull effect. Chem Phys Lett 685:251–258
90. Tao Y, Zou W, Cremer D, Kraka E (2017) Characterizing chemical similarity with vibrational spectroscopy: new insights into the substituent effects in monosubstituted benzenes. J Phys Chem A 121(42):8086–8096
91. Verma N, Tao Y, Marcial BL, Kraka E (2019) Correlation between molecular acidity (pka) and vibrational spectroscopy. J Mol Model 25(2):48
92. Tao Y, Qiu Y, Zou W, Nanayakkara S, Yannacone S, Kraka E (2020) In situ assessment of intrinsic strength of X-I$\cdots$OA type halogen bonds in molecular crystals with periodic local vibrational mode theory. Molecules 25(7):1589
93. Verma N, Tao Y, Zou W, Chen X, Chen X, Freindorf M, Kraka E (2020) A critical evaluation of Vibrational Stark Effect (VSE) probes with the local vibrational mode theory. Sensors 20(8):2358
94. Konkoli Z, Cremer D (1998) A new way of analyzing vibrational spectra. I. Derivation of adiabatic internal modes. Int J Quantum Chem 67(1):1–9
95. Konkoli Z, Larsson JA, Cremer D (1998) A new way of analyzing vibrational spectra. II. Comparison of internal mode frequencies. Int J Quantum Chem 67(1):11–27
96. Konkoli Z, Cremer D (1998) A new way of analyzing vibrational spectra. III. Characterization of normal vibrational modes in terms of internal vibrational modes. Int J Quantum Chem 67(1):29–40
97. Konkoli Z, Larsson JA, Cremer D (1998) 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(1):41–55
98. Kraka E, Cremer D (2019) Dieter cremer's contribution to the field of theoretical chemistry. Int J Quantum Chem 119(6):e25849
99. Kraka E (2019) Preface: dieter cremer's scientific journey. Mol Phys 117(9-12):1047–1058
100. Kraka E, Zou W, Tao Y (2020) Decoding chemical information from vibrational spectroscopy data – local vibrational mode theory. WIREs: Comput Mol Sci 10(5):e1480. https://doi.org/10.1002/wcms.wcms.1480
101. Stephens PJ, Devlin FJ, Chabalowski CF, Frisch MJ (1994) Ab initio calculation of vibrational absorption and circular dichroism spectra using density functional force fields. J Phys Chem 98(45):11623–11627
102. Dunning TH (1989) Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen. J Chem Phys 90(2):1007–1023
103. Woon DE, Dunning TH (1993) Gaussian basis sets for use in correlated molecular calculations. III. The atoms aluminum through argon. J Chem Phys 98(2):1358–1371
104. Frisch MJ, Trucks GW, Schlegel HB, Scuseria GE, Robb MA, Cheeseman JR, Scalmani G, Barone V, Petersson GA, Nakatsuji H, Li X, Caricato M, Marenich AV, Bloino J, Janesko BG,

Gomperts R, Mennucci B, Hratchian HP, Ortiz JV, Izmaylov AF, Sonnenberg JL, Williams-Young D, Ding F, Lipparini F, Egidi F, Goings J, Peng B, Petrone A, Henderson T, Ranasinghe D, Zakrzewski VG, Gao J, Rega N, Zheng G, Liang W, Hada M, Ehara M, Toyota K, Fukuda R, Hasegawa J, Ishida M, Nakajima T, Honda Y, Kitao O, Nakai H, Vreven T, Throssell K, Montgomery JA, Peralta JE, Ogliaro F, Bearpark MJ, Heyd JJ, Brothers EYN, Kudin KN, Staroverov VN, Keith TYA, Kobayashi R, Normand J, Raghavachari K, Rendell AP, Burant JC, Iyengar SS, Tomasi J, Cossi M, Millam JM, Klene M, Adamo C, Cammi R, Ochterski JW, Martin RL, Morokuma K, Farkas O, Foresman JB, Fox DJ (2016) Gaussian 16 Revision B.01. Gaussian Inc. Wallingford CT

105. Dasgupta S, Herbert JM (2017) Standard grids for high-precision integration of modern density functionals: sg-2 and sg-3. J Comput Chem 38(12):869–882

106. Kraka E, Zou W, Filatov M, Tao Y, Grafenstein J, Izotov D, Gauss J, He Y, Wu A, Konkoli Z, Polo V, Olsson L, He Z, Cremer D (2020) COLOGNE2020. see http://www.smu.edu/catco

107. Kresse G, Hafner J (1993) Ab initio molecular dynamics for liquid metals. Phys Rev B 47(1):558–561

108. Kresse G, Hafner J (1994) Ab initio molecular-dynamics simulation of the liquid-metal-amorphous-semiconductor transition in germanium. Phys Rev B 49(20):14251–14269

109. Kresse G, Furthmüller J (1996) Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. Comput Mater Sci 6(1):15–50

110. Kresse G, Furthmüller J (1996) Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. Phys Rev B 54(16):11169–11186

111. Perdew JP, Burke K, Ernzerhof M (1996) Generalized gradient approximation made simple. Phys Rev Lett 77(18):3865

112. Perdew JP, Burke K, Ernzerhof M (1997) Erratum: generalized gradient approximation made simple [Phys. Rev. Lett. 77, 3865 (1996)]. Phys Rev Lett 78(7):1396–1396

113. Blöchl PE (1994) Projector augmented-wave method. Phys Rev B 50(24):17953

114. Kresse G, Joubert D (1999) From ultrasoft pseudopotentials to the projector augmented-wave method. Phys Rev B 59(3):1758

115. Monkhorst HJ, Pack JD (1976) Special points for brillouin-zone integrations. Phys Rev B 13(12):5188–5192

116. Baroni S, deGironcoli S, Corso AD, Giannozzi P (2001) Phonons and related crystal properties from density-functional perturbation theory. Rev Mod Phys 73(2):515–562

117. Dovesi R, Erba A, Orlando R, Zicovich-Wilson CM, Civalleri B, Maschio L, Rérat M, Casassa S, Baima J, Salustro S, Kirtman B (2018) Quantum-mechanical condensed matter simulations with crystal. Wiley Interdiscip Rev Comput Mol Sci 8(4):e1360

118. Dovesi R, Saunders VR, Roetti C, Orlando R, Zicovich-Wilson CM, Pascale F, Civalleri B, Doll K, Harrison NM, Bush IJ, D'Arco P, Llunell M, Causà M, Noël Y, Maschio L, Erba A, Rerat M, Casassa S (2017) CRYSTAL17 User's Manual. University of Torino, Torino

119. Zhao Y, Truhlar DG (2008) The m06 suite of density functionals for main group thermochemistry, thermochemical kinetics, noncovalent interactions, excited states, and transition elements: two new functionals and systematic testing of four m06-class functionals and 12 other functionals. Theor Chem Acc 120(1):215–241

120. Dill JYD, Pople JA (1975) Self-consistent molecular orbital methods. XV. Extended gaussian-type basis sets for lithium, beryllium, and boron. J Chem Phys 62(7):2921–2923

121. Hehre WJ, Ditchfield R, Pople JA (1972) Self-consistent molecular orbital methods. xii. Further extensions of gaussian-type basis sets for use in molecular orbital studies of organic molecules. J Chem Phys 56(5):2257–2261

122. Hariharan PC, Pople JA (1973) The influence of polarization functions on molecular orbital hydrogenation energies. Theor Chem Acc 28(3):213–222

123. Togo A, Tanaka I (2015) First principles phonon calculations in materials science. Scr Mater 108:1–5

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.