CHEM-279 Final Project

Determining and Visualizing the Normal Modes of Various Molecular Systems

Abstract

Normal modes are an important property of molecular systems. They allow us to measure the direction and intensity of internal molecular vibrations. Normal modes play an important role in numerous techniques, such as spectroscopy. For this project, we developed a C++ program for obtaining and visualizing the normal modes of molecular systems. Our program optimizes the geometry of molecular systems, calculates the normal modes, and produces a sequential set of PDB/XYZ files for visualization using software such as Avogadro or PyMOL. Our program provides a computational tool that produces valuable insights about the behavior of a molecular system.

Background

Vibrational analysis provides insights and information regarding the stability, symmetry, and spectroscopic signature of a molecular system. Normal modes can reveal molecular properties such as whether the molecule is at the optimal geometry and transition state characteristics. Performing these calculations is a critical component of computational chemistry. For this project, we developed a C++ program that calculates a molecular system's normal modes and produces a file containing a series of coordinates, allowing the vibrational modes to be visualized using visualization software, such as Avogadro.

Normal modes are the coordinated internal atomic motions that occur at the molecule's optimal geometry. If a molecule is at its optimal geometry, any atomic displacement will increase the energy of the system, causing the restoring forces of the system to return the molecule to the lower energy state. The number of normal modes a molecular system has is a function of the number of atoms it has, 3N - 5 for linear systems and 3N - 6 for nonlinear systems, and each normal mode is independent of all the other modes of the system. The reason is that these motions behave like springs that vibrate on their own with a unique spring constant, and thus the normal modes do not interact with each other.

Vibrational analysis allows researchers to associate molecules and their structures with observed properties. The vibrational frequencies observed with normal modes correspond to spectroscopic features, allowing comparisons between computational predictions and experimental results. Additionally, vibrational analysis allows researchers to verify whether a molecule is at the optimal geometry. At the optimal geometry, the eigenvalues of all the calculated eigenvectors that represent the normal modes will be nonnegative. Eigenvectors with nonzero nonnegative eigenvalues are the normal modes. Any eigenvectors with eigenvalues equal to zero are the translational and rotational modes. If an eigenvector has a negative eigenvalue, then that eigenvector represents an imaginary mode. If the geometry is indeed optimized, there will not be any imaginary modes. If an atom moves in a direction that does not increase the energy, the restoring forces will not act to return the molecule to its previous position. This indicates the geometry was not optimal. Therefore, vibrational analysis can alert researchers to this.

Objectives

Our objective was to develop a computational program that analyzes a molecular system and provides the user with the calculated normal modes and files for visualizing them. We aimed to build a program that could, from a list of atoms and atomic coordinates, build basis functions, compute overlap integrals, run the self-consistent field algorithm, optimize the geometry, and compute and produce the files required for visualizing the normal modes. Our goal was to provide researchers with a way to calculate the modes and produce the visualizations needed for any downstream applications. The program was designed to distinguish between translational, rotational, and vibrational modes and to verify whether the molecule was at the optimal geometry via the presence or absence of imaginary modes.

Methodology

To achieve our objectives, we conducted the following protocol. Starting from our previously written CNDO/2 code, we added a method for optimizing the molecule's geometry and methods for producing the mass-weighted Hessian matrix. We used the linear algebra Armadillo to solve the eigenvalue problem, choosing the appropriate modes for visualization, and using external software to perform the visualizations.

Geometry

To optimize the geometry, we refactored our steepest descent with line search code to be compatible with our new implementation. In order to focus on correctly producing the Hessian matrix, we used AI to streamline the refactoring. At each optimization step, the nuclear coordinates were displaced in the direction of the negative energy gradient, which pointed in the direction of the steepest descent. The step size was determined using line search with the bracketing strategy presented in Numerical Algorithms. Trial displacements were evaluated by computing the energy at new coordinate positions, and the step size was increased or decreased depending on whether the trial step resulted in an energy decrease. This process was repeated until the change in energy between successive steps was below a predefined threshold. This ensures the final geometry corresponds to a local minimum on the potential energy surface, providing a suitable reference structure for subsequent vibrational normal mode analysis.

Hessian

We used the central differences approximation to construct the Hessian from our homework 5 force matrix. This allowed us to obtain the second derivatives of the energy that define the Hessian matrix. We first flattened our force matrix into a 3N vector, then used central differences to approximate the derivative with respect to x, y, and z for each component in the vector, storing these values in the Hessian, resulting in a 3N by 3N Hessian matrix. After constructing the Cartesian Hessian matrix, the Hessian was transformed into its mass-weighted form to obtain physically meaningful vibrational modes. Mass weighting accounts for the different atomic masses in the system and is required to decouple the equations of motion into independent normal modes. Each element of the Hessian was scaled by the inverse square root of the masses associated with the corresponding Cartesian coordinates, producing the mass-weighted Hessian matrix.

Eigenvalues

Next, we used Armadillo to diagonalize the mass-weighted Hessian. Solving this eigenvalue problem yielded a set of eigenvectors and corresponding eigenvalues. The eigenvalues were proportional to the squared vibrational frequencies, while the eigenvectors describe the displacement direction associated with each normal mode. Eigenvectors with near-zero eigenvalues were identified as translational and rotational modes, while eigenvectors with positive eigenvalues corresponded to the vibrational normal modes. Negative eigenvalues indicated the presence of imaginary frequencies, and if present, also indicated that the geometry was not at a true local minimum.

Visualization

To visualize the normal modes, the eigenvectors corresponding to the vibrational modes were converted into atomic displacement vectors in Cartesian space. Small displacements were applied to the optimized molecular geometry along each normal mode direction, generating a sequence of atomic coordinates that represent oscillatory motion about the equilibrium structure. These coordinate sequences were written to a PDB file format suitable for use with molecular visualization software.

We visualized the PDB files using external software such as Avogadro and PyMOL, which allows the normal modes to be animated and inspected qualitatively. The visualizations of the vibrational modes allowed us to distinguish between stretching, bending, and other collective motions, and validate that the computed normal modes were physically reasonable and consistent with molecular symmetry and bonding.

Results and Discussion

Results

The following tables highlight our results for running our module on various molecules and highlight how they compare with experimental results by calculating the error percentage. In addition, each table is accompanied by molecular visualizations to facilitate observing vibrational modes associated with particular frequencies.

A. Linear molecules:

Molecules - Linear	**Frequencies (cm^{-1})**	**Mode of motion**	**Experimental values (cm^{-1})**	**Error(%)**
H_2	Mode 1: 0 Mode 2: 0 Mode 3: 0 Mode 4: 0 Mode 5: 0	Translational & Rotational	–	–
	Mode 6: 6126.04	Symmetric stretch	4401	39.2

HF	Mode 1: 0 Mode 2: 0 Mode 3: 0 Mode 4: 0 Mode 5: 0 Mode 6: 6094.76	Translational & Rotational Symmetric stretch	– 4138	– 47.3
HO	Mode 1: 0 Mode 2: 0 Mode 3: 0 Mode 4: 0 Mode 5: 0 Mode 6: 5703.63	Translational & Rotational Symmetric stretch	– 3737	– 52.6

B. Non linear molecules:

Molecules – Non linear	**Frequencies (cm^{-1})**	**Mode of motion**	**Experimental values (cm^{-1})**	**Error(%)**
H_2O	Mode 1: 0 Mode 2: 0 Mode 3: 0 Mode 4: 0 Mode 5: 0 Mode 6: 0 Mode 7: 1766.42 Mode 8: 5679.51 Mode 9: 5694.04	Translational & Rotational Symmetric bend Symmetric stretch Asymmetric stretch	– 1595 3657 3756	– 10.74 55.3 51.6
NH3	Mode 1: 0 Mode 2: 0 Mode 3: 0 Mode 4: 0 Mode 5: 0 Mode 6: 0 Mode 7: 1506.12 Mode 8: 1778.95 Mode 9: 1779.13 Mode 10: 5225.6 Mode 11: 5264.05 Mode 12: 5264.13	Translational & Rotational Symmetric bend Asymmetric bend Asymmetric bend Symmetric stretch Asymmetric stretch Asymmetric stretch	– 950 1627 1627 3337 3414 3414	– 58.5 9.33 9.35 56.6 54.19 54.19

CH_4	Mode 1: 0 Mode 2: 0 Mode 3: 0 Mode 4: 0 Mode 5: 0 Mode 6: 0	Translational & Rotational	–	–
	Mode 7: 1648.43 Mode 8: 1648.43 Mode 9: 1648.43	Asymmetric bend	1380	19.45
	Mode 10: 1760.53 Mode 11: 1760.53	Symmetric bend	1412	24.68
	Mode 12: 4837.61	Symmetric stretch	3215	50.46
	Mode 13: 4950.33 Mode 14: 4950.33 Mode 15: 4950.33	Asymmetric stretch	3104	59.48

Interpretation

Rigid Body Physics & Degrees of Freedom

The primary major validation of our codebase was observed in the zero-frequency modes. Our data consistently highlighted a distinct difference in the number of zero-frequency modes between different molecular geometries: linear molecules, such as H_2, HF, and OH, showed exactly five zero modes, while non-linear molecules (H_2O, NH_3, and CH_4) exhibited six. These values correspond to rigid-body molecular motions –translation and rotation. When a molecule translates or rotates freely, its potential energy remains unchanged. Mathematically, this results in a force constant equal to zero, and consequently, a vibrational frequency of zero.

This distinction proved to be an important test of our program's ability to successfully identify degrees of freedom. While all molecules have three translational degrees of freedom, the rotational degrees of freedom differ due to molecular geometry. A linear molecule possesses a negligible moment of inertia along its bond axis, and as a result, loses one rotational degree of freedom (3N -5) compared to a non-linear molecule (3N - 6). The fact that our module naturally produced the exact counts of zero frequency modes is a good indication of our solver's understanding of physical molecular motions.

The hierarchy of vibrational modes

Molecular vibrations can be categorized into different modes of motion, namely, bond bending, which are molecular motions that change bond lengths, and bond stretching, motions that alter bond angles. A clear trend emerged while comparing the frequency magnitude for various molecular motions. Across all non-linear molecules, we observed that bending modes appeared at significantly lower frequencies than their stretching counterparts. For instance, the bending modes for H_2O (~1766 cm^{-1}), NH_3 (~1780 cm^{-1}), and CH_4 (~1648 cm^{-1}) are well separated from the high-frequency bond stretching modes.

The stark difference between these frequencies suggests a distinction in the “stiffness” of the restoring forces involved in these motions. Using a classical spring analogy, the potential energy surface for bond angles is less than the surface energy of bond lengths; i.e. the “springs” responsible for angular displacement are significantly looser than those controlling bond elongation. In terms of the energy magnitude, this means it requires less energy to deform a bond angle than to stretch a chemical bond, which is a physical property that our weighted hessian matrix correctly captures across all molecules we tested.

Symmetry and Degeneracy

For highly symmetric molecules, our solver module displayed the mathematical robustness of our diagonalization algorithm. We observed distinct sets of identical frequencies for ammonia and methane at different mode configurations, a phenomenon known as degeneracy. This reflects the triangular pyramidal symmetry of ammonia and the tetrahedral geometry of methane.

In these molecular models, atoms attached to the central atom (C, N), for instance, ‘hydrogen A’ and ‘hydrogen B’ can appear physically indistinguishable. Since they appear in identical chemical environments, any vibrational mode involving them encounters equivalent restoring force, hence exhibiting similar frequencies, even though they all occupy distinct equilibrium positions in space. This is highlighted by the fact that our solver computed distinct eigenvectors (motions) that yielded the same exact eigenvalues (frequencies), which preserved molecular symmetry for both linear and non-linear structures.

Theoretical limitations and accuracy

Looking at the absolute frequency values in comparison to experimental frequencies, there’s a significant overestimation, with deviations ranging from 39-59% for linear and non-linear molecules for their symmetric and asymmetric stretching modes. It is crucial to note that this is not a computational error in our C++ implementation, but rather a theoretical limitation of the model we used. Our codebase employs the Hartree-Fock theory with a minimal basis set (STO-3G).

Historically, the Hartree-Fock model is inherently considered a “stiff” model because it tends to restrict electron flexibility, which results in molecular environments where the potential surface energy is steeper than reality. Steeper potential wells produce higher force constants and, therefore, exaggerate frequencies. Alternatively, our solvers fared more accurately for bending modes, with deviations ranging from 10-25%. This suggests that while the minimal basis set struggled to describe electron density changes during bond stretching, it captured angular alternations with reasonable accuracy.

Conclusion

In this project, we successfully developed a computational chemistry program that computes, analyzes, and produces the files required to visualize the vibrational normal modes of molecular systems. From a list of atoms and atomic coordinates, our C++ implementation can construct the necessary electronic structure components, optimize the molecular geometry, build the mass-weighted Hessian matrix, solve the associated eigenvalue problem, and produce the visualization files. This end-to-end workflow demonstrates that our program functions as a complete tool for vibrational analysis.

The correct identification of translational, rotational, and vibrational modes confirmed the successful implementation of our program. For linear molecules, we observed five zero-frequency modes. For non-linear molecules, we observed six zero-frequency modes. This is consistent with the expected results and shows that our code correctly distinguished rigid-body motions from true vibrational modes. This behavior validates that the mass-weighted Hessian was properly constructed.

Our visualizations were also in line with the expected results, which further demonstrates that our calculated modes were physically meaningful. We tested several molecules and observed that bending modes consistently appeared at lower frequencies than stretching modes. This reflects the relative stiffness of angular versus bond-length restoring forces. For symmetric molecules such as NH_3 and CH_4, our code correctly reproduced degenerate vibrational modes, indicating that molecular symmetry was preserved during diagonalization.

Although the vibrational frequencies were overestimated when compared to experimental values, particularly for stretching modes, we attribute this to the limitations of the CNDO/2 method rather than errors in the numerical implementation (Note: In some runs, we observed values closer to the experimental results, but some of the associated visualizations were incorrect). The reasonable agreement observed for bending modes further supports the correctness of the computational pipeline while highlighting the known trade-offs between computational efficiency and physical accuracy.

Overall, our project met its design goals and provides a practical research tool for exploring molecular vibrations. By combining numerical rigor with intuitive visualization, the software offers valuable insight into molecular structure, stability, and symmetry, laying a strong foundation for more advanced electronic structure and vibrational analyses in future work.

Future work

The primary limitation identified was the systematic overestimation of vibrational frequencies. This is a byproduct of the Hartree-Fock-derived CNDO/2 model using a minimal basis set. By treating electron clouds as a static mean field, the model unfortunately produces a potential energy surface that is "stiff", resulting in higher force constants. For future work, we would like to implement Density Functional Theory (DFT) or use larger polarized basis sets (e.g., 6-31G*). These methods take into account electron correlation, which would shorten the potential energy wells, hence reducing frequency errors, and bringing alignment close to experimental values.

In addition, another critical limitation of the current implementation is its limited support of atoms. To ensure a robust tool that can provide insights to numerous systems, the program must be extended to support every element on the periodic table. Further, once extended, more care will need to be paid to the geometry optimization. Vibrational analysis is sensitive to the precise location of the molecule on the potential energy surface, so ensuring the geometry optimization successfully pushes the molecules to the local minimum is a critical component of a properly functioning vibrational analysis code.

References

- Pople, J. A., & Segal, G. A. (1966). Approximate Self-Consistent Molecular Orbital Theory. III. CNDO Results for AB_2 and AB_3 Systems. J. Chem. Phys., 44, 3289.
- Wilson, E. B., Decius, J. C., & Cross, P. C. (1955). Molecular Vibrations: The Theory of Infrared and Raman Vibrational Spectra. McGraw-Hill.
- Experimental frequencies: https://www.chem.purdue.edu/jmol/vibs/
- CHEM-279 Lecture & Homework materials
- Code Review & Troubleshooting: Google Gemini Pro

Appendix

Supporting materials

- Group project GitHub link: https://github.com/pgraggs247/Chem_279-Final-Project/

Contributions

- Independent implementations of all critical code were written by both Paul and Festo. The final version chosen was Festo's.
- Intensive collaboration on the slideshow presentation. Final Visualizations are Festo's via PyMOL (Paul used Avogadro, and the visualizations were not as good).
- Full collaboration on the final project write-up. Paul focused more on the theoretical background of the normal modes. Festo focused on running the code, interpreting the results, and visualizing in PyMOL.
- We collaborated on debugging the code and ensuring error-free code, and completed the write-up collaboratively.