

# Calculation of Molecular Vibrational Normal Modes

Benjamin Rosman  
0407237H

September 4, 2008

## Supervisor

Dr Alex Welte

## Abstract

Normal mode analysis provides a vital key to understanding the dynamics of a complicated system. In this case, this is the motion and vibrations of the atoms in a molecule. It is shown in several test cases that the algorithm successfully detects every normal mode of the molecule, as well as all rigid body rotations and translations. The frequencies of these vibrations are also computed to within 6% of the values reported in the literature.

## 1 Introduction

The energy and wave function describing the states of a molecule are computed by solving the time-independent Schrödinger Equation. However, for large molecules this can involve finding the solution to partial differential equations of several hundred variables, which can become completely infeasible. This calculation can be simplified by use of the Born-Oppenheimer approximation, which allows for the electronic and nuclear problems to be solved independently [Griffiths 2005].

Research is currently being conducted into different ways of solving the electronic problem. The work presented in this document serves to complement that, by discussing the development of tools which would allow for solving the nuclear component of the molecular energy problem, assuming that the electronic problem had already been resolved. The output from the electronic problem is a grid of potentials, which is not something that is directly measurable in a molecule. Instead observables such as frequencies are calculated by this software.

Section 2 provides a discussion of normal modes and their particular relevance to quantum chemistry, including the method used in their computation. A high-level overview of the code that was developed as a deliverable for this project is given in Section 3. The documented code itself appears in the Appendix. Section 4 illustrates the use of the coded tools with some examples: both a synthetic test case and some applications using real data.

## 2 Normal Modes

### 2.1 Definition

The normal modes of a system are the vibrational motions of the system, such that each coordinate of the system oscillates under simple harmonic motion with the same frequency. These normal modes are all linearly independent and form an orthonormal basis, and as a result, any state of the system can be represented by a linear combination of its normal modes. They are thus used to decompose and understand motion near the equilibrium of a system, which may be very complex in nature.

A normal mode is a vibration in a system such that each coordinate  $q_i$  undergoes motion described by

$$q_i = A_i \sin(\omega t + \phi_i) \quad (1)$$

where the amplitudes  $A_i$  and phase  $\phi_i$  vary between each coordinate involved in the motion, but the frequency  $\omega$  of the oscillations is common to all motion within that system. A direct result of this definition is that all components of the system pass through their equilibrium position simultaneously. Furthermore, the centre of mass of the entire system remains fixed in place with these vibrations. Equation (1) is known as an ansatz, and the normal modes of a system are found by requiring that the solutions take this form.

Normal modes provide a valuable technique for analysis of very complicated systems, and are used in numerous fields of science and engineering. Complex dynamics which may not otherwise be understood can be decomposed into component harmonic oscillations. The fact that any motion can be represented by a superposition of normal modes is a consequence of the linearity of the equations of motion from which they are calculated. This property of normal modes means they are not just a special case of motion, but rather a key to the whole problem of the dynamics of a system.

A failure to examine the normal modes of a structure can often have disastrous consequences. When the Millennium Bridge across the Thames River in London was opened to the public in June 2000, it was discovered that it underwent a swaying motion as pedestrians crossed it. The vibrations from people walking over it caused oscillations in the bridge, which in turn amplified the effect as the people swayed with the bridge. This feedback occurred because the vibrational modes of the bridge were excited by the pedestrians, contrary to the expectations of the bridge designers [Dallard *et al.* 2001]. The problem was eventually solved by constructing dampers at various positions on the bridge.

### 2.2 Computation

Normal modes are defined by the ansatz (1) as the harmonic oscillation around a point of equilibrium of multiple coordinates of a system. Considering the second derivative of this leads to the differential equation

$$\ddot{q} = -\omega^2 q \quad (2)$$

where  $q$  is a vector which gives the Cartesian displacements of the  $N$  elements of the system as  $q = (x_1, y_1, z_1, x_2, y_2, z_2, \dots, x_N, y_N, z_N)$ . The relative amplitudes  $A_i$  in (1) of the vibrating coordinates are determined by the initial conditions to this equation.

The dynamics of a system are governed by Lagrange's equation which tracks the evolution of space coordinates over time, given the kinetic and potential energy of the system. Lagrange's equation is expressed as

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = 0 \quad (3)$$

where  $L = T - U$ . Any motions of the system can be calculated by solving this differential equation after inserting the kinetic energy  $T$  and the potential energy  $U$ .

In Cartesian coordinates, the kinetic energy of a mass  $M$  has the standard form of

$$T = \frac{1}{2}M\dot{q}^2 \quad (4)$$

The potential energy  $U$  can be any general function. However, working in the regime of small oscillations, it can be assumed that only small displacements  $\delta = q - q_0$  from the equilibrium configuration  $q_0$  with minimum potential energy are considered. As a result of this definition of  $q_0$ ,  $U'(q_0) = 0$  and  $U''(q_0) > 0$ . Also,  $U(q_0)$  is arbitrarily defined and so can be neglected.

Now, because only small displacements are considered, all terms of order higher than quadratic are discarded in the Taylor expansion of  $U$ , resulting in the approximation

$$U = \frac{1}{2}Kq^2. \quad (5)$$

where

$$K = \begin{pmatrix} k_{xx}^{11} & k_{xy}^{11} & k_{xz}^{11} & \cdots & k_{xz}^{1N} \\ k_{yx}^{11} & k_{yy}^{11} & k_{yz}^{11} & \cdots & k_{yz}^{1N} \\ k_{zx}^{11} & k_{zy}^{11} & k_{zz}^{11} & \cdots & k_{zz}^{1N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ k_{zx}^{N1} & k_{zy}^{N1} & k_{zz}^{N1} & \cdots & k_{zz}^{NN} \end{pmatrix} \quad (6)$$

and  $k_{ab}^{ij}$  is the force on atom  $i$  in the  $a$  direction when atom  $j$  is moved in the  $b$  direction.

The premise of this small oscillations argument is thus that both the kinetic and potential energy can be written as quadratic functions in terms of the displacements and velocities.

Substituting energy expressions (4) and (5) into (3) yields the equation of motion

$$M\ddot{q} = -Kq. \quad (7)$$

Then, combining (2) and (7) gives

$$Kq = \omega^2 Mq. \quad (8)$$

This can be further simplified by converting to mass-weighted coordinates by making the transformations

$$\bar{x}_i = \sqrt{m_i}x_i \quad (9)$$

$$\bar{y}_i = \sqrt{m_i}y_i \quad (10)$$

$$\bar{z}_i = \sqrt{m_i}z_i \quad (11)$$

$$\bar{k}_{ab}^{ij} = \frac{k_{ab}^{ij}}{\sqrt{m_i}\sqrt{m_j}}. \quad (12)$$

The equation of motion (8) can now be written as

$$\bar{K}\bar{q} = \omega^2\bar{q} \quad (13)$$

which is an eigenvalue problem. Each normal mode is given by an eigenvector  $q$  which describes the displacements from equilibrium of each coordinate in the system. The frequency  $\omega$  of each normal mode is obtained from the corresponding eigenvalue  $\lambda = \omega^2$ .

## 2.3 Classical vs Quantum Mechanics

There is an important fundamental difference between properties of particles in the sense of Classical Mechanics, and their interpretation in Quantum Mechanics. Classically, when one discusses the motion of a particle, one would refer to a trajectory which the particle traverses over time. This is not the case in Quantum Mechanics. In this situation, the position of a particle is defined by a wave function, which is in turn obtained from the solution to Schrödinger's Equation [Griffiths 2005],

$$i\hbar \frac{\partial \Psi}{\partial t} = -\frac{\hbar^2}{2m} \frac{\partial^2 \Psi}{\partial x^2} + V\Psi \quad (14)$$

for some potential function  $V(x, t)$ .

The wave function instead defines a probability distribution of where that particle may be located at a point in time. This distribution then evolves over time.

Furthermore, the ground state of a Classical oscillator such as a suspended pendulum would be the equilibrium position with minimum potential energy. In this case, the pendulum would not be in motion. The equivalent Quantum oscillator in the ground state is instead still defined by a wave function. As a result, the ground state does not consist of just a single position, but a probability distribution which can be approximately modeled as a Gaussian distribution.

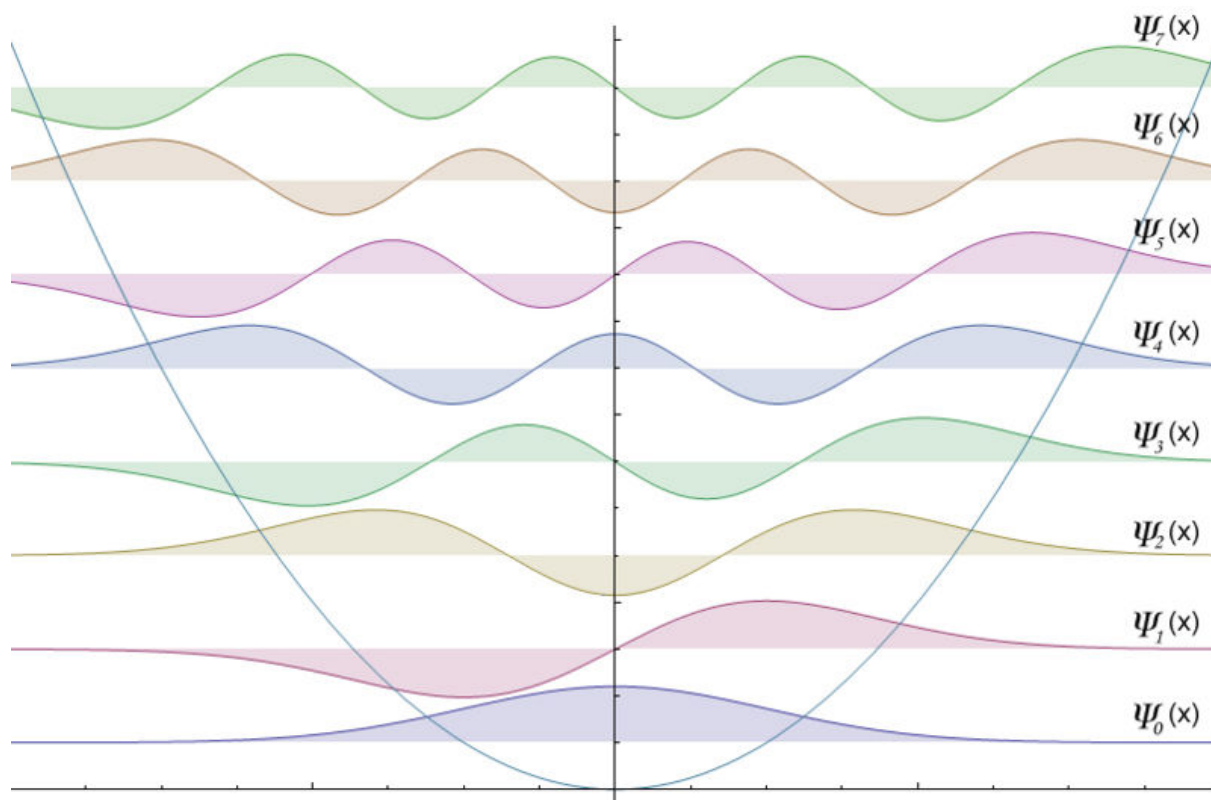


Figure 1: Constant energy gap spacing

A molecule in a ground state may be pushed into an excited state. Higher excited states are reached by repeatedly increasing the energy of the molecule by  $\hbar\omega$ . This is because, when the potential is quadratic, the energy spacing between excited states is constant. These gaps can be seen in figure 1 which shows the wave functions at various excited states of a molecule. With each higher excited state,

the distribution given by the wave function becomes increasingly spread and gains an extra mode. The state of the molecule can thus be tested by probing the molecule with photons. These photons will only be absorbed at a frequency corresponding to one of these states. They can thus characterise the state of the molecule.

It is important to note that molecular vibrations occur within the infrared and microwave spectrum and are thus invisible to the human eye.

## 2.4 Molecules

In the specific context of Quantum Chemistry, normal modes are important to understand the internal structure of molecules. A normal mode of a molecule describes the ways in which the nuclei of the constituent atoms of a molecule interact and vibrate around the most stable configuration (lowest potential energy) in a ground state.

Let a molecule consist of  $N$  atoms. Therefore in order to represent the state of this in Cartesian coordinates,  $3N$  coordinates are required. However in a generalised sense, 3 coordinates are required to describe the translation of the molecule in space, and another 3 for its rotation. Normal modes are insensitive to the position and orientation of the molecule, and as a result there are  $3N - 6$  normal modes for a general molecule. In the special case of a linear molecule, the orientation of the molecule is described by only two angles, as rotation about the axis of the molecule cannot be observed, leaving  $3N - 5$  normal modes.

The state of a molecule is usually described in an ad hoc manner in terms of internal coordinates. Although these can be defined in any arbitrary manner, they generally represent physical aspects such as bond lengths and angles. This has the advantage of discarding the extraneous details of the rigid body motion, the translational and rotational information.

Internal coordinates are often used in the generating of data to minimise the number of required computations. Consider the case of a simple diatomic molecule. Using internal coordinates, the data can be generated by varying a single parameter: the bond length. However, in Cartesian coordinates, six parameters would be required: an x, y and z coordinate for each atom. As rotations and translations do not affect the vibrations, the use of internal coordinates results in a drastic decline in the number of parameters with which to experiment, without loss of information about the internal dynamics.

Normal mode calculations can be perhaps most systematically performed in Cartesian coordinates. The reason for this is that in Cartesian coordinates, the kinetic energy is represented in a standard form as  $T = \frac{1}{2}M\dot{q}^2$ , where  $q$  is the set of coordinates. This means that the required coefficients of the velocities are merely the masses of the atoms. When using an ad hoc coordinate system, the formulation of  $T$  may not be so straightforward.

An important consideration is molecular symmetries, which are defined in terms of group theory. These are used to classify molecules in terms of their axes of rotation and reflection. However, this is an analytical approach to the problem and has not been considered in this project.

The vibrations of a molecule are not directly observable as in the case of classical mechanics. They are probed by means of bombarding the molecule with photons through a process such as Infrared or Raman spectroscopy.

The Born-Oppenheimer approximation allows for the incredibly complicated calculations required to understand the workings of a molecule to be decomposed into the electronic structure problem and the nuclear problem. The electronic problem models interactions of electrons with each other and the atomic centres of the molecule. The result is that an overall potential energy for the molecule can be computed for each geometrical positioning of the nuclei. However, this potential energy grid which is provided to the nuclear problem actually contains kinetic energies of the electrons, but can be regarded as a potential for the purposes of a tool solving the nuclear problem. This problem of the dynamics of

the atomic nuclei is tackled by this project.

## 3 Program Code

### 3.1 Goals

This code is required to analyse the interactions between the constituent atoms of a molecule. A requirement of this is that the problem of the interactions between the electrons of the molecule has already been solved. The output of this solution would be a grid consisting of the energies of a number of geometrical configurations of the nuclei in the molecule. This output energy surface is a function of the repulsion between electrons, the attraction between electrons and protons in the nuclei, as well as the repulsion between the different atomic centres.

Input to the tools developed here is a file describing these potentials in a molecule. This file consists of the specification of multiple configurations of the molecule in some internal coordinate system, as well as the potential energy of each configuration. Atomic masses of each atom in the input molecule are also required. The output of this code should be all the normal modes of that input molecule, given in terms of displacements of each atom in Cartesian coordinates, as well as the vibrational frequencies of the normal modes.

Also required is a function to convert the internal coordinate system specific to each molecule into the Cartesian coordinates for each atom. Internal coordinates are provided to vastly reduce the number of iterations or experiments that must be performed to produce this data in the first place, and so is acceptable as input.

All development of these functions was done in MATLAB. The interfaces and functioning of each of these methods is described below, and the actual code is available in the Appendix.

### 3.2 MATLAB Functions

#### 3.2.1 NuclearAnalysis

**Interface:**  $[\mathbf{x}_0, \mathbf{W}, \mathbf{NM}] = \text{NuclearAnalysis}(\mathbf{M}, \mathbf{X}, \mathbf{V}, \mathbf{S}, \mathbf{I})$

**Input:** A vector  $\mathbf{M}$  of the  $n$  masses of the atomic nuclei in the molecule (in atomic mass units), a matrix  $\mathbf{X}$  whose rows describe different configurations of the internal coordinates of a molecule, a vector  $\mathbf{V}$  of the corresponding potentials for these configurations in  $\mathbf{X}$ .  $\mathbf{S}$  is a handle to a function which converts internal coordinates into a vector giving the coordinates of all the atoms in the molecule. This is used for providing an initial configuration in Cartesian coordinates. Conversely,  $\mathbf{I}$  is a function handle to convert internal coordinates to Cartesian coordinates.

**Output:** A diagonal matrix  $\mathbf{W}$  of the wavenumbers of the system, and a matrix  $\mathbf{NM}$  with each column being the normalised normal mode for the corresponding wavenumber.  $\mathbf{x}_0$  is the typical length scale of the problem, in Angstroms.

**Description:** This is the main function used to perform the analysis. It requires a full description of the molecule to perform the analysis, including data on the potential of various configurations, the masses of the atoms, and the functions  $\mathbf{S}$  and  $\mathbf{I}$  for providing an interpretation of the geometry of the specific coordinate system used for that molecule.

The masses are entered in the form of a vector  $M = [m_A, m_B, m_C, \dots]$ , where  $m_A$  is the mass of the first atom in the molecule,  $m_B$  the mass of the second, etc (chosen arbitrarily). The function  $\mathbf{I}$  is required to produce a vector  $C = [x_A, y_A, z_A, x_B, y_B, z_B, \dots]$  where the order of atoms is consistent with the ordering in the vector  $M$ .

The configurations and potentials are first converted into Cartesian coordinates. This can be done non-uniquely as the potentials of the molecule are not functions of its orientation and translation. As a

result, a quadratic is fit to the provided data in internal coordinates.

An initial configuration in Cartesian coordinates is then chosen by applying the function **S** to the equilibrium coordinates. New Cartesian data points are generated by perturbing the atoms in the molecule and looking up the potentials for these configurations on the fit quadratic. The perturbations are done by individually shifting the coordinates of the atoms in a positive and negative direction by a value  $\delta$ . The shifts are also done in pairs to provide enough data points to fit the cross terms.

After having generated the Cartesian data points, a quadratic is fit to this new data. The curvature matrix (quadratic coefficients) arising from this is then used as the potential matrix for performing the normal mode analysis by the function **NormalMode**. Using the resulting normal modes as well as the initial configuration provided by **S**, the motion of the individual atoms can easily be illustrated. The calculated vibrational frequencies are converted into wavenumbers for convenience. This conversion depends on the units of the original data.

The characteristic length scale  $\mathbf{x}_0$  is defined as  $x_0 = \sqrt{\frac{\hbar}{\mu\omega}}$  where  $\mu$  is the reduced mass for the problem.  $\mathbf{x}_0$  is the standard deviation of the Gaussian which can be used to approximate the wave function of the initial energy surface. This is reported to the user as it can be used to gauge whether the data points are sufficiently close to the equilibrium point. This property has no equivalent in Classical Mechanics.

### 3.2.2 FitQuadratic

**Interface:**  $[\mathbf{A}, \gamma, \beta, \mathbf{n}] = \text{FitQuadratic}(\mathbf{X}, \mathbf{V})$

**Input:** A list of sets of coordinates  $\mathbf{X}$  describing the system state, and the corresponding potentials  $\mathbf{V}$

**Output:** The curvature matrix  $\mathbf{A}$  of the fit quadratic, the equilibrium coordinates  $\gamma$ , the minimum energy  $\beta$ , and the dimension  $\mathbf{n}$  of the fit quadratic

**Description:** This function accepts data describing the energy potential of a molecule for various configurations, and fits a quadratic to that data, of the form  $V = (\mathbf{X} - \gamma)^T \mathbf{A} (\mathbf{X} - \gamma) + \beta$ . The assumption is made that  $\gamma$  and  $\beta$  appear explicitly within matrix  $\mathbf{X}$  and vector  $\mathbf{V}$ . This is a reasonable assumption, as any software that solves the electronic structure problem should find the configuration with minimum energy.  $\mathbf{A}$  is then found by fitting a quadratic to the data using Least-Squares approximations [Burden and Faires 2001], and repeatedly discarding the points furthest from the minimum until the error of the fit is within a tolerance of  $10^{-5}$ . The dimension of the quadratic which is fit to the data, is determined from the dimension of the data itself.

### 3.2.3 NormalMode

**Interface:**  $[\text{freq}, \text{NM}] = \text{NormalMode}(\mathbf{M}, \mathbf{K})$

**Input:** A vector  $\mathbf{M}$  of the  $n$  masses of the atomic nuclei in the molecule, and a  $3n \times 3n$  matrix  $\mathbf{K}$  of force coefficients describing the forces exerted by each atom on each other atom in every Cartesian direction

**Output:** A diagonal matrix **freq** of the vibrational frequencies of the system, and a matrix **NM** with each column being the normal mode for the corresponding frequency

**Description:** The potential coefficient matrix  $\mathbf{K}$  is first converted to mass-weighted Cartesian coordinates by dividing by the root of the corresponding masses. Then the eigenvectors of this matrix give the normal modes, and the eigenvalues are the vibrational frequencies.

## 4 Results

This section describes the use of the MATLAB tools by means of examples. This illustrates the correctness of the output on two real molecules, as well as providing a forum for a discussion of the errors that

are introduced through this numerical method.

The data that were produced as a solution to the electronic structure problem for the HF and H<sub>2</sub>O molecules were generated by GAMESS (General Atomic and Molecular Electronic Structure System) which is a freely available computational chemistry application. The calculations were however somewhat crude and used primarily for proof of concept.

## 4.1 Synthetic Test Case

In order to demonstrate and test the methods, a dummy test case was fabricated. The potentials of this molecule were created from a quadratic function with a small cubic piece

$$V = \epsilon(X - \gamma)^3 + A(X - \gamma)^2 + \beta \quad (15)$$

where  $|\frac{\epsilon(X-\gamma)^3}{A(X-\gamma)^2}| \ll 1$ . The value of  $A$  was chosen to be 2, and the masses of the two synthetic atoms were each chosen to be 1 atomic mass unit, making them equivalent to Hydrogen.

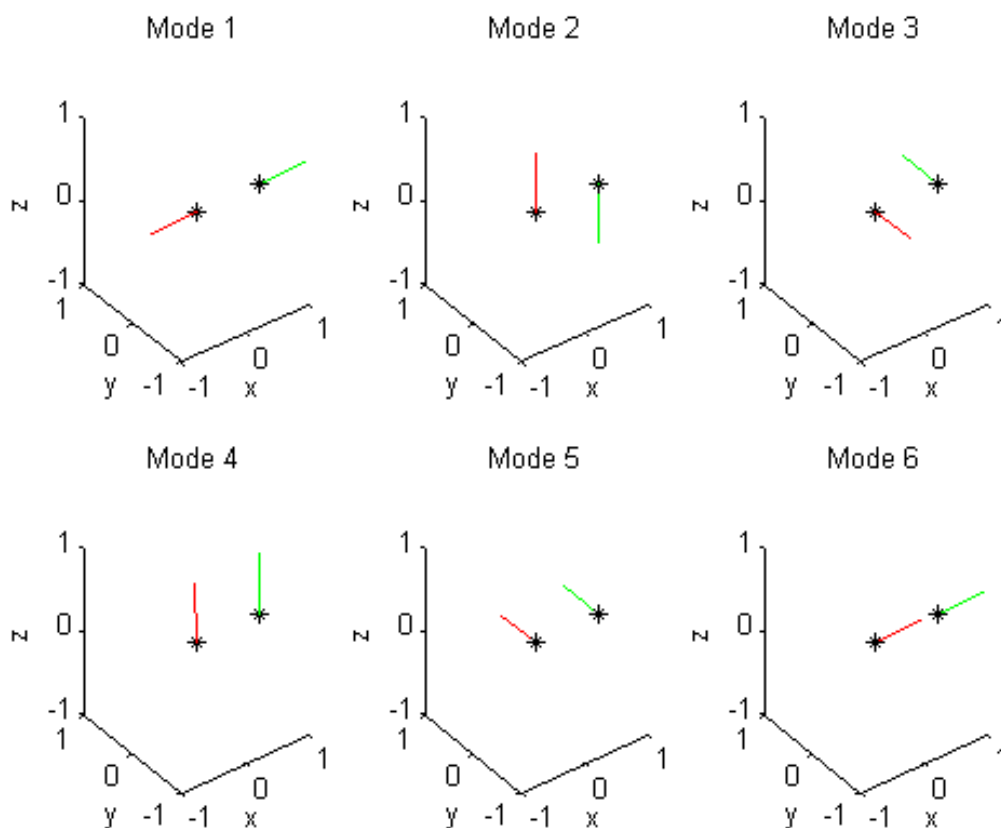


Figure 2: Calculated modes of the synthetic molecule

The function  $S$  giving the initial configuration of the molecule in Cartesian coordinates was specified as  $S(\gamma) = [0, 0, 0, \gamma, 0, 0]$ . This places one atom at the origin, and the second a distance of optimum bond length  $\gamma$  in the positive x-direction. The function  $I$  computes the internal coordinates from the Cartesians by returning the distance between the atoms.



Figure 2 clearly shows the one vibrational normal mode in the first diagram, which illustrates the displacements of the atoms away from each other. The next two images show molecular rotations about the two axes which are perpendicular to the axis of the molecule. These involve the two atoms moving by the same amount in opposite directions. The last three images show translations in the x-, z- and y-directions respectively. This translation is recognised by the fact that both atoms travel the same distance in the same direction.

Now consider the vibrational frequencies of these modes. One would expect that the vibrational mode would have a non-zero frequency while others corresponding to the rotational and translational motions would be negligible. This is indeed the case in the results generated by this tool. The eigenvalues of this problem are shown in Table 1. As can be seen, the rotational and translational frequencies are negligible compared to that of the vibrational mode.

Mode	Vibration	yz Rotation	xy Rotation	x Translation	y Translation	z Translation
Value	3.9913	0.0039	0.0039	-0.0015	-0.0015	-0.0016

Table 1: Eigenvalues from synthetic molecule

Table 2 gives the curvature matrix  $A$  that is returned from the FitQuadratic function, once the single dimensional data provided by equation (15) has been converted into the six dimensions required to represent a diatomic molecule in Cartesian coordinates. This matrix is symmetric and equals the matrix  $K$  of components of the forces the atoms exert on each other. There are four values in this matrix which are not approximately zero. These are the values which correspond to the restoring force  $K_{11}$  on atom 1 as it is moved in the x-direction (along the bond), the symmetric repulsive forces,  $K_{41}$  and  $K_{14}$  of atom 1 on atom 2 in the x-direction and vice versa, as well as the restoring force  $K_{44}$  on atom 2 in the x-direction.

1.9949	0	0	-1.9964	0	0
0	0.0012	0.0000	0.0000	-0.0027	-0.0000
0	0.0000	0.0012	0.0000	0.0000	-0.0027
-1.9964	0.0000	0.0000	1.9949	0	0
0	-0.0027	0.0000	0	0.0012	0.0000
0	-0.0000	-0.0027	0	0.0000	0.0012

Table 2: Curvature matrix  $K$  for synthetic molecule

The two atoms in this synthetic molecule have the same mass, and thus it is logical that the magnitudes of all these non-zero forces are approximately equal. Furthermore, they are approximately equal to the curvature  $A$  defined in equation (15), but are not exact due to the cubic term in that expression, as well as errors that have been introduced during the Least-Squares curve fitting procedure.

## 4.2 Hydrogen Fluoride (HF)

Figure 3 shows the modes that were computed for the HF molecule. In each case, the black asterisks show the configuration of the molecule with minimum energy. The red line then shows the displacement for the Hydrogen atom, and the green line the displacement for the Fluorine atom.

The masses used were one atomic mass unit for the Hydrogen atom, and 19 for the Fluorine atom. Similarly to the synthetic molecule,  $S(\gamma) = [0, 0, 0, \gamma, 0, 0]$ . Again, the function  $I$  returns the distance between the atoms.

Mode 1 shows the vibrational mode of HF. It is observed that the normal mode lies along the axis of the molecule, and the displacement of the Hydrogen is much greater than that of the Fluorine. It is

logical considering that the mass of Fluorine is 19 times that of the Hydrogen. The next two modes are rotational, in the x-z and x-y planes respectively. The final three modes are translational, in the x-, z- and y-directions.

The modes for Hydrogen Fluoride are the same as those in the synthetic test case as they are both diatomic molecules. The difference is that in the case of HF, the two atoms are displaced by different amounts. This is a result of the difference in mass of these two atoms.

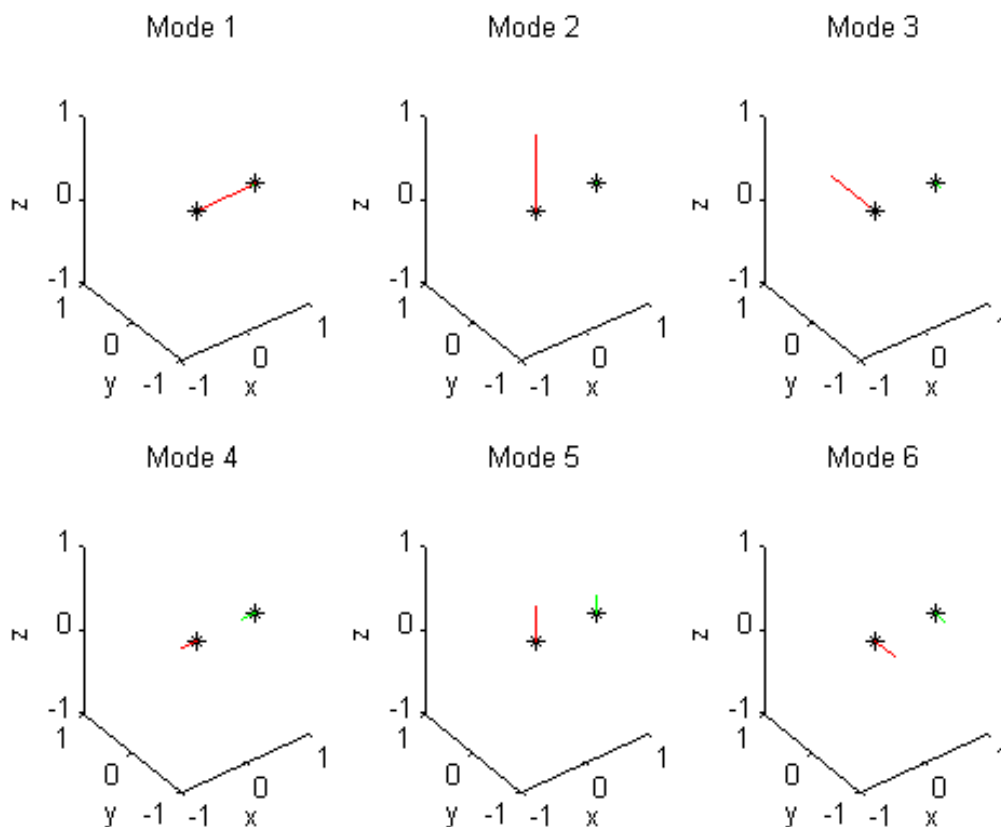


Figure 3: Calculated modes of HF molecule

Mode	Vibration	xz Rotation	xy Rotation	x Translation	z Translation	y Translation
Value	1.1947	0.0009	0.0009	-0.0001	-0.0001	-0.0001

Table 3: Eigenvalues from HF molecule

Table 3 shows the eigenvalues of this system. The only eigenvalue which is not approximately zero is that for the vibrational mode. This agrees with expectation, as this is the only oscillatory motion. Converting the only non-zero eigenvalue into a wavenumber gives a wavenumber of  $3972.9 \text{ cm}^{-1}$ . The actual wavenumber for this normal mode in HF is reported as  $4138 \text{ cm}^{-1}$ . This establishes that the tool underestimated the result and the error of this result is 3.99%. The reduced mass for this problem is also correctly calculated at 0.95.

### 4.3 Water (H<sub>2</sub>O)

The motions of the H<sub>2</sub>O molecule should consist of three vibrational modes, three rotations and three translations. As can be seen in Figure 4 these are all discovered by this tool.

The masses used in this example were 16, 1 and 1 atomic mass units for the Oxygen and two Hydrogen atoms respectively. The initial configuration for the molecule was specified by  $S(\theta, r_1, r_2) = [0, 0, 0, r_1, 0, 0, r_2 \cos \theta, r_2 \sin \theta, 0]$ . The first three of these coordinates places the Oxygen at the origin, the next three position one Hydrogen a distance  $r_1$  along the positive x-axis, and the last three coordinates place the other Hydrogen a distance  $r_2$ , at an angle of  $\theta$  from the first Hydrogen. The function  $I$  calculates  $r_1$ ,  $r_2$  and  $\theta$  from the Cartesian coordinates of the three atoms.

Figure 4 also indicates the geometry of the molecule with minimum potential energy by means of black asterisks. Here, displacements of the Oxygen atom are shown with a red line, and the displacements of the two Hydrogens are in blue and green.

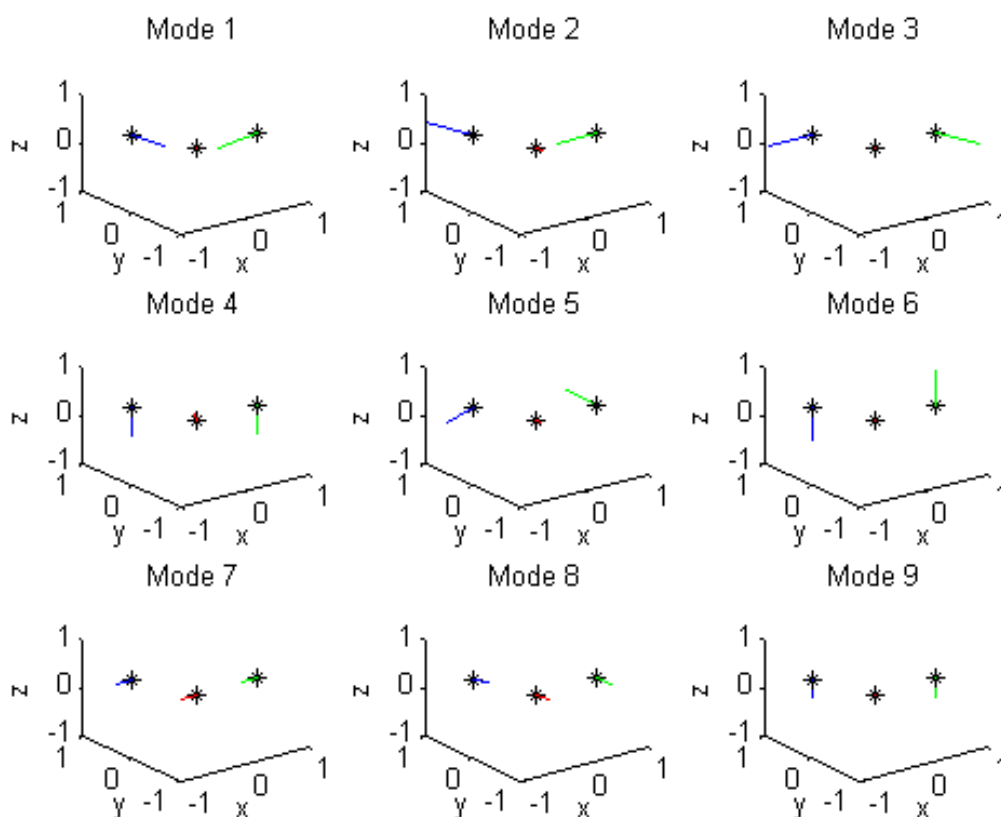


Figure 4: Calculated modes of H<sub>2</sub>O molecule

Modes 1, 2 and 3 show vibrational modes. The first of these is a symmetric stretching of both OH bonds. The second is an asymmetric stretching of the same bonds. Finally, Mode 3 shows the bending mode of the molecule. The subsequent three modes are rotational, and the last three are translations.

The computed vibrational wavenumbers corresponded well with actual experimental values, as is shown in Table 4 [Tennyson 2006]. As can be seen, they all lie within 6% of the actual values, and again are underestimations of the true results.

	Symmetric stretch	Asymmetric stretch	Bending
Actual value (in $\text{cm}^{-1}$ )	3657.05	3755.93	1594.75
Computed value (in $\text{cm}^{-1}$ )	3853.5	3836.8	1686.0
Error	5.37%	2.15%	5.72%

Table 4: Vibrational wavenumbers for H<sub>2</sub>O

## 5 Conclusion

Determining the normal modes of any complex system is an approach that is vital for the understanding of its dynamics. This is especially the case for a quantum system such as a molecule. The interactions between the nuclei of numerous atoms can only be analysed by decomposing their motions into normal modes.

The deliverable of this project was a MATLAB tool which would perform such a normal mode analysis from a given energy surface derived from the solution to the electronic structure problem of that molecule. As was shown through the use of several examples, the tool was effective at detecting all the normal modes of molecules, as well as the rigid body translations and rotations thereof. The vibrational frequencies were also shown to correlate well with known values.

The wavenumbers obtained in the results in Section 4 are shown to be within 6% of the values that are reported in the literature. Although this is very close, these results could coincide better. The reason for this is largely twofold. Firstly, the data from the electronic problem was not calculated to a high degree of precision. This was done mainly to reduce computation times, and it was mainly required as just a proof of concept.

The other reason is that the approach taken was purely numerical in nature. No advantage was taken of certain properties of the molecules, such as their inherent group symmetries. This provides scope for future work on this topic.

The code developed here has been made available for inclusion into a broader suit of tools under development, which include methods for solving problems such as the electronic structure problem. It is the hope of the author that these will provide an efficient and generalised package which can be used to gain greater insights into the dynamics of the quantum world.

## Acknowledgments

The author wishes to thank Alex Welte for the many consultations, discussions and advice which contributed to this project. Thanks also to Tom McWalter for the additional assistance provided, as well as generating all the data from the electronic problem, which was used to test these tools.

## References

- [Burden and Faires 2001] R.L. Burden and J.D. Faires. *Numerical Analysis*. Brooks/Cole, 7th edition, 2001.
- [Dallard *et al.* 2001] P. Dallard, A.J. Fitzpatrick, A. Flint, S. Le Bourva, A. Low, R.M. Ridsdill Smith, and M. Willford. The London Millennium Footbridge. *Structural Engineer*, 79(22):17–35, November 2001.
- [Griffiths 2005] D.J. Griffiths. *Introduction to Quantum Mechanics*. Pearson Prentice Hall, 2nd edition, 2005.

[Tennyson 2006] Jonathan Tennyson. Calculating the vibration-rotation spectrum of water. *Physica Scripta*, 73:53–56, 2006.

## Appendix

### Function NuclearAnalysis

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Function NuclearAnalysis
% -----
% This function performs a multidimensional quadratic fit to an
% energy surface defined by the potentials V at configurations X
% The quadratic is of the form  $V = (X-\gamma)'*A*(X-\gamma) + \beta$ 
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [x0, W, E] = NuclearAnalysis(atoms, X, V, S, internal)

% define the mass of a proton
pm = 1.67e-27;
% define the reduced Planck constant
hbar = 1.05457e-34;

% fit a quadratic to the input data in internal coordinates
[A, gamma, beta, n] = FitQuadratic(X, V);
% define a function C to be the fit quadratic
C = @(x) (x'-gamma)'*A*(x'-gamma) + beta;

% m = the number of atoms in the molecule
m = length(atoms);

% Sp = the equilibrium configuration of the molecule
Sp = S(gamma);

% initialise set of Cartesian perturbations with the equilibrium
% configuration
PX = Sp;
PV = C(internal(Sp));

% define the perturbations
delta = 0.02;
shift = [delta -delta 2*delta -2*delta];
% coord = number of coordinates
coords = length(Sp);

% generate perturbations of the Cartesian coordinates, two at a time
for i = 1:coords
    for g = 1:length(shift)
        delta1 = shift(g);
        % first perform xi^2 perturbations
        temp = zeros(1, coords);
        temp(i) = delta1;
        PX = [PX; Sp+temp];
    end
end
```

```

    % perform a lookup into C of the potential at that point
    tempint = internal(Sp+temp);
    lookup = C(tempint);
    PV = [PV; lookup];

    for j = i+1:coords
        for h = 1:length(shift)
            delta2 = shift(h);
            % generate xi*xj perturbations
            tempj = temp;
            tempj(j) = delta2;
            PX = [PX; Sp+tempj];
            % perform a lookup into C of the potential
            tempint = internal(Sp+tempj);
            lookup = C(tempint);
            PV = [PV; lookup];
        end
    end
end
end

% fit a quadratic to the Cartesian data points
[AP, gammaP, betaP, nP] = FitQuadratic(PX', PV');

% perform the normal mode analysis using curvature matrix AP
[v, E] = NormalMode(atoms, AP);

% adjust for mass weighted coordinates
a = length(Sp)/3;
b = 3;
Et = E*3;
for i = 1:3:3*length(atoms)
    E(i:i+2,:) = E(i:i+2,)/sqrt(atoms(fix(i/3)+1));
end

% draw the normal modes
for drawmode = 1:length(Sp)
    i = length(Sp) - (drawmode-1);
    subplot(a,b,drawmode),plot3(Sp([1,4]),Sp([2,5]),Sp([3,6]),'k*')
    hold on
    if m == 3
        subplot(a,b,drawmode), plot3(Sp(7), Sp(8), Sp(9), 'k*')
    end
    subplot(a,b,drawmode), axis([-1 1 -1 1 -1 1])
    subplot(a,b,drawmode), xlabel('x')
    subplot(a,b,drawmode), ylabel('y')
    subplot(a,b,drawmode), zlabel('z')
    subplot(a,b,drawmode), title(['Mode ', int2str(drawmode)])
end

```

```

px = [Sp(1) Sp(1)+E(1,i)];
py = [Sp(2) Sp(2)+E(2,i)];
pz = [Sp(3) Sp(3)+E(3,i)];
subplot(a,b,drawmode), plot3(px,py,pz,'r')

px = [Sp(4) Sp(4)+E(4,i)];
py = [Sp(5) Sp(5)+E(5,i)];
pz = [Sp(6) Sp(6)+E(6,i)];
subplot(a,b,drawmode), plot3(px,py,pz,'g')

if m == 3
    px = [Sp(7) Sp(7)+E(7,i)];
    py = [Sp(8) Sp(8)+E(8,i)];
    pz = [Sp(9) Sp(9)+E(9,i)];
    subplot(a,b,drawmode), plot3(px,py,pz,'b')
end
end

% calculate the reduced mass for the system
M = AP/(v(end,end));
reducedmass = M(1)*pm;
% calculate the typical length scale x0
x0 = sqrt(hbar/(reducedmass*sqrt(v(end,end))));
% convert the frequencies into wavenumbers
W = sqrt(1.3211e7*v);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This helper function converts from degrees to radians
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function out = deg2rad(in)

out = in/180*pi;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This helper function converts from radians to degrees
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function out = rad2deg(in)

out = in/pi*180;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

**Function NormalMode**

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```



```

% Function NormalMode
% -----
% This function calculates normal modes and their wave numbers
% from a vector of masses M and a matrix of force coefficients K
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [freq, modes] = NormalMode(M, K)

% obtain the number of atoms in the molecule, n
n = length(M);
% obtain the dimension of the problem (typically 3)
m = length(K)/n;

% generate a matrix of masses
M1 = repmat(reshape(repmat(M,m,1),m*n,1),1,m*n);
M2 = M1';
% convert the matrix K to mass-weighted coordinates
K = K./sqrt(M1.*M2);

% determine the eigenvalues and eigenvectors (normal modes)
% of the matrix K
[modes, evalue] = eig(K);

% return the frequencies
freq = evalue;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

## Function FitQuadratic

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Function FitQuadratic
% -----
% This function performs a multidimensional quadratic fit to an
% energy surface defined by the potentials V at configurations X
% The quadratic is of the form  $V = (X-\gamma)'*A*(X-\gamma) + \beta$ 
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [A, gamma, beta, n] = FitQuadratic(X, V)

% n = number of dimensions to data
n = size(X,1);

% find minimum energy beta, at coordinates gamma
[beta, minPos] = min(V);
gamma = X(:,minPos);

% subtract gamma, beta from data to centre it at the origin
for i = 1:n
    X(i,:) = X(i,:) - gamma(i);
end

```

```

V = V - beta;

% select an initial estimate for the curvature coefficients
a = zeros(1,(n+1)*n/2);

% ensure loop doesn't terminate immediately
resnorm = Inf;

% repeat until required accuracy or too few data points
while resnorm > 10^-5 & size(X,2) > n
    % m = number of data points
    m = size(X,2);
    % fit curve to current values using Least Squares
    a0 = a;
    [a, resnorm] = lsqcurvefit(@quadform, a0, X, V);

    % choose the minimum point on the quadratic
    [ig, pmX] = min(V);
    mX = X(:,pmX)';

    % remove furthest points from mX from set
    D = dist(mX, X);
    [resD, ordD] = sort(D);
    perc = 0.01;
    keep = ceil(perc * m);
    newOrd = sort(ordD(1:end-keep));
    newX = X(:,newOrd);
    newV = V(newOrd);
    X = newX;
    V = newV;
    % repeat fit with fewer points to try improve error

end

% convert the quadratic coefficients into a symmetric matrix
sA = [];
q = 1;
for i = 1:n
    sA = [sA; zeros(1, i - 1) a(q:q+(n-i))];
    q = q + n - i + 1;
end
A = (sA + sA')/2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This helper function defines the required form of the quadratic
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function F = quadform(a, v)

```

```

n = size(v,1);
F = 0;
k = 1;
for i = 1:n
    for j = i:n
        F = F + a(k)*v(i,:).*v(j,:);
        k = k+1;
    end
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This helper function calculates the distance between two points
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function D = dist(x, y)
X = repmat(x', [1 size(y,2)]);
D = sqrt(sum((X - y).^2,1));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```