

Normal Mode Analysis of the Clathrin Cage
and Other Semi-Rigid Systems

A Thesis
Presented to
The Division of Mathematics and Natural Sciences
Reed College

In Partial Fulfillment
of the Requirements for the Degree
Bachelor of Arts

Drew Skillman

May 2002

Approved for the Division
(Physics)

Richard Crandall

This thesis is dedicated to my parents. Your limitless support has been my greatest asset.

I would also like to thank my advisor, Richard Crandall. In addition, advice from other Reed professors including Ethan Jackson, John Powell, Morgan Mitchell, Jerry Shurman, and Nicholas Wheeler helped me overcome a variety of obstacles.

Finally, I would like to thank my officemates, Tyson and Jeffrey, who reminded me that, occasionally, there are more important things to do than sleep.

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 1 |
| 2 | The Clathrin Cage | 3 |
| 2.1 | The clathrin triskelion and endocytosis | 3 |
| 2.2 | The relevance of dopamine reuptake | 6 |
| 2.3 | Clathrin cage vibration | 9 |
| 3 | Analytic Approach to Coupled Oscillations | 11 |
| 3.1 | Introduction | 11 |
| 3.2 | The general approach | 12 |
| 3.3 | Example: the triangle | 15 |
| 3.3.1 | Determining the potential energy function | 16 |
| 3.3.2 | Computing \mathbb{H} and its eigensystem | 17 |
| 3.3.3 | The general solution | 19 |
| 4 | Numerical Approach to Coupled Oscillations | 23 |
| 4.1 | Mass-spring implementation | 28 |
| 4.1.1 | Defining constants and initial conditions | 29 |
| 4.1.2 | Determining the force function | 30 |
| 4.1.3 | Implementing $RK4$ | 31 |
| 4.2 | Example: the triangle | 32 |

| | | |
|--------------------|---|-----------|
| 4.2.1 | Determining $\mathbf{F}(\mathbf{r})$ for the triangle | 33 |
| 4.2.2 | Checking energy conservation | 33 |
| 4.2.3 | Comparing the results | 34 |
| 5 | The Vibrating Tetrahedron | 41 |
| 5.1 | The analytic solution | 41 |
| 5.2 | The computational solution | 44 |
| 6 | Semi-Rigid Systems | 49 |
| 6.1 | The square | 49 |
| 6.2 | The cross-braced cube | 54 |
| 6.2.1 | Determining the potential | 55 |
| 7 | Normal Modes of the Clathrin Cage | 59 |
| 7.1 | Finding the equilibrium position | 60 |
| 7.2 | Analytic results | 61 |
| 7.3 | Computational results | 65 |
| 8 | Conclusion | 69 |
| Appendix A: | | |
| | Algebraic Considerations | 73 |
| Appendix B: | | |
| | The Mathematica Code | 75 |

Constants and Variables

For easy reference, the significance of important variables and constants appears here. The mass of each vertex, the force constant for each spring, and the equilibrium length of each spring is set to 1 for all of the systems considered. Whenever possible, the notation used here is preserved in the *Mathematica* code which appears in the appendix.

Notation used in both methods

- m \longrightarrow mass of each vertex $\longrightarrow 1$
- k \longrightarrow force constant of each spring $\longrightarrow 1$
- d \longrightarrow equilibrium length of each spring $\longrightarrow 1$
- N \longrightarrow the number of vertices in the system

Notation used in the analytic method

- \mathbf{R} \longrightarrow the set of vectors which designate the equilibrium configuration of a system
each entry takes the form $\mathbf{R}_i = (x_i, y_i, z_i)$
where i counts from 0 to $N - 1$
- \mathbf{r} \longrightarrow the set of vectors which designate the displaced position of each vertex
each entry takes the form $\mathbf{r}_i = (q_{(3i)}, q_{(3i+1)}, q_{(3i+2)})$
where i counts from 0 to $N - 1$
- V \longrightarrow the potential energy of the system as a function of \mathbf{r} .
- \mathbb{H} \longrightarrow the Hessian matrix: $\left. \frac{\partial^2 V}{\partial q_j \partial q_k} \right|_0$
- ω_r \longrightarrow the normal frequency corresponding to the
square root of the r^{th} eigenvalue of \mathbb{H}
- \mathbf{a}_r \longrightarrow the matching eigenvector of \mathbb{H}

Notation used in the computational method

- h \longrightarrow integration time step
- t_f \longrightarrow target integration time
- n \longrightarrow the number of steps made in the integration ($n = \frac{t_f}{h}$)
- \mathbf{r} \longrightarrow the set of vectors which designate the position of each vertex
each entry takes the form $\mathbf{r}_i = (x_i, y_i, z_i)$
- \mathbf{v} \longrightarrow the set of velocity vectors for each vertex
each entry takes the form $\mathbf{v}_i = (v_{xi}, v_{yi}, v_{zi})$

A clarification:

The combination of analytic and computational approaches makes the notation tricky. Rather than introducing another variable, we let \mathbf{r} assume two different roles, one for each method. Because the two methods are not treated concurrently, the inconvenience caused by this overloading should be minimal.

Abstract

Normal modes in a triangle, tetrahedron, cube, and a mass-spring model of the clathrin cage are analyzed using two complementary approaches: classical Lagrangian mechanics and a fourth-order Runge-Kutta integration method. The two approaches yield identical results in the small oscillation regime and differ when nonlinearities are introduced by larger amplitude oscillation. The cube and the clathrin cage exhibit semi-rigid behavior, and an index for characterizing their flexibility is introduced by considering the number of zero-frequency modes which correspond with deformations rather than simply translation and rotation. A frequency spectrum of the clathrin cage is obtained but the numerical solution suggests the results may be problematic.

Chapter 1

Introduction

Thanks to the ever increasing power and availability of personal computers, computational physics has become a common tool used to explore the behavior of complex systems. One advantage of this approach is that it allows the experimenter to control the physical forces involved, the initial conditions, and details of the system's dynamics without the restrictions often imposed by an analogous mathematical treatment. When combined with effective visualization and data analysis techniques, numerical simulations provide insight into problems which cannot be easily considered algebraically. Equally important, however, is the computational method's ability to explore the limits of a known algebraic solution. One class of problems for which algebraic and numerical solutions can be found and compared are those involving coupled oscillations.

The classical treatment of coupled oscillators was introduced by Bernoulli in 1753, and was later refined by Lagrange in the 1760's. Although it is robust and thoroughly studied, this approach is only applicable to very small vibrations. A parallel treatment of these systems using computational methods avoids such a restriction, and is therefore useful in assessing the classical solutions.

In order to appreciate why coupled oscillations can be computationally favorable, consider the problem of N bodies influencing each other via gravity. Computing the

net force on each body requires $N - 1$ calculations, so computing the net force on all bodies requires $(N - 1)^2$ calculations. A simulation of 1000 bodies will require 10^6 calculations for each time step. After the net force for each body is known, a variety of integration methods exist that increment the time forward with varying degrees of precision. For a typical time step of .01 seconds, a ten second study of the system requires 10^9 calculations.

Now consider the example of N atoms in a crystal lattice, where we approximate the quantum mechanical forces between neighboring atoms as anharmonic springs. Computing the net force on one atom requires 6 calculations because it is only affected by its nearest-neighbors. Computing the net force on all the atoms requires $6N$ calculations. For the same time step of .01 seconds, a ten second study requires only 6×10^6 calculations. The systems analyzed in the following chapters are computationally manageable because of the same nearest-neighbor simplification. The most complicated one requires about 10 minutes of computation time per simulation.

Confident that the mathematical and computational tools exist to thoroughly analyze vibrational problems, this thesis examines one such system, the clathrin cage. This complex protein provides a mechanism for released neurotransmitters to be collected back within the neuron for future use. Researchers have been able to isolate many of the key steps in the process, but there remain several mysteries. Understanding the protein's vibrational characteristics may provide useful insight into some of these details. Although biologically complex, the structure also engages our interest from a geometric, and consequently mathematical perspective. When considered as a system of point masses connected by springs, it turns out that the clathrin cage exhibits semi-rigid characteristics. The meaning of this will become clear in later chapters, and it conveniently turns out that these semi-rigid systems are ideal for the combined analytic/computational approach proposed.

Chapter 2

The Clathrin Cage

Clathrin is crucial for the transmission of materials into and out of cells throughout the body. From concentrating nutrients in the human placenta to the release of neurotransmitters in the brain, this protein is indispensable. Its role in material uptake (*receptor-mediated-endocytosis*) has only been fully grasped in the last decade, and several details of this complex process are still unknown. What follows is a summary of these steps, a description of why the process interests us, and a brief discussion of how vibration analysis may provide insight into one of the least understood steps.

2.1 The clathrin triskelion and endocytosis

To illustrate the steps involved in receptor-mediated-endocytosis, consider the part of the brain that is responsible for motor functions, the striatum (*basal ganglia*). Axons protruding from nerves in another region, the *substantia nigra*, extend into the striatum and release the neurotransmitter dopamine when triggered by an action potential. Although dopamine is actually produced in the body of the neurons in the substantia nigra, they are carried to the nerve terminal by a complex inter-cellular molecular transport system composed of vesicles. During dopamine release, these vesicles fuse with the cell membrane and disgorge their contents of neurotransmitters

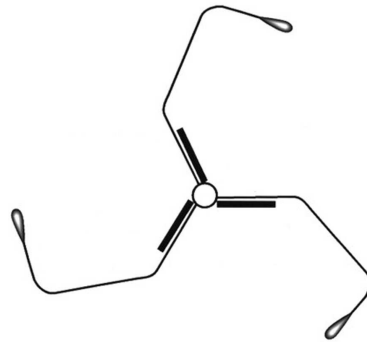


Figure 2.1: **The clathrin triskelion, shown here, has a diameter of approximately 500 Å [1].**

into the synapse. Dendrites from neurons on the other side of the synapse respond to the increase in dopamine concentration by firing a new action potential, which travels to the next neuron in the chain. Clathrin comes into play during the next step of the process, neurotransmitter reuptake.

Reuptake describes the process of recycling neurotransmitters. That is, after its release, dopamine is sequestered back into vesicles in the axon of the neuron that released it. In a healthy subject as much as 90 percent of the neurotransmitters are reused. When dopamine is released, the vesicle actually fuses with the membrane and is absorbed into it. As a result, reuptake requires the reformation of these vesicles from the cell's outer membrane. This is precisely clathrin's role. Attached to the inside of the cell membrane, clathrin protein assemblies consist of a pinwheel skeleton structure with three extended arms, a *triskelion* (See Figure 2.1). These triskelions are composed of one heavy chain and one light chain tightly linked to each other. The arms also have a certain degree of flexibility due to a joint which separates it into two pieces. The first piece is about 170 Å, the second piece is 220 Å in length [2]. Large numbers of clathrin proteins assemble on the inner cell membrane and their triskelion arms overlap to form polygonal lattices known as

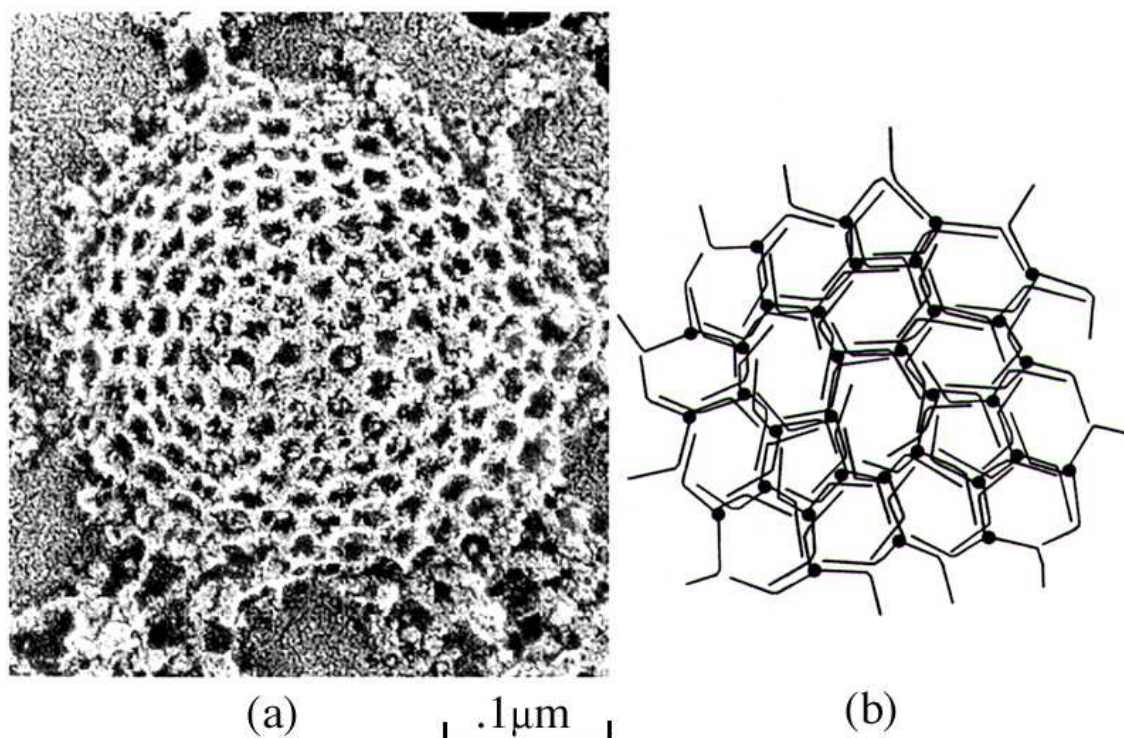


Figure 2.2: (a) **Electron micrograph of a clathrin-coated pit** ([9], p735). (b) **Diagram showing the interlocking triskelion structure of the pit** [2].

clathrin-coated pits (See Figure 2.2).

These lattices vary in composition and curvature since the triskelions readily bind in patterns that exhibit pentagonal, hexagonal, and a heptagonal symmetry. In describing the subsequent steps in reuptake, we follow Jin and Nossal's proposal that, "...triskelion energetics and interactions, influenced by the physiological environment, cause coated pits to invaginate and to bud off near the center of flat hexagonal lattices ([2] p1526)." This process begins when a small section of the lattice (on the order of ten polygonal faces) extends into the cell membrane as new triskelions are introduced. Because of their rigid geometry, the additional triskelions force the membrane to curve inward. Eventually, a "bud" is formed (See

Figure 2.3). Note that this bud can grow into a variety of different sizes and shapes depending on environmental conditions. One of the most common shapes, known as the *hexagonal barrel*, has been studied extensively. Smith et al used cryo-electron microscopy to generate a model of this structure with 21 Å resolution (See Figure 2.4). Throughout this process specific proteins, (dopamine in our case), bind to *assembly particles* which in turn bind to points on the clathrin lattice. Thus, only the proteins selected by these assembly particles will be concentrated within the recently formed clathrin bud.

Next, the clathrin bud must “pinch” off from the cell membrane. This is the least well understood step in the process. It is known that *dynamain*, a 900-amino-acid protein, and GTP (*guanosine 5-triphosphate*) are required for the successful separation of the vesicle [9] (See Figure 2.5). However, the exact mechanism involved remains unknown. Once the clathrin bud has separated from the lattice (becoming the so called “clathrin cage”), it depolymerizes via another protein interaction into triskelions, which can be reused in the formation of other vesicles. Meanwhile the newly formed, dopamine filled vesicle is ready to release its neurotransmitters again with the next action potential.

2.2 The relevance of dopamine reuptake

For an example which demonstrates why the details of vesicular formation are sought after, consider Parkinson’s disease. This neurodegenerative disease is characterized by dramatically decreased dopamine in the striatum. Although it is known that the number of neurons in the substantia nigra decrease as the disease progresses, and that this reduction in dopamine producing neurons influences dopamine concentration in the striatum, it is widely thought that the disease is more complicated. For instance, treatments exist which can artificially increase the quantity of dopamine in

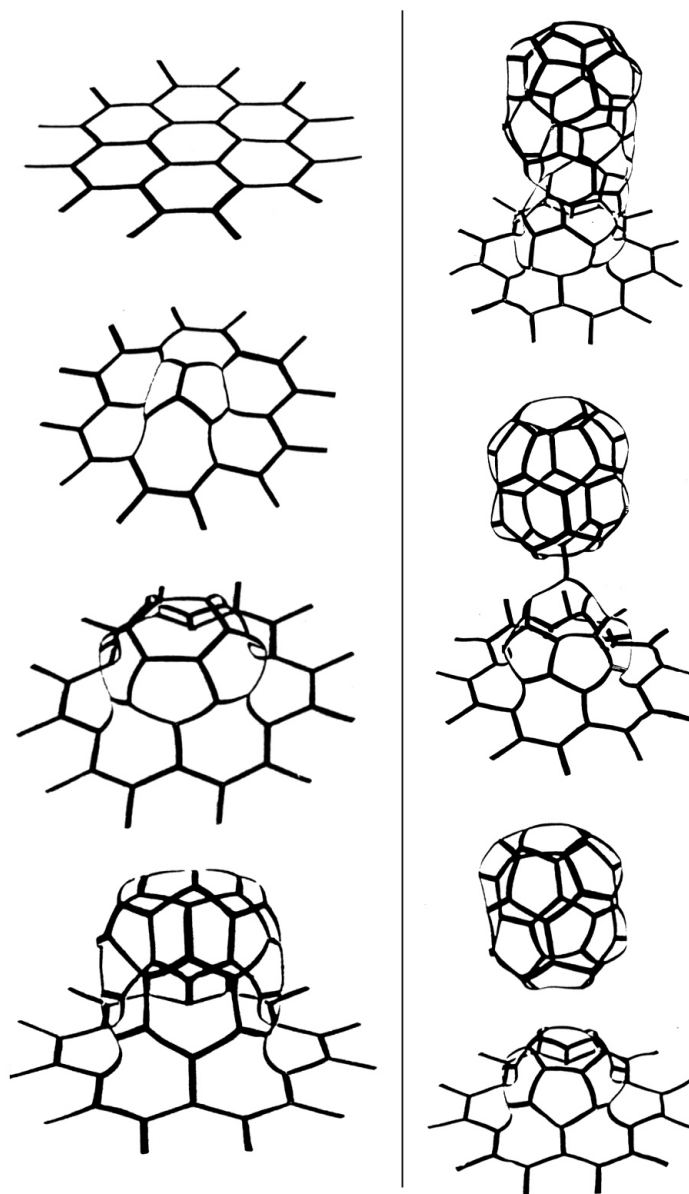


Figure 2.3: Diagram of steps involved in the formation of a free vesicle with *hexagonal barrel coat* symmetry [2].

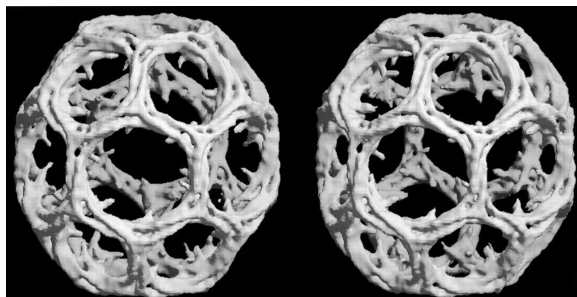


Figure 2.4: Stereo image representation of the clathrin cage. Obtained by cryo-electron microscopy and single-particle reconstruction [1].

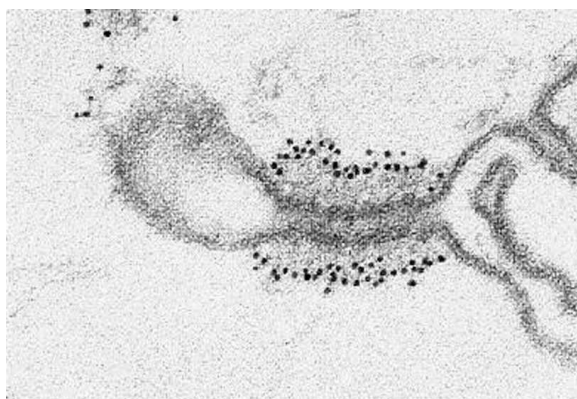


Figure 2.5: Electron micrograph of a clathrin-coated pit in the absence of GTP hydrolysis. Without GTP, the pit can form but cannot pinch off. ([9] p737).

the striatum. However, experience shows that these treatments are associated with a tolerance which increases with disease progression. Eventually the drug relieves Parkinsonian symptoms only at the peak of its concentration in the brain. This suggests that in addition to the slow death of substantia nigra cells, the brain's ability to efficiently recycle the remaining dopamine that it has (whether produced naturally or artificially) decreases with the progression of the disease. This drop in reuptake is indicative of a problem with vesicular formation.

2.3 Clathrin cage vibration

As mentioned earlier, clathrin bud separation requires GTP. This molecule, like its more powerful analog ATP, provides energy for various biological processes. With this functionality in mind, suppose the GTP energizes a vibrational mode of the clathrin cage such that its amplitude of oscillation steadily increases. The work that follows demonstrates the possibility that one such mode would involve the uniform expansion and contraction of all the vertices, a so-called breathing mode. Using this mode as an example, the increased amplitude of oscillation would eventually cause a sufficient contraction of the vertices to pinch the bud off from the remainder of the clathrin lattice. If the normal modes of clathrin can be identified, then their frequencies could be searched for experimentally with appropriate spectroscopy techniques.

Chapter 3

Analytic Approach to Coupled Oscillations

3.1 Introduction

The mathematics behind coupled oscillations unfolds in a concise and clear way when the interaction between particles is harmonic. In this case there exist *normal modes* of oscillation between which there is no coupling. This is to say that given appropriate initial conditions, each particle in the system will oscillate at a single frequency without exciting any other frequencies of oscillation. This fundamental quality of harmonic motion allows the use of linear algebra to greatly simplify the treatment of many-bodied systems.

The geometric shapes that will be considered in the following chapters are coupled in a complex way. Consider the triangle: if two points are connected by a spring, the force is determined by Hooke's law and therefore the distance between the two points. For systems which exist in multiple dimensions, the distance is found using the Pythagorean theorem. This means that wiggling some vertex in the x direction, for example, induces a force along the axis of the spring which depends on the square root of both the x and y coordinates. As much as we would like it to

be, this force is not linear.

However, If one chooses to consider only very small displacements from the system's equilibrium position, nonlinear components of the force can be ignored. We approximate the true spring force with a Taylor expansion and ignore higher order terms in the result, thus limiting the problem to small, harmonic vibrations. In this regime a variety of elegant linear algebra tools are available. The following section formulates this approach from the perspective of Lagrangian mechanics¹.

3.2 The general approach

Begin by describing our system of N particles (for a total of $3N$ degrees of freedom) by a set of generalized coordinates q_k where $k = 0, 1, \dots, N - 1$. Consider the equilibrium configuration where all the springs are relaxed at their rest length and each particle is in its equilibrium position defined as q_{k0} . Then because the particles are stationary, we have

$$q_k = q_{k0}, \quad \dot{q}_k = 0, \quad \ddot{q}_k = 0$$

We make the following claim regarding Lagrange's equations of motion at equilibrium:

$$\left. \frac{\partial L}{\partial q_k} \right|_0 - \left. \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_k} \right|_0 = \left. \frac{\partial L}{\partial q_k} \right|_0$$

The time derivative of $\left. \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_k} \right|_0$ will always include \dot{q}_k or \ddot{q}_k at equilibrium (denoted by the subscript 0) so it drops out of the formula. From the definition of the Lagrangian, $L = T - V$, we have

$$\left. \frac{\partial L}{\partial q_k} \right|_0 = \left. \frac{\partial T}{\partial q_k} \right|_0 - \left. \frac{\partial V}{\partial q_k} \right|_0 = 0. \quad (3.1)$$

¹Derivation taken from Marion et al (See [10] p466-471). Goldstein (See [12] p243-246) and Wilson et al (See [11] p11-33) provide alternative derivations.

The coordinate transformations that relate the cartesian coordinates to the generalized coordinates are assumed to be time independent, which allows us to invoke a definition of kinetic energy that depends only on the velocities of the generalized coordinates.

$$T = \frac{1}{2} \sum_{j,k} M_{jk} \dot{q}_j \dot{q}_k \quad (3.2)$$

So

$$\left. \frac{\partial T}{\partial q_k} \right|_0 = 0$$

and from 3.1

$$\left. \frac{\partial V}{\partial q_k} \right|_0 = 0 \quad (3.3)$$

This result simplifies the following series expansion. It is also useful to let the generalized coordinates q_k be measured from the equilibrium position by setting $q_{k0} = 0$. Next we Taylor expand the potential energy about this equilibrium position.

$$V = V_0 + \sum_{k=1}^n \left. \frac{\partial V}{\partial q_k} \right|_0 q_k + \frac{1}{2} \sum_{j,k} \left. \frac{\partial^2 V}{\partial q_j \partial q_k} \right|_0 q_j q_k + \dots \quad (3.4)$$

From 3.3 we see that the second term vanishes, and we're free to choose $V_0 = 0$. Ignoring higher order terms in the expansion isolates the harmonic vibrations. Therefore,

$$V = \frac{1}{2} \sum_{j,k} H_{jk} q_j q_k \quad (3.5)$$

where we have introduced the *Hessian* matrix defined as

$$H_{jk} = \left. \frac{\partial^2 V}{\partial q_j \partial q_k} \right|_0 \quad (3.6)$$

Defining our system in cartesian coordinates immediately diagonalizes the matrix \mathbb{M} in the kinetic energy,

$$T = \frac{1}{2} \sum_{j,k} M_{jk} \dot{q}_j \dot{q}_k = \frac{1}{2} \sum_r M_r \dot{q}_r^2$$

which is simply the sum of the kinetic energy associated with each particle.

Because T is a function of the velocities only and V is a function of the coordinates only, we can write the Lagrangian as

$$\frac{\partial L}{\partial q_k} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_k} = 0 \Rightarrow \frac{\partial V}{\partial q_k} - \frac{d}{dt} \frac{\partial T}{\partial \dot{q}_k} = 0$$

From 3.2 and 3.5 we compute the derivatives:

$$\frac{\partial V}{\partial q_k} = \sum_j H_{jk} q_j \quad \frac{\partial T}{\partial \dot{q}_k} = \sum_j M_{jk} \dot{q}_j$$

So the equations of motion are

$$\sum_j (A_{jk} q_j + M_{jk} \ddot{q}_j) = 0 \quad (3.7)$$

This is a set of n second-order linear homogenous differential equations with constant coefficients. We assume the solution for simple harmonic oscillation:

$$q_j(t) = a_j e^{i(\omega t - \phi)}$$

Substituting this into the equations of motion yields

$$\sum_j (H_{jk} - \omega^2 M_{jk}) a_j = 0 \quad (3.8)$$

This reduces the second-order linear differential equations down to linear algebraic equations. Non-trivial solutions exist only when the determinant is equal to zero:

$$|H_{jk} - \omega^2 M_{jk}| = 0 \quad (3.9)$$

In matrix form this is

$$\begin{vmatrix} H_{11} - \omega^2 M_{11} & H_{12} - \omega^2 M_{12} & H_{13} - \omega^2 M_{13} & \dots \\ H_{12} - \omega^2 M_{12} & H_{22} - \omega^2 M_{22} & H_{23} - \omega^2 M_{23} & \dots \\ H_{13} - \omega^2 M_{13} & H_{23} - \omega^2 M_{23} & H_{33} - \omega^2 M_{33} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{vmatrix} = 0 \quad (3.10)$$

Notice that both \mathbb{H} and \mathbb{M} are written symmetrically. This is because changing the order of the indices reverses the order of the partial derivatives, and in general the order of partial differentiation doesn't matter.

The equation shown in 3.10 is commonly referred to as the *characteristic equation*. Its solution consists of n roots which can be called ω_r^2 . These are *eigenvalues* of the characteristic equation. Choosing one of the roots, ω_r , and substituting it into the equations of motion (3.7) allows one to determine the vector a_j . These constants correspond to an *eigenvector* for that value of ω_r . Together, this data defines the general solution for one of the generalized coordinates, $q_j(t)$. Using the principle of superposition to add all of the resulting normal modes yields a linear combination of solutions which take the form

$$q_j(t) = \sum_r a_{jr} e^{i(\omega_r t - \phi_r)} \quad (3.11)$$

The real part of which is simply

$$q_j(t) = \sum_r a_{jr} \cos(\omega_r t - \phi_r) \quad (3.12)$$

3.3 Example: the triangle

One of the simplest, non-trivial systems that we can consider with this method is the triangle. Applying the results of the previous discussion involves determining the equilibrium configuration of the system, writing out the potential energy as a function of the generalized coordinates, computing the Hessian (\mathbb{H}) and its eigensystem², and then visualizing the motion. The equilibrium configuration is found using

²Rather than having to compute $|\mathbb{H} - \omega^2 \mathbb{M}|$, by setting $M = 1$ for all of our systems we simplify the analysis to $|\mathbb{H} - \omega^2 \mathbb{I}|$, which is equivalent to computing the eigenvalues of \mathbb{H} . By transforming to the so called *mass-weighted* coordinates, we could retain the simple dependence on \mathbb{H} while letting M vary [11].

geometry and the potential energy is found by inspection. All the other steps can be easily accomplished using *Mathematica*. The *Mathematica* code for the triangle appears in the appendix. Whenever possible I have used the same notation that appears in the text³. We adopt a notation in which the equilibrium configuration of the system is defined by \mathbf{R} , and the displacement of each vertex from equilibrium is given by \mathbf{r} . Thus, in cartesian coordinates the position of a vertex at a time t is given by

$$\mathbf{R}_i + \mathbf{r}_i(t) = (x_i(t), y_i(t), z_i(t)). \quad (3.13)$$

3.3.1 Determining the potential energy function

The energy stored in a spring is found by integrating Hooke's law. We define $\Delta r = r - d$, where r is the length of the spring and d is the equilibrium length.

$$V = - \int f_{\text{hooke}} dr = - \int -k\Delta r dr = \int kr dr - \int k d dr = \frac{1}{2}kr^2 - kdr + C$$

but

$$V(r = d) = 0 \implies C = \frac{1}{2}kd^2$$

which implies

$$V = \frac{1}{2}kr^2 - kdr + \frac{1}{2}kd^2 = \frac{1}{2}k(r - d)^2. \quad (3.14)$$

In order to implement this result, we must first define a function that calculates Δr in higher dimensional cases:

$$dr(\mathbf{r}_i, \mathbf{r}_j) = |(\mathbf{R}_i + \mathbf{r}_i) - (\mathbf{R}_j + \mathbf{r}_j)| - d. \quad (3.15)$$

If the vertices of the triangle are labeled as in figure 3.1, then the energy stored in the three springs is

$$V = \frac{1}{2}k (dr(r_0, r_1)^2 + dr(r_1, r_2)^2 + dr(r_2, r_0)^2). \quad (3.16)$$

³In *Mathematica*, elements of matrices and vectors are selected in a special way: $\mathbf{x}_i \rightarrow \mathbf{x}[[i]]$.

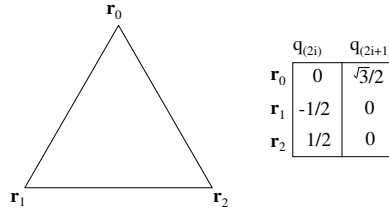


Figure 3.1: The coordinates for the triangle are defined in a counter-clockwise direction and start with the zero index: $q_0, q_1, q_2, \dots, q_{2N-1}$. This makes certain aspects of the *Mathematica* code cleaner than if we started the index at 1.

3.3.2 Computing \mathbb{H} and its eigensystem

We compute the Hessian by evaluating the appropriate partial derivatives and substituting the equilibrium positions of the coordinates.⁴

$$\mathbb{H} = \left. \frac{\partial^2 V}{\partial q_j \partial q_k} \right|_0 =$$

$$\begin{pmatrix} 0.5 & 0 & -0.25 & -0.433013 & -0.25 & 0.433013 \\ 0 & 1.5 & -0.433013 & -0.75 & 0.433013 & -0.75 \\ -0.25 & -0.433013 & 1.25 & 0.433013 & -1. & 0 \\ -0.433013 & -0.75 & 0.433013 & 0.75 & 0 & 0. \\ -0.25 & 0.433013 & -1. & 0 & 1.25 & -0.433013 \\ 0.433013 & -0.75 & 0 & 0. & -0.433013 & 0.75 \end{pmatrix} \quad (3.17)$$

Note that this matrix is symmetric as expected. *Mathematica* can compute the eigenvalues and eigenvectors. The square root of the eigenvalues, (i.e. the normal

⁴The coordinates' equilibrium positions are defined by \mathbf{R} . The zero subscript refers to the displacement vector (\mathbf{r}), which by definition, must go to zero.

frequencies), are shown here:

$$\omega = \begin{pmatrix} 1.73205 \\ 1.22474 \\ 1.22474 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (3.18)$$

Before visualizing these results, we immediately notice that three of the eigenvalues are zero. The three zero-modes are expected for the following reason: our solution does not restrict the motion of the particles in a translational or rotational way, and it requires exactly two translational and one rotational directions to define all the possible positions and orientations.

Because of the simplicity of this system, it is possible to show the eigenvectors, \mathbf{a}_r , as the rows of the following matrix:

$$\begin{pmatrix} 0 & 0.57735 & -0.5 & -0.288675 & 0.5 & -0.288675 \\ -0.00405143 & 0.577336 & 0.502013 & -0.285159 & -0.497962 & -0.292177 \\ 0.57735 & 0 & -0.288675 & -0.5 & -0.288675 & 0.5 \\ 0.0083665 & -0.372842 & 0.46961 & -0.639141 & 0.46961 & -0.106542 \\ -0.816497 & 0 & -0.204124 & -0.353553 & -0.204124 & 0.353553 \\ -0.580283 & -0.397816 & -0.231964 & -0.598918 & -0.231964 & -0.196714 \end{pmatrix} \quad (3.19)$$

The last three rows are eigenvectors of zero-frequency modes, implying that they are translational and rotational in nature. Any linear combination of horizontal translation, vertical translation, and rotation can be isolated by choosing the appropriate scaling factors, α_4 , α_5 , and α_6 . Although it is not shown here, these scaling factors can be easily found. Suppose one wants to find out which α represents translation

in the positive x direction. We know that a matrix of the form

$$\mathbf{a}_x = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} \quad (3.20)$$

describes this motion. In this case, each horizontal displacement coordinate, (q_0 , q_2 , and q_4), translates with zero frequency, and each vertical displacement coordinate, (q_1 , q_3 , and q_5) is fixed. Therefore, scaling the magnitude of this matrix translates the triangle to some new position on the x -axis. To show that the eigenvectors we have solved for include this possibility, we solve the following equation:

$$\alpha_4 \mathbf{a}_4 + \alpha_5 \mathbf{a}_5 + \alpha_6 \mathbf{a}_6 = \mathbf{a}_x \quad (3.21)$$

This is a set of 6 simultaneous equations with 3 unknowns, and although it is not shown here, the solution exists. The same argument holds for vertical translation and rotation, and three-dimensional cases can be treated similarly.

One subtle aspect of the rotation should be elaborated. The machinery we have developed looks at infinitesimal disturbances from equilibrium. Rotational motion doesn't disturb the potential for small or large disturbances from equilibrium, but our techniques do not make that distinction. Consequently, scaling a rotational mode accurately reflects small rotations, but distorts larger ones. Zero-modes are discussed in more detail in Appendix A.

3.3.3 The general solution

The normal frequency, w_r , and each element of the corresponding eigenvector, $a_{r,j}$, define the angular frequency and the amplitude of the simple harmonic motion executed by q_j . In general, the motion is a linear combination of all possibilities.

Although the true solution allows each mode to oscillate with its own, specific phase, this analysis omits that possibility. We do include the necessary scaling factor, α_r , that determines the magnitude of the r^{th} normal mode. From this and equation 3.12 we can write:

$$q_j(t) = \sum_r \alpha_r a_{jr} \cos(\omega_r t) \quad (3.22)$$

The motivation for choosing our index j to count from zero becomes apparent when putting these results back in vector form:

$$\mathbf{r}_j(t) \longrightarrow (q_{(2j)}(t), q_{(2j+1)}(t)) \quad (3.23)$$

The solution is now in a form that *Mathematica* can animate. Figure 3.2 shows a series of snapshots taken at regular intervals after animating the vibrational modes of the triangle. Figure 3.3 shows how the triangle moves when a zero-frequency mode is selected, and instead of incrementing the time, t , we increment the amplitude of the motion, α_r .

Vibrational Modes of the Triangle

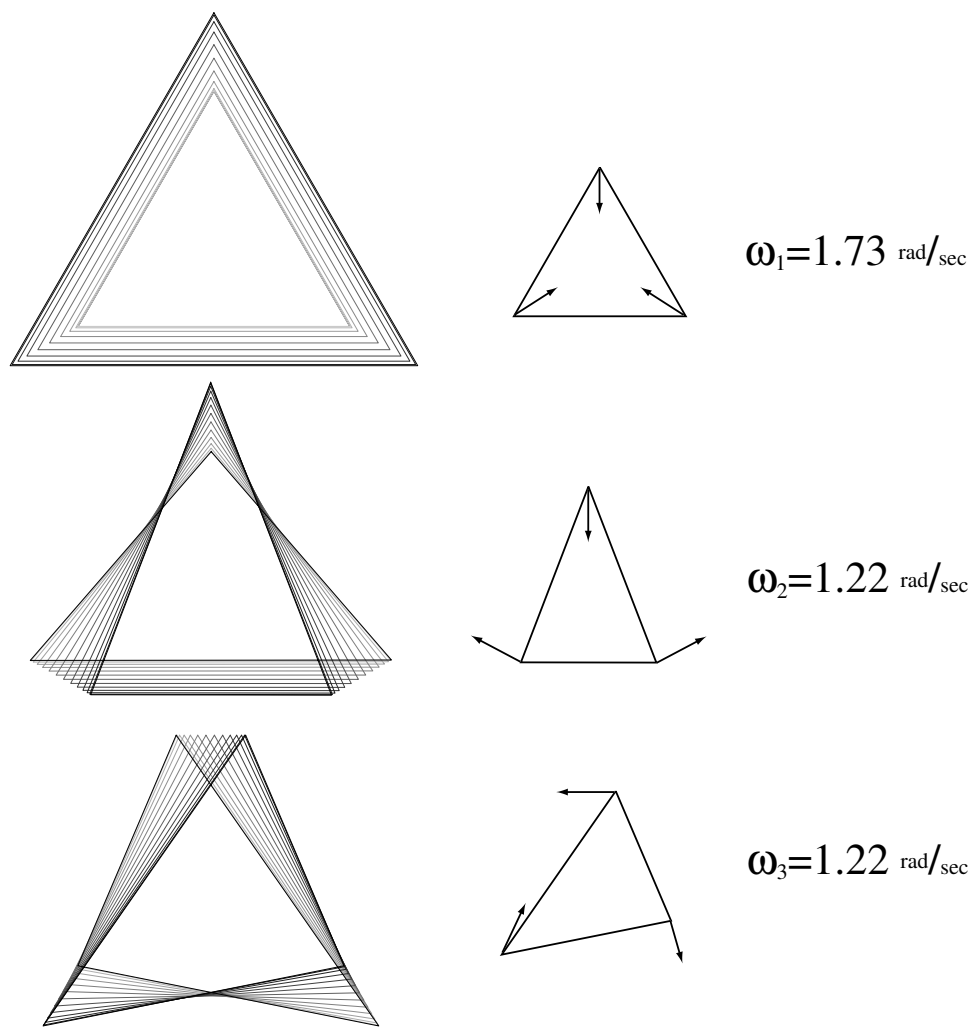


Figure 3.2: The normal modes of a triangle with the vertices connected by springs. Snapshots are taken at .2 second intervals and fade into black at the final time $\frac{T}{2} = \frac{2\pi}{\omega_r}$.

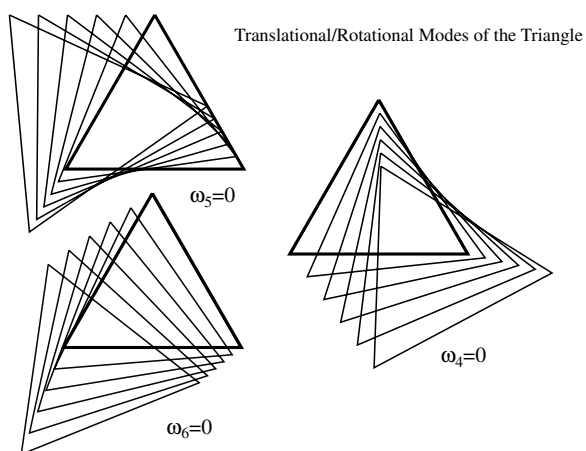


Figure 3.3: Three of the triangle's normal mode are zero. These correspond to translational and rotational motion in which all of the springs remain at equilibrium. By increasing the amplitude of α we can see how the associated eigenvectors \mathbf{a}_r affect the triangle.

Chapter 4

Numerical Approach to Coupled Oscillations

The analytical method introduced in the previous chapter has been successfully applied in a wide variety of applications. However, it is possible to understand the motion of mass-spring systems more completely by pairing this mathematically rigorous analytical approach with a computational one. By incrementing the equations of motion forward in time, the trajectory of a given system can be calculated knowing only the physical laws that govern it. This “brute force” method doesn’t require the same mathematical restrictions occasionally needed in the analogous mathematical treatment, and it can be easily modified to treat a wide variety of systems. Although the numerical method does not illuminate fundamental behavior the same way that algebraic solutions often do, the results can be probed using a number of techniques and methods commonly found in traditional experimental physics.

For mass-spring systems, a physical analogy to the computational approach would involve going into the laboratory and connecting a bunch of masses with springs in the appropriate way, then plucking one of them and listening to the object vibrate. Given the personal computing power available today, it is more practical to construct and manipulate these objects within a digital framework.

To accomplish this, it is necessary to introduce a general technique for transforming equations of motion, like Newton's 2nd law, into a form appropriate for numerical integration algorithms¹.

$$\frac{d\mathbf{y}}{dt}(t) = \mathbf{f}(t, \mathbf{y}) \quad (4.1)$$

where \mathbf{y} and \mathbf{f} are N -dimensional vectors.

$$\mathbf{y} = \begin{pmatrix} y_1(t) \\ y_2(t) \\ y_3(t) \\ \vdots \\ y_N(t) \end{pmatrix}, \quad \mathbf{f} = \begin{pmatrix} f_1(t, \mathbf{y}) \\ f_2(t, \mathbf{y}) \\ f_3(t, \mathbf{y}) \\ \vdots \\ f_N(t, \mathbf{y}) \end{pmatrix} \quad (4.2)$$

Solving equation 4.1 amounts to solving the following N simultaneous first-order ODE's:

$$\begin{aligned} \frac{dy_1}{dt}(t) &= f_1(t, \mathbf{y}), \\ \frac{dy_2}{dt}(t) &= f_2(t, \mathbf{y}), \\ &\vdots \\ \frac{dy_N}{dt}(t) &= f_N(t, \mathbf{y}). \end{aligned}$$

When modeling physical systems, second-order differential equations are often encountered. Newton's second law, for example, takes the form

$$\frac{d^2x}{dt} = \frac{1}{m}F(t, \frac{dx}{dt}, x). \quad (4.3)$$

This second-order ODE can be rewritten as two simultaneous first-order ODE's by defining a new variable, v , which is the first derivative of x :

$$\frac{dx}{dt}(t) = v(t) \quad (4.4)$$

¹The following approach is taken from Landau et al (See [7] p122-128). Press et al (See [8] p707-752) provide a more comprehensive discussion including a variety of C implementations.

$$\frac{dv}{dt}(t) = \frac{1}{m}F(t, v, x) \quad (4.5)$$

In the form above, Newton's equation of motion can be crunched by a variety of integration algorithms. Note that, if our system contains N particles instead of just one, equations 4.4 and 4.5 would become a set of $2N$ first-order equations. The force function on the right hand side would then become a function of all the possible interactions: $F(t, v, x) \rightarrow F(t, v_1, v_2, \dots, v_N, x_1, x_2, \dots, x_N)$. Expressing this in terms of the N element vectors \mathbf{x} and \mathbf{v} , we have

$$\frac{d\mathbf{x}}{dt}(t) = \mathbf{v}(t) \quad (4.6)$$

$$\frac{d\mathbf{v}}{dt}(t) = \frac{1}{m}\mathbf{F}(t, \mathbf{x}, \mathbf{v}) \quad (4.7)$$

The physical systems considered in the following chapters lie in three dimensions, so the actual number of equations hidden in the sleek vector notation above increases from $2N$ for a one-dimensional system to $3 \times 2N = 6N$ for a three-dimensional one. Once the differential equations are organized in this manner, the application of various integration algorithms follows smoothly.

The simplest and most intuitive of these is Euler's algorithm. It evaluates the force function $\mathbf{f}(t, \mathbf{y})$ at the beginning of some time interval, t_0 , for which $\mathbf{y}(t_0)$ is known. Then it makes the crude assumption that this value, $\mathbf{f}(t_0, \mathbf{y}(t_0))$, remains constant over the small time interval h . Thus the value of \mathbf{y} at some future time is simply $\mathbf{y}(t_0 + h) \approx \mathbf{y}(t_0) + h\mathbf{f}(t_0, \mathbf{y}(t_0))$ (See figure 4.1). Because the implementation of integration algorithms involves many iterations, it is common to write them in terms of n , the number of time steps. Written this way Euler's algorithm is

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{f}(t_n, \mathbf{y}_n) + O(h^2) \quad (4.8)$$

where it is important to note that the subscript refers to the number of time steps, not specific components of the vectors. The large correction factor, $O(h^2)$, is one reason why this method is rarely recommended.

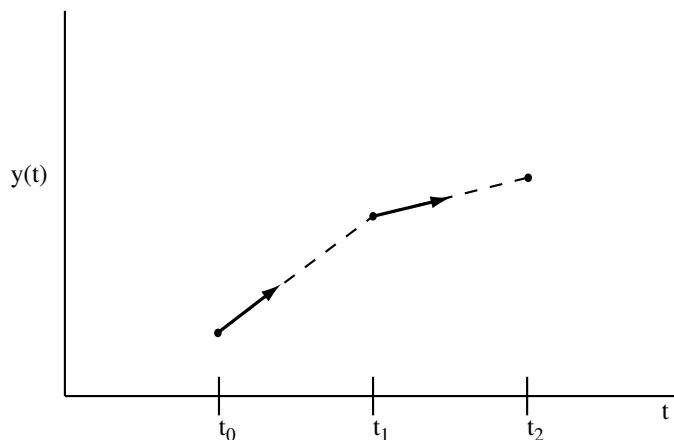


Figure 4.1: **Euler's method.** The derivative at the starting point of each interval is extrapolated to find the next function value. This method has first-order accuracy [8].

The second-order Runge-Kutta, or the midpoint method, achieves greater accuracy by using a trial point in the middle of the interval to estimate the value of \mathbf{f} (See figure 4.2). The algorithm is

$$\begin{aligned}
 \mathbf{k}_1 &= h\mathbf{f}(t_n, \mathbf{y}_n) \\
 \mathbf{k}_2 &= h\mathbf{f}\left(t_n + \frac{1}{2}h, \mathbf{y}_n + \frac{1}{2}\mathbf{k}_1\right) \\
 \mathbf{y}_{n+1} &= \mathbf{y}_n + \mathbf{k}_2 + O(h^3).
 \end{aligned} \tag{4.9}$$

The fourth-order Runge-Kutta algorithm (abbreviated as RK_4) samples \mathbf{f} at four trial points within the interval (See figure 4.3). By cleverly choosing and combining these samples, it achieves fourth-order accuracy (that is the correction factor is

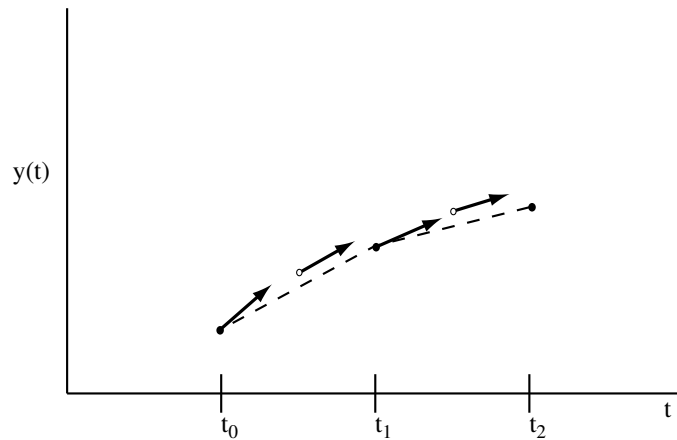


Figure 4.2: Midpoint method. Second-order accuracy is obtained by using the initial derivative at each step to find a point halfway across the interval, then using the midpoint derivative across the full width of the interval. Filled dots represent final function values. Open dots represent function values that are discarded once their derivatives have been calculated and used [8].

$O(h^5)$). The algorithm is

$$\begin{aligned}
 \mathbf{k}_1 &= hf(t_n, \mathbf{y}_n) \\
 \mathbf{k}_2 &= hf\left(t_n + \frac{1}{2}h, \mathbf{y}_n + \frac{1}{2}\mathbf{k}_1\right) \\
 \mathbf{k}_3 &= hf\left(t_n + \frac{1}{2}, \mathbf{y}_n + \frac{1}{2}\mathbf{k}_2\right) \\
 \mathbf{k}_4 &= hf(t_n + h, \mathbf{y}_n + \mathbf{k}_3) \\
 \mathbf{y}_{n+1} &= \mathbf{y}_n + \frac{\mathbf{k}_1}{6} + \frac{\mathbf{k}_2}{3} + \frac{\mathbf{k}_3}{3} + \frac{\mathbf{k}_4}{6} + O(h^5).
 \end{aligned} \tag{4.10}$$

Even though this algorithm requires four times as many calculations as Euler's method and twice as many calculations as the midpoint method, it will be used to address our problem of mass-spring systems. The main motivation for this choice is simple: the necessary processing power is available, so there is no need to exchange accuracy for speed. This is partly due to the computationally favorable nature of

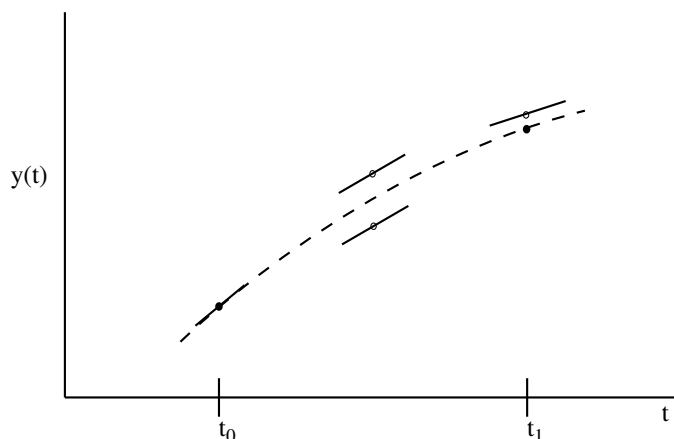


Figure 4.3: **Fourth-order Runge-Kutta method.** The derivative is evaluated four times: once at the initial point, twice at trial midpoints, and once at a trial endpoint. From these derivatives the final function value (the filled dot) is calculated [8].

problems involving nearest-neighbor interactions, and also because the most complicated system we consider, the clathrin cage, contains 36 vertices. Thanks to the rapidly growing power of personal computers, the motion of these 36 coupled point-masses can be integrated using the fourth-order Runge-Kutta method in a reasonable amount of time.

4.1 Mass-spring implementation

Applying the techniques of the previous section to systems of coupled point-masses is a straightforward exercise. In practice, it can be broken down into the steps shown in figure 4.4. After defining the variables, the initial conditions, and the force function, the $RK4$ algorithm computes the trajectory of each of the masses in time. The result can be analyzed in a variety of different ways. In this discussion we focus on visualization and Fourier transform techniques.

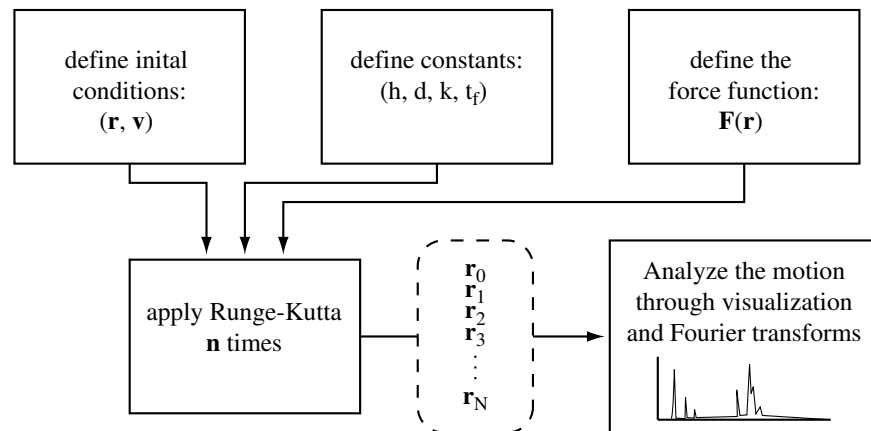


Figure 4.4: **Schematic representation of the mass-spring simulation. The actual Mathematica code appears in the Appendix.**

4.1.1 Defining constants and initial conditions

The notation used to implement the $RK4$ algorithm appears in the frontmatter and is repeated here:

- m \longrightarrow mass of each vertex
- k \longrightarrow force constant of each spring
- d \longrightarrow equilibrium length of each spring
- h \longrightarrow integration time step
- t_f \longrightarrow target integration time
- n \longrightarrow the number of steps made in the integration ($n = \frac{t_f}{h}$)
- N \longrightarrow the number of vertices in the system
- \mathbf{r} \longrightarrow vector of vertex positions (each entry takes the form (x, y, z))
- \mathbf{v} \longrightarrow vector of vertex velocities (each entry takes the form (v_x, v_y, v_z))

We can freely select any appropriate value for k , d , t_f , and m , but because this exploration is based on a comparison between a variety of systems and approaches it is in our best interest to keep them constant whenever possible. No effort will be made to select values which correlate with real physical systems, although these

types of considerations are appropriate for future work.

For simple objects like the triangle, the cube, and various polyhedra, the equilibrium position of each of the vertices, \mathbf{R} , can be computed using geometry. With more complicated objects, the obvious example being the clathrin cage, \mathbf{R} is not so easily identified. We can capitalize on the fact that we are interested in equilibrium configurations, however, by making a guess at the initial positions of the vertices and letting the system relax to its lowest energy state. This is the method employed to find clathrin's vertices in chapter 7. Meanwhile, the velocity vector, \mathbf{v} , is zero at equilibrium.

4.1.2 Determining the force function

As discussed earlier, physical systems are governed by Newton's 2nd law which we can write in terms of two first-order differential equations:²

$$\frac{d\mathbf{r}}{dt}(t) = \mathbf{v}(t) \quad (4.11)$$

$$\frac{d\mathbf{v}}{dt}(t) = \frac{1}{m}\mathbf{F}(t, \mathbf{r}, \mathbf{v}) = \frac{1}{m}\mathbf{F}(\mathbf{r}) \quad (4.12)$$

From these equations it is clear that the force function of \mathbf{r} is simply the value of \mathbf{v} at that time. The force function of \mathbf{v} will be given by Hooke's law which depends only on \mathbf{r} . In order to find $\mathbf{F}(\mathbf{r})$ we define the tension in the spring between the vertices r_i and r_j as

$$T(\mathbf{r}_i, \mathbf{r}_j) = -k(|\mathbf{r}_i - \mathbf{r}_j| - d)\frac{\mathbf{r}_i - \mathbf{r}_j}{|\mathbf{r}_i - \mathbf{r}_j|} \quad (4.13)$$

One can verify that this carries the correct sign by assuming the two vectors are stretched beyond the equilibrium distance. If so, the first half of the equation,

²Here the vector of one-dimensional components, \mathbf{x} , has been replaced by a vector of three-dimensional components, \mathbf{r} . This notation is slightly ambiguous because from now on \mathbf{v} will also be treated as a vector of three-dimensional components, even though its label remains unchanged.

$-k(|\mathbf{r}_i - \mathbf{r}_j| - d)$ carries a negative value. The second term, $\frac{\mathbf{r}_i - \mathbf{r}_j}{|\mathbf{r}_i - \mathbf{r}_j|}$, is a normal vector pointing from \mathbf{r}_j to \mathbf{r}_i . Thus, the vertex at \mathbf{r}_i will be pulled toward the vertex at \mathbf{r}_j , as expected.

The total force on any one vertex is simply the sum of the tensions acting on it

$$F(\mathbf{r}_i) = \sum_j T(\mathbf{r}_i, \mathbf{r}_j) \quad (4.14)$$

where j ranges over any vertices that are connected to \mathbf{r}_i . The force function then, takes the vector of positions, \mathbf{r} , applies F to each entry, and spits out a vector which represents the total force on each vertex:

$$\mathbf{F}(\mathbf{r}) = \begin{pmatrix} F(\mathbf{r}_1) = \sum_j T(\mathbf{r}_1, \mathbf{r}_j) \\ F(\mathbf{r}_2) = \sum_j T(\mathbf{r}_2, \mathbf{r}_j) \\ \vdots \\ F(\mathbf{r}_N) = \sum_j T(\mathbf{r}_N, \mathbf{r}_j) \end{pmatrix} \quad (4.15)$$

The previous two functions, $T(\mathbf{r}_i, \mathbf{r}_j)$ and $\mathbf{F}(\mathbf{r})$ appear in the included mathematica code as `$\mathbf{T}[\mathbf{i}][\mathbf{j}]$` and `$\mathbf{F}[\mathbf{r}]$` .

4.1.3 Implementing *RK4*

To integrate Newton's second law and obtain the positions of the masses at some future time, we enter the initial positions, the constants, and the force function into the *RK4* algorithm. Because we actually have 2 simultaneous sets of N differential equations, one set that defines $\frac{d\mathbf{r}}{dt}$ and one set that defines $\frac{d\mathbf{v}}{dt}$, it is necessary to

rewrite equation 4.10 in the following way:³

$$\begin{aligned}
\mathbf{k}_1 &= h \frac{1}{m} \mathbf{F}(\mathbf{r}_n) \\
\mathbf{j}_1 &= h \mathbf{v} \\
\mathbf{k}_2 &= h \frac{1}{m} \mathbf{F}(\mathbf{r}_n + \frac{1}{2} \mathbf{j}_1) \\
\mathbf{j}_2 &= h(\mathbf{v} + \frac{1}{2} \mathbf{k}_1) \\
\mathbf{k}_3 &= h \frac{1}{m} \mathbf{F}(\mathbf{r}_n + \frac{1}{2} \mathbf{j}_2) \\
\mathbf{j}_3 &= h(\mathbf{v} + \frac{1}{2} \mathbf{k}_2) \\
\mathbf{k}_4 &= h \frac{1}{m} \mathbf{F}(\mathbf{r}_n + \mathbf{j}_3) \\
\mathbf{j}_4 &= h(\mathbf{v} + \mathbf{k}_3) \\
\mathbf{v}_{n+1} &= \mathbf{v}_n + \frac{\mathbf{k}_1}{6} + \frac{\mathbf{k}_2}{3} + \frac{\mathbf{k}_3}{3} + \frac{\mathbf{k}_4}{6} \\
\mathbf{r}_{n+1} &= \mathbf{r}_n + \frac{\mathbf{j}_1}{6} + \frac{\mathbf{j}_2}{3} + \frac{\mathbf{j}_3}{3} + \frac{\mathbf{j}_4}{6}
\end{aligned} \tag{4.16}$$

As confusing as this set of equations first appears, it is easy to assign physical meaning to each of the terms. For example, \mathbf{j}_1 is just the change in the position vector over the interval h , assuming \mathbf{v} holds its initial value. \mathbf{j}_2 is the change in the position vector over the interval assuming \mathbf{v} holds its estimated midpoint value. The other \mathbf{j} 's and \mathbf{k} 's can be interpreted in a similar fashion. It is a testament to the RK_4 algorithm that appropriately combining these terms obtains such high accuracy.

4.2 Example: the triangle

In order to illustrate several methods of interpreting the results of this algorithm and to assess its accuracy, we consider a system whose solution was obtained in

³The subscripts on \mathbf{v} and \mathbf{r} refer to their values at different intervals of h , not specific components of the vectors.

the previous chapter: the triangle. First the specifics of setting up \mathbf{F} are discussed. Then a method of testing the algorithm's accuracy will be introduced. Finally, Fourier analysis is used to compare the classical and the computational approach.

4.2.1 Determining $\mathbf{F}(\mathbf{r})$ for the triangle

Determining the force function of each vertex in the triangle is easily done using the function for tension, $T(\mathbf{r}_i, \mathbf{r}_j)$.⁴ If the vertices are labeled as before (See figure 3.1), then

$$\mathbf{F}(\mathbf{r}) = \begin{pmatrix} T(\mathbf{r}_0, \mathbf{r}_1) + T(\mathbf{r}_0, \mathbf{r}_2) \\ T(\mathbf{r}_1, \mathbf{r}_0) + T(\mathbf{r}_1, \mathbf{r}_2) \\ T(\mathbf{r}_2, \mathbf{r}_0) + T(\mathbf{r}_2, \mathbf{r}_1) \end{pmatrix} \quad (4.17)$$

With the force function known, we execute the *RK4* algorithm to obtain the solution. As the simulation runs, a series of values for \mathbf{r} and \mathbf{v} are generated corresponding to the changing positions and velocities, respectively, of the triangle's vertices in time. It is useful to concatenate these values of \mathbf{r} together in a list called **rdata** for future analysis. Note that **rdata** is a nested list: each value of \mathbf{r} within **rdata** contains a set of (x, y) coordinates defining the position of each vertex. In order to pick out specific components of this data, we introduce the relation

$$\mathbf{rdata}(i) = \mathbf{r}(t = h \times i) \quad (4.18)$$

4.2.2 Checking energy conservation

The algorithm's accuracy can be gauged by observing how the total energy of the system changes in time. Because it is a closed system, we know that the sum of the potential and the kinetic energy must remain constant. These two quantities can be calculated from the model by noting that the potential can be written in terms of a

⁴You may have noticed that this equation is over-characterized; we haven't taken advantage of the fact that the tension on \mathbf{r}_i due to \mathbf{r}_j is just the opposite of the tension on \mathbf{r}_j due to \mathbf{r}_i . It would be quite easy to define a function which deals with this in a more elegant way, but for the level of complexity considered here there's no need. We have the CPU cycles to spare.

sum of spring tensions, and the kinetic energy can be calculated from the velocity in the usual way. The energy stored in a single spring can be written:

$$V = \frac{1}{2}k\Delta r^2 = \frac{1}{2k}T^2 \quad (4.19)$$

So the potential energy at time t is

$$V(t) = \frac{1}{2k} \sum_j T_j(t)^2 \quad (4.20)$$

where j varies over each of the springs. The kinetic energy is

$$K(t) = \frac{1}{2}m \sum_k |\mathbf{v}_k(t)|^2 \quad (4.21)$$

where k varies over each of the vertices. After running the simulation and monitoring this quantity, $E = V(t) + K(t)$, we notice that the total energy is slowly draining (See figure 4.5). To quantize how “bad” this is for our model, we can take advantage of the one-way change in E to easily compute the fractional energy loss:

$$\frac{\Delta E}{E} = \frac{E(0) - E(t_f)}{E(0)}$$

For typical values of h and n , about 0.3% of the initial energy is lost (See figure 4.5). It is difficult to predict how this will affect the computational model, but small vibration comparisons with the analytic solution agree to high precision, which suggests that the energy loss is acceptable.

4.2.3 Comparing the results

In order to compare the computational and analytical results, we use Fourier analysis to obtain the frequency spectrum of the computational result⁵. *Mathematica* includes a built in Fourier transform function called *Fourier* which is ideal for our

⁵A detailed discussion of discrete Fourier transforms can be found in Landau et al (See [7] p157-161).

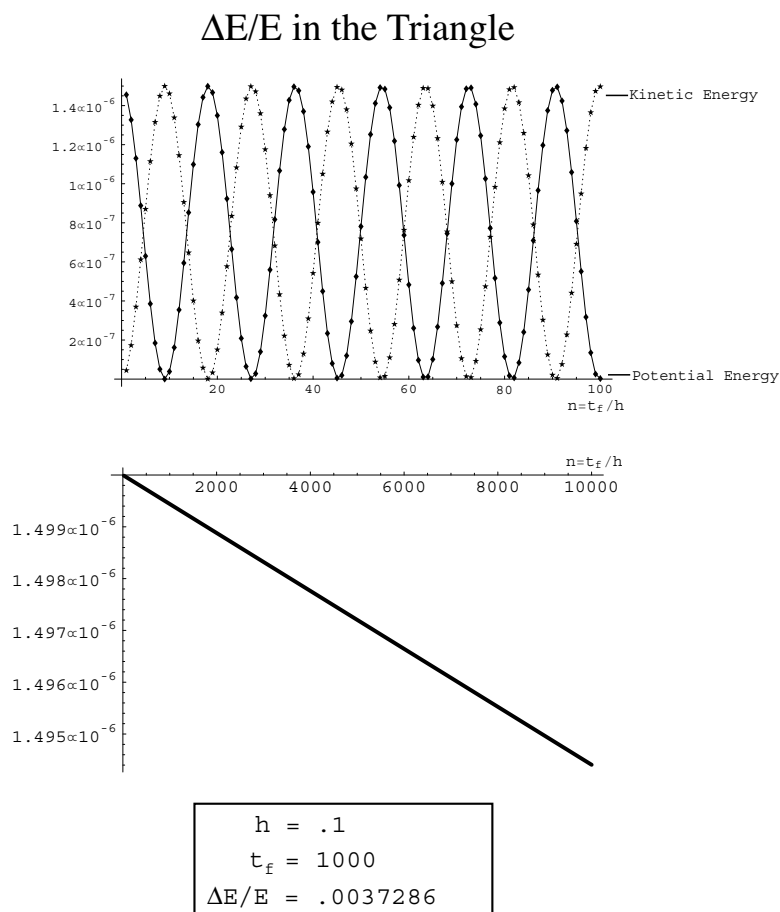


Figure 4.5: When simulating the vibrating triangle with the fourth-order Runge-Kutta algorithm, we see that conservation of energy is violated. Looking at the fractional energy lost, $\Delta E/E$, for typical values of the variables allows us to gauge the accuracy of the algorithm.

purposes⁶. Before applying this function, we force the data to be periodic within the interval n by multiplying it by an appropriate window function. This prevents the discrete Fourier transform from trying to process data which is not truly periodic (except in the unlikely event that $\mathbf{r}(t_f)$ and $\mathbf{v}(t_f)$ exactly match the initial conditions). The square of a sine function is used to obtain the windowed data:

$$\mathbf{wrdata}(i) = \mathbf{rdata}(i) \times \sin\left(\frac{\pi}{n}i\right)^2 \quad (4.22)$$

where i ranges from 0 to n . In order to get as much information out of the data as possible, we apply *Fourier* to the motion of each coordinate q_i . The results of these $2N$ discrete Fourier transforms (or $3N$ for a three-dimensional system) are then summed together to obtain an accurate understanding of the triangle's frequency spectrum. The discrete Fourier transform of n data points produces n new transformed data points. The magnitude of the i^{th} element of the transformed data determines the amount of the frequency ω_i present in the original data, where

$$\omega_i = i \frac{2\pi}{nh} \quad (4.23)$$

We use this equation to pair the results of *Fourier* with matching angular frequencies. It is important to notice that this equation implies that our frequency resolution will have a width of

$$\omega_{i+1} - \omega_i = \frac{2\pi}{nh} = \frac{2\pi}{t_f}. \quad (4.24)$$

Therefore, increasing h and t_f simultaneously will *not* increase frequency resolution. We must pick a suitable value of h and then run the simulation long enough to obtain the necessary resolution. In the following figures, $t_f = 1000$, so we expect a frequency resolution of $\sim .005$ radians/second.

In order to use this method for comparison, we sample the analytic solution at intervals of h and use its initial position, $q_0(0), q_1(0), \dots, q_5(0)$, as the starting point

⁶This function actually calls an FFT algorithm.

of our simulation⁷. If the Runge-Kutta algorithm works perfectly, the two Fourier transforms will be identical for small displacements from equilibrium. Recall that α_r scales the magnitude of the vibrational mode ω_r . Figure 4.6 shows the Fourier transforms for the first two normal modes, and then a linear combination of them. Because our analytical solution is for infinitesimal displacements from equilibrium, the small values of α selected are suitable. We can also demonstrate that arbitrary motion is truly a linear combination of motion at the normal frequencies. In figure 4.7 the initial value of \mathbf{r} is randomly selected, but the resulting motion is still a composite of two frequencies. Comforted with the knowledge that both solutions agree and behave as expected, the next several chapters consider more interesting structures including the tetrahedron, cube, and clathrin cage.

⁷Because we picked the general solution to be a sum of cosine terms, when $t = 0$ the triangle will be maximally stretched and will have no velocity: $\dot{q}_i(0) = \sum(\text{constant terms}) \times \sin(0) = 0$. This means we can set the initial velocity of the computational solution to be zero in all cases.

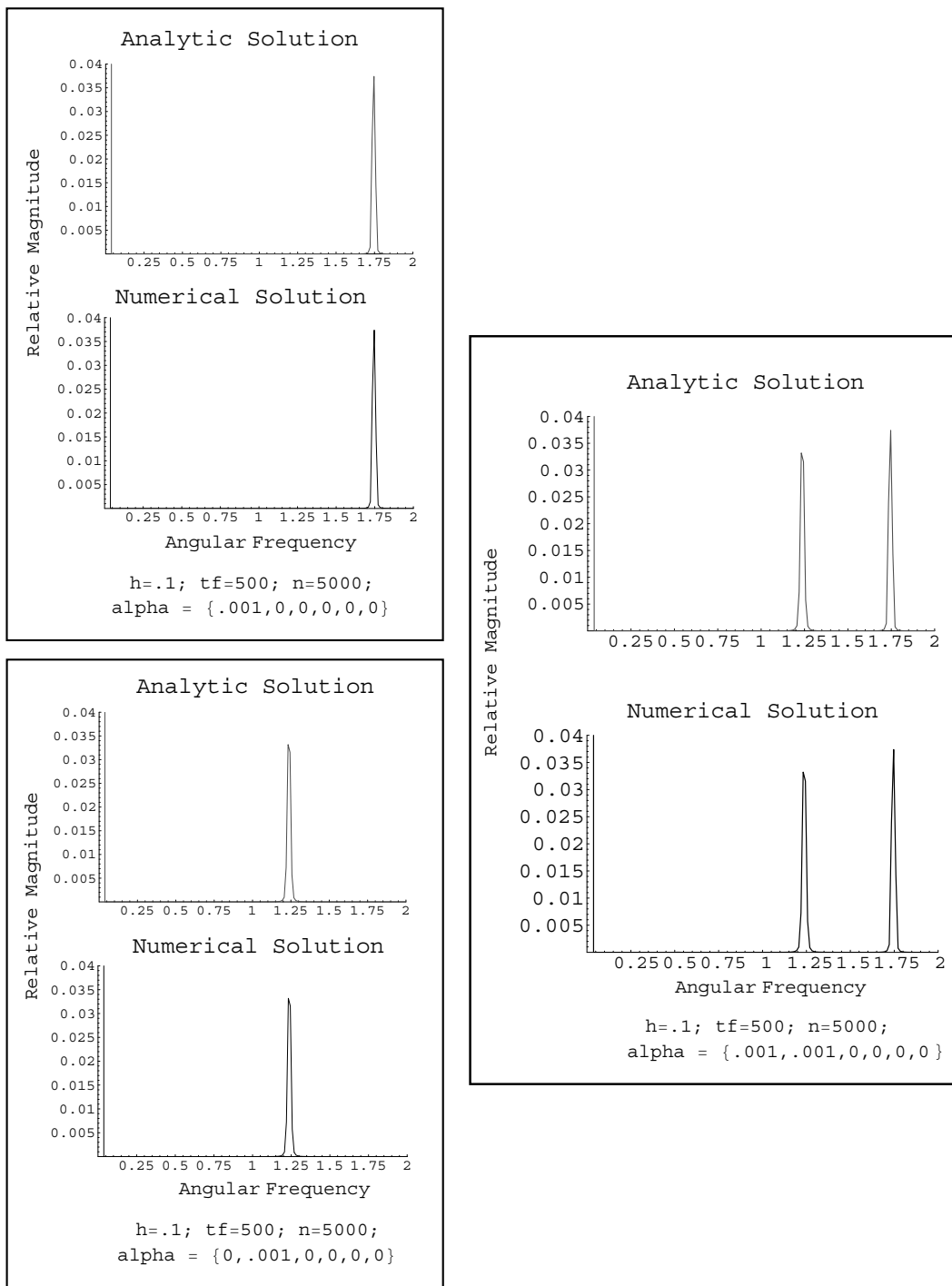


Figure 4.6: Comparison of the analytical and computational solutions to the vibrating triangle. Two of the three vibrational modes are individually stimulated by stretching the triangle in the appropriate way. Then both modes are stimulated simultaneously. Extremely small initial displacements are used to make sure the system behaves harmonically

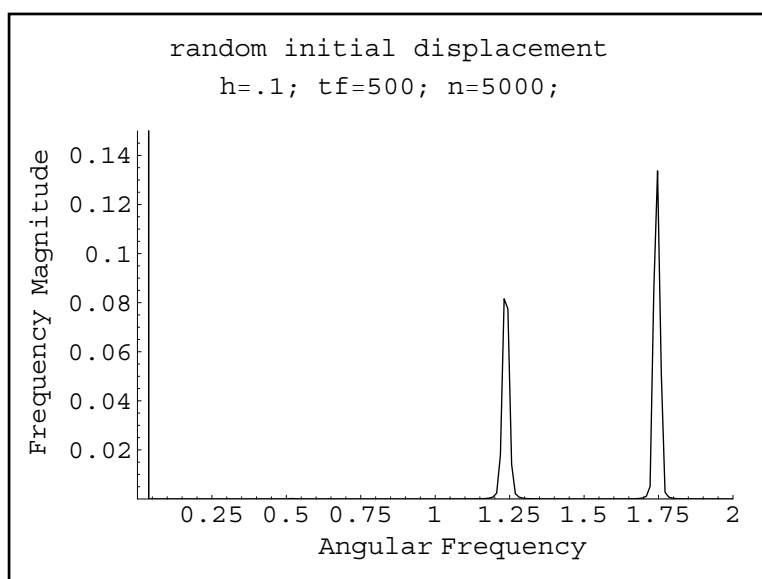


Figure 4.7: Small displacements from the equilibrium position of r are randomly selected. The numerical solution shows that the resulting motion is a linear combination of oscillations occurring at the two normal frequencies.

Chapter 5

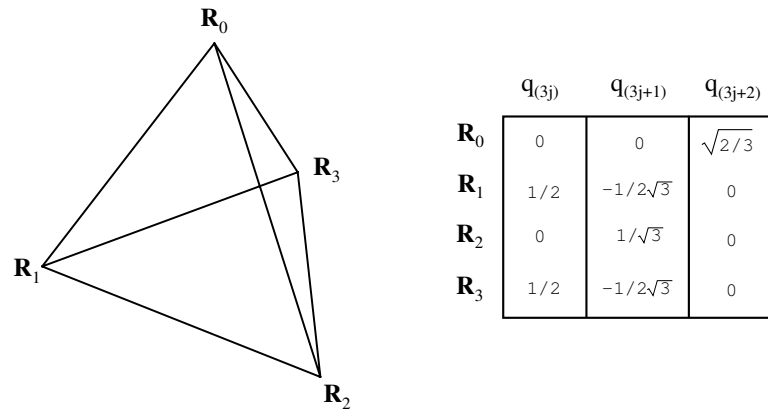
The Vibrating Tetrahedron

Normal frequencies in the tetrahedron can be found using either of the two techniques discussed previously. In order to understand the motion more completely, we choose to apply both of them. The analytic solution provides us with the general solution for small vibrations, but by examining the Fourier transform of the computational solution we can explore some of its nonlinear qualities.

5.1 The analytic solution

Unlike the triangle, the tetrahedron is three-dimensional. This means that we have $3N = 12$ degrees of freedom. We expect six of these to correspond with rotation and translation (the tetrahedron can rotate around two more axes and translate in one more direction than the triangle lying in a plane). It is easy to write out the potential energy since every vertex is connected to every other vertex. There are six terms in the potential energy, one for each spring. Figure 5.1 orients and labels our tetrahedron.

$$V = \frac{1}{2}k (dr(r_0, r_1)^2 + dr(r_0, r_2)^2 + dr(r_0, r_3)^2 + dr(r_1, r_2)^2 + dr(r_1, r_3)^2 + dr(r_2, r_3)^2) . \quad (5.1)$$

Figure 5.1: **The tetrahedron.**

Assuming the rest length of each spring is 1, geometry determines the equilibrium positions of the vertices. Using this result we compute \mathbb{H} using the familiar equation:

$$\mathbb{H} = \frac{\partial^2 V}{\partial q_j \partial q_k} \Big|_0 =$$

| | | | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0.5 | 0 | 0 | -0.25 | 0.14 | 0.41 | 0 | 0 | 0 | -0.25 | -0.14 | -0.41 |
| 0 | 0.5 | 0 | 0.14 | -0.08 | -0.24 | 0 | -0.33 | 0.47 | -0.14 | -0.08 | -0.24 |
| 0 | 0 | 2. | 0.41 | -0.24 | -0.67 | 0 | 0.47 | -0.67 | -0.41 | -0.24 | -0.67 |
| -0.25 | 0.14 | 0.41 | 1.5 | -0.58 | -0.41 | -0.25 | 0.43 | 0 | -1. | 0 | 0 |
| 0.14 | -0.08 | -0.24 | -0.58 | 0.83 | 0.24 | 0.43 | -0.75 | 0 | 0 | 0 | 0 |
| 0.41 | -0.24 | -0.67 | -0.41 | 0.24 | 0.67 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | -0.25 | 0.43 | 0 | 0.5 | 0 | 0 | -0.25 | -0.43 | 0 |
| 0 | -0.33 | 0.47 | 0.43 | -0.75 | 0 | 0 | 1.83 | -0.47 | -0.43 | -0.75 | 0 |
| 0 | 0.47 | -0.67 | 0 | 0 | 0 | 0 | -0.47 | 0.67 | 0 | 0 | 0 |
| -0.25 | -0.14 | -0.41 | -1. | 0 | 0 | -0.25 | -0.43 | 0 | 1.5 | 0.58 | 0.41 |
| -0.14 | -0.08 | -0.24 | 0 | 0 | 0 | -0.43 | -0.75 | 0 | 0.58 | 0.83 | 0.24 |
| -0.41 | -0.24 | -0.67 | 0 | 0 | 0 | 0 | 0 | 0 | 0.41 | 0.24 | 0.67 |

(5.2)

where we have rounded each element of the matrix to the hundredths place. The normal frequencies corresponding to this matrix are

$$\omega = \begin{pmatrix} 2 \\ 1.41421 \\ 1.41421 \\ 1.41421 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (5.3)$$

As expected, there are six zero-frequency modes. The corresponding eigenvectors are not shown here, although they can be found in the *Mathematica* section of the appendix. The general solution is formed by a linear combination of these modes:

$$q_i(t) = \sum_r \alpha_r a_{ir} \cos(\omega_r t) \quad (5.4)$$

where i ranges from 0 to 11. We put in vector form with the equation

$$\mathbf{r}_j(t) \longrightarrow (q_{(3j)}(t), q_{(3j+1)}(t), q_{3j+2}(t)) \quad (5.5)$$

where j ranges from 0 to 4. Figure 5.2 attempts to illustrate how the tetrahedron behaves when each normal mode is excited. Note that for the tetrahedron, the so called "breathing" mode (ω_1) has the highest frequency. The triply degenerate normal modes (ω_2, ω_3 , and ω_4) have slightly lower frequency. Each of these modes exhibits quite different behavior. In ω_2 , two of the vertices stretch apart and then collapse back in. In ω_3 , the top of the tetrahedron squashes down pushing the 3 vertices at the base out before bouncing back to its initial position. In ω_4 , one vertex is pulled into the tetrahedron while another is pushed out followed by the reverse process.

5.2 The computational solution

Modeling the tetrahedron with our Runge-Kutta simulation requires only the force function. We already know the equilibrium positions of the points, and we will use the results of the analytical solutions to stimulate the desired modes. Since each vertex experiences a force due to the spring tension between itself and every other vertex, the force function is

$$\mathbf{F}(\mathbf{r}) = \begin{pmatrix} T(\mathbf{r}_0, \mathbf{r}_1) + T(\mathbf{r}_0, \mathbf{r}_2) + T(\mathbf{r}_0, \mathbf{r}_3) \\ T(\mathbf{r}_1, \mathbf{r}_0) + T(\mathbf{r}_1, \mathbf{r}_2) + T(\mathbf{r}_1, \mathbf{r}_3) \\ T(\mathbf{r}_2, \mathbf{r}_0) + T(\mathbf{r}_2, \mathbf{r}_1) + T(\mathbf{r}_2, \mathbf{r}_3) \\ T(\mathbf{r}_3, \mathbf{r}_0) + T(\mathbf{r}_3, \mathbf{r}_1) + T(\mathbf{r}_3, \mathbf{r}_2) \end{pmatrix} \quad (5.6)$$

where $T(\mathbf{r}_i, \mathbf{r}_j)$ is defined by equation 4.13. Entering this into *RK4* and picking α such that an equal portion of each normal mode is present in the motion yields figure 5.3. Both solutions agree. Nonlinear qualities of the tetrahedron can be observed by slowly increasing the amplitude of α (See figure 5.4). Notice that the originally stimulated frequency, $\omega_1 = 2$ radians/second, decreases in magnitude relative to a new frequency. This new frequency is consistent with ω_5 and ω_6 having a value of 1 radian/second. Thus it is a demonstration of a nonlinear phenomenon known as *energy transfer*. This example illustrates the computational method's ability to explore nonlinear ranges of motion, a subject that will return with our consideration of the clathrin cage. In preparation of this task, however, we turn to a simple system which exhibits semi-rigid characteristics: the cube.

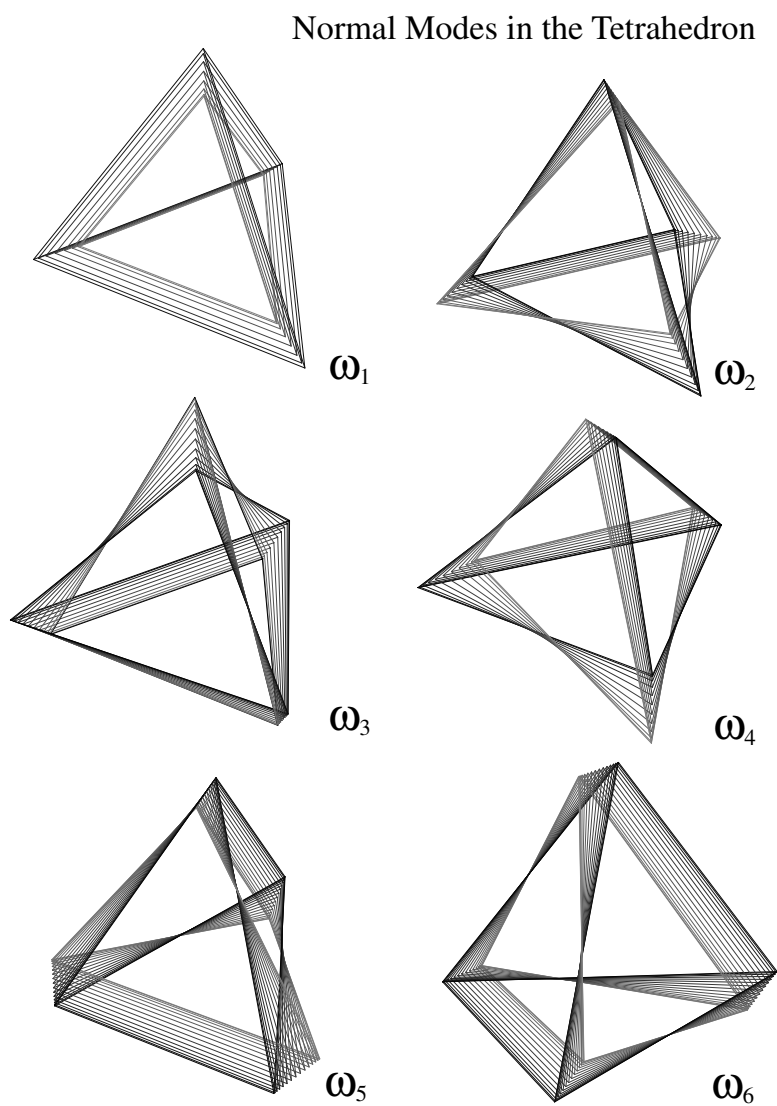


Figure 5.2: The normal modes of a tetrahedron with vertices connected by springs. Snapshots are taken at .2 second intervals and fade in to black at the final time $\frac{T}{2} = \frac{\pi}{\omega_r}$.

Normal modes of the Tetrahedron

$h=.01; t_f=1000; n=20000;$
 $\alpha=\{.001, .001, .001, .001, .001, .001, 0, 0, 0, 0, 0, 0\}$

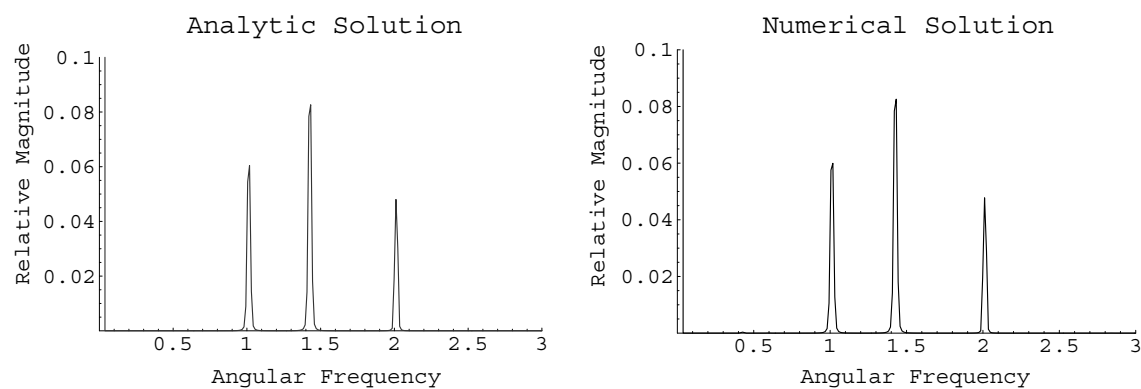


Figure 5.3: All the normal modes in the tetrahedron are stimulated. The analytic and computational solutions agree. Normal frequency degeneracy increases the magnitude of the oscillation in the first two peaks.

Nonlinear Effects in the Tetrahedron

varying the initial amplitude of ω_1
 $h=.01$; $t_f=1000$; $n=20000$;

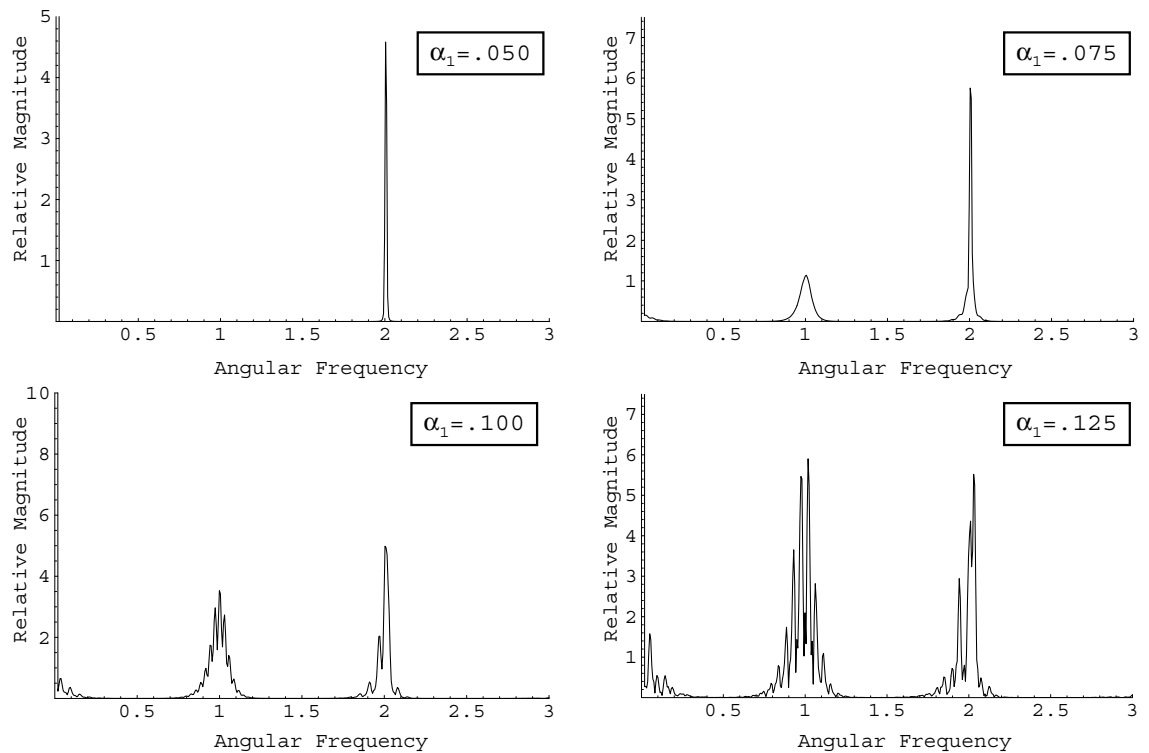


Figure 5.4: Nonlinearity in the tetrahedron can be seen by increasing the amplitude of the vibrations. Rather than simply increasing the magnitude of ω_1 , energy is transferred from the original frequency to the degenerate ω_5 and ω_6 frequency. Although the scale chosen here does not show it, there are probably higher frequency harmonics in this case as well.

Chapter 6

Semi-Rigid Systems

Vibrational analysis of the triangle and the tetrahedron demonstrate the equivalence and accuracy of both solutions for systems at equilibrium. But what happens when we apply these techniques to systems which do not have a well defined equilibrium position? We will show that in this case the analytic solution can still provide information about the vibrational modes, and in addition it can serve as a method of assessing the rigidity of a system.

6.1 The square

One simple system which does not have a single equilibrium position is the square with a spring along each edge. It is possible to deform this shape into a rhombus without stretching any of its sides (See figure 6.1). Thus, the potential energy remains at its minimum value for all values of \mathbf{r} that define a rhombus with edges of the same length as the original square. However, all other vertex configurations do change the potential energy of the system. What happens when we compute the eigenvectors and eigenvalues of \mathbb{H} in this case?

The potential energy is just the sum over each spring,

$$V = \frac{1}{2}k(dr(\mathbf{r}_0, \mathbf{r}_1)^2 + dr(\mathbf{r}_1, \mathbf{r}_2)^2 + dr(\mathbf{r}_2, \mathbf{r}_3)^2 + dr(\mathbf{r}_3, \mathbf{r}_0)^2) \quad (6.1)$$

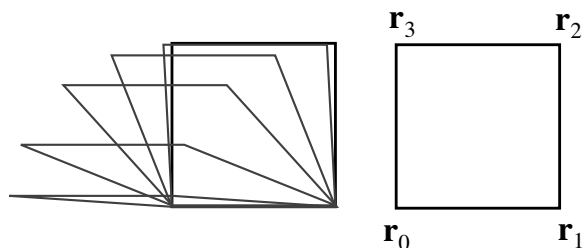


Figure 6.1: A square can deform into a rhombus without stretching any of its sides. In the vibrational treatment of the square, this characteristic appears in the form of a new, zero-frequency mode.

and we have selected the equilibrium length of each side to be 1. We compute \mathbb{H} :

$$\mathbb{H} = \left. \frac{\partial^2 V}{\partial q_j \partial q_k} \right|_{\text{equilibrium}} =$$

$$\begin{pmatrix} 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (6.2)$$

Taking the square root of the eigenvalues gives us the normal frequencies:

$$\omega = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1.41421 \\ 1.41421 \\ 1.41421 \\ 1.41421 \end{pmatrix} \quad (6.3)$$

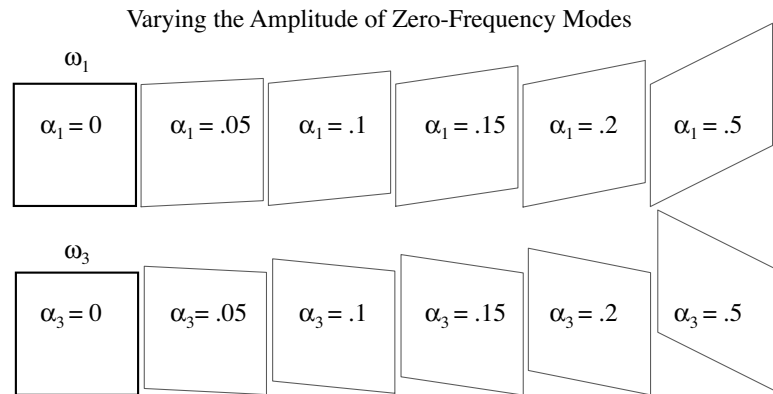


Figure 6.2: The general solution of vibration in the square is found by computing \mathbb{H} and looking at its eigenvectors and eigenvalues. The solution contains 4 modes in which the vertices translate with zero-frequency. Visualizing these modes shows that they correspond with a deformation of the square which does not stretch any springs for small displacements from equilibrium. A linear combination of the two modes shown would cause a translation in the y-direction.

and the eigenvectors are the rows of the following matrix:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad (6.4)$$

We generate the general solution in the usual way and choose appropriate values of α to examine each normal mode. Notice that instead of finding 3 zero-frequency modes, as we did in the triangle, there are 4. We increase the amplitude of several of these modes and observe the result. (See figure 6.2). Although we do not show all the zero-frequency modes, glancing at their associated eigenvectors reveals that there are no obvious translational or rotational modes. However, it is easy to make

the appropriate linear combinations of zero-frequency modes to describe translation and rotation in this case (See figure 6.2).

These results can be interpreted by remembering that the analytic solution is valid only for small displacements from equilibrium. Imagine taking the right edge of the square and pulling it in the y-direction. Although you would clearly experience a force when the right edge is pulled a good distance upwards, if you pull that edge up only an *infinitesimal* amount, it is less obvious how and if the horizontal springs would be stretched. Recalling that the Hessian matrix is merely the 2nd order term in the Taylor expansion of the potential energy, we realize that if there is a restoring force in this infinitesimal displacement, it must appear in the Taylor expansion as a 3rd order term or higher.

One ramification of this insight can be seen by modifying the previous example. Instead of holding the right edge and pulling it upwards, imagine pulling the top-right vertex up while pulling the bottom-right vertex down. If the previous analysis is correct, the only force either of these vertices experience would be due to the tension of the spring connecting them. Looking back at ω and visualizing the solution shows that this is indeed true. The frequency we expect for two masses connected by a spring is easily found from Newtons 2nd law without using the machinery we have developed thus far. The tension in the spring is given by Hooke's law:

$$T = -k(x - x_1)$$

Because the center of mass must not move, we have

$$x_1 = -x$$

Vibration in the square

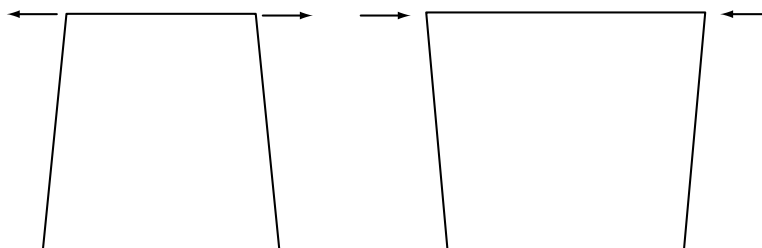


Figure 6.3: The general solution of vibration in the square is found by computing \mathbb{H} and looking at its eigenvectors and eigenvalues. The solution contains 4 vibrational modes in which each set of two vertices oscillate along their shared edge.

so

$$\begin{aligned}
 T &= -2kx \\
 \ddot{x} &= \frac{T}{m} = -\frac{2k}{m}x \\
 \implies \omega &= \sqrt{\frac{2k}{m}}
 \end{aligned} \tag{6.5}$$

where x_1 and x measure the displacement of each point from equilibrium along the axis of the spring connecting them. In our simulations, we have set

$$\begin{aligned}
 k &\rightarrow 1 \\
 m &\rightarrow 1 \\
 \implies \omega &= \sqrt{2} = 1.41421
 \end{aligned} \tag{6.6}$$

This is exactly the frequency of the four vibrational modes we found in the square. A quick look at the eigenvectors shows that each mode corresponds to oscillation between two vertices which share an edge. This is illustrated in figure 6.3.

So, applying the analytic solution to a system that is not at a well defined equilibrium is still quite revealing. It has informed us what vibrational modes can

exist in the structure, and considering the zero-frequency modes reveal the ways it can deform without experiencing 2nd order forces. Recall that for a rigid, nonlinear system in three-dimensions we expect the number of zero-frequency modes to be six (three corresponding to translation, and three corresponding to rotation). Our consideration of the square suggests that its worthwhile to introduce a new variable, η , which describes the rigidity of a system, where $\eta = 0$ implies a completely rigid system, and increasing values of η indicate a less stable structure. If we define ζ to be the total number of zero-frequency modes and f to be the expected number of translational and vibrational modes¹, then we propose a new equation:

$$\eta = \zeta - f \tag{6.7}$$

Or in words, the rigidity of the system is related to the number of zero-frequency modes minus the expected translational and rotational modes. To see if this equation effectively describes the rigidity of a system, we apply it to a cube.

6.2 The cross-braced cube

Previously, we saw that the square has four vibrational modes and four zero-frequency modes. To test equation 6.7, consider a similar but more complicated system: the cube. We can immediately make some conjectures about the expected solution. As the square example illustrates, moving the endpoint of a spring along a path perpendicular to the direction of the spring does not cause a 2nd order restoring force. Thus, in the cube it seems reasonable to expect at least 8 normal modes corresponding to the vibration of each edge. The goal is to start with this system and then add additional springs between vertices not normally connected. As we do this, we watch the rigidity index, η .

¹In general, this would be 6 for nonlinear systems, 5 for linear ones, or for two-dimensional cases like the triangle and the square, 3.

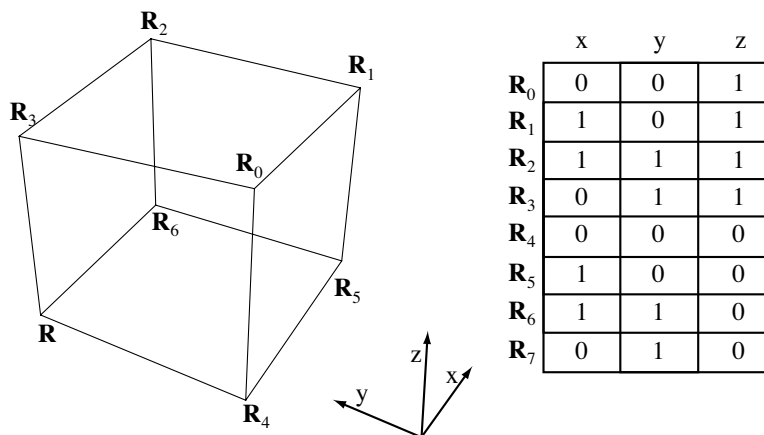


Figure 6.4: Labeling the cube vertices.

6.2.1 Determining the potential

The potential of a cube is easy to define with the help of figure 6.4. Since we will be adding springs to the system which vary in equilibrium length, it is important to include the appropriate rest length in each term of the potential energy. The function that determines the displacement of each spring is

$$dr(\mathbf{r}_i, \mathbf{r}_j) = |\mathbf{r}_i - \mathbf{r}_j| - d. \quad (6.8)$$

As usual, we set each edge to be of length 1. Geometry shows that for diagonal springs that cross through the center of the cube, $d = \sqrt{3}$, while for diagonal springs that lie on a face, $d = \sqrt{2}$. Rather than writing out the various potential functions here, the reader is referred to the *Mathematica* code in the appendix. The following values of ω are calculated by adding the energy terms from cross-bracing springs to the cube's potential function. In order to present the results more elegantly, we will use a series of figures which graphically show the included springs along with the system's normal frequencies and the index of rigidity, η (See figures 6.5 and 6.6). Notice that, as we expected, the unbraced cube has exactly 8 identical, non-zero, vibrational modes. The corresponding frequency is identical to that of two

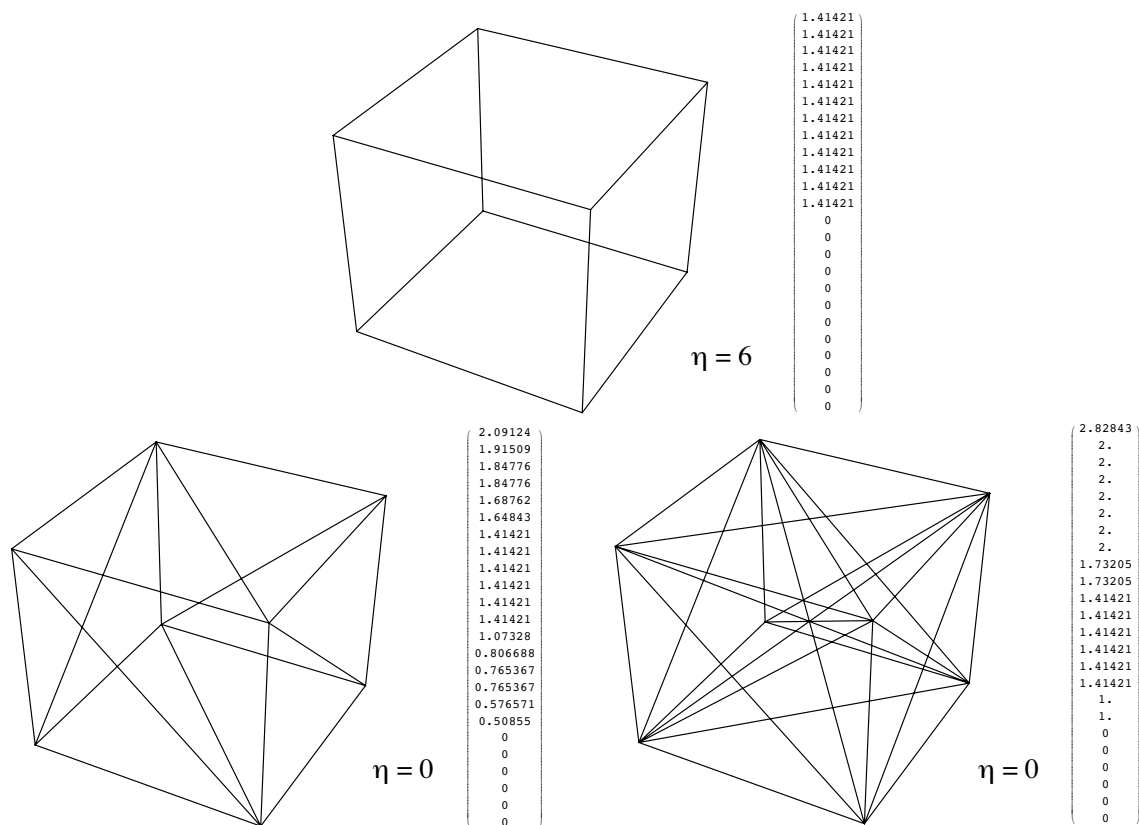


Figure 6.5: **Top:** the unbraced cube with normal frequencies and the index of rigidity: $\eta = \text{zero-frequencies} - 6$. **Bottom left:** adding one diagonal spring across each face reduces η to zero. **Bottom right:** Adding more springs, one between every pair of vertices, changes the normal frequencies but does not alter the number of zero-frequency modes.

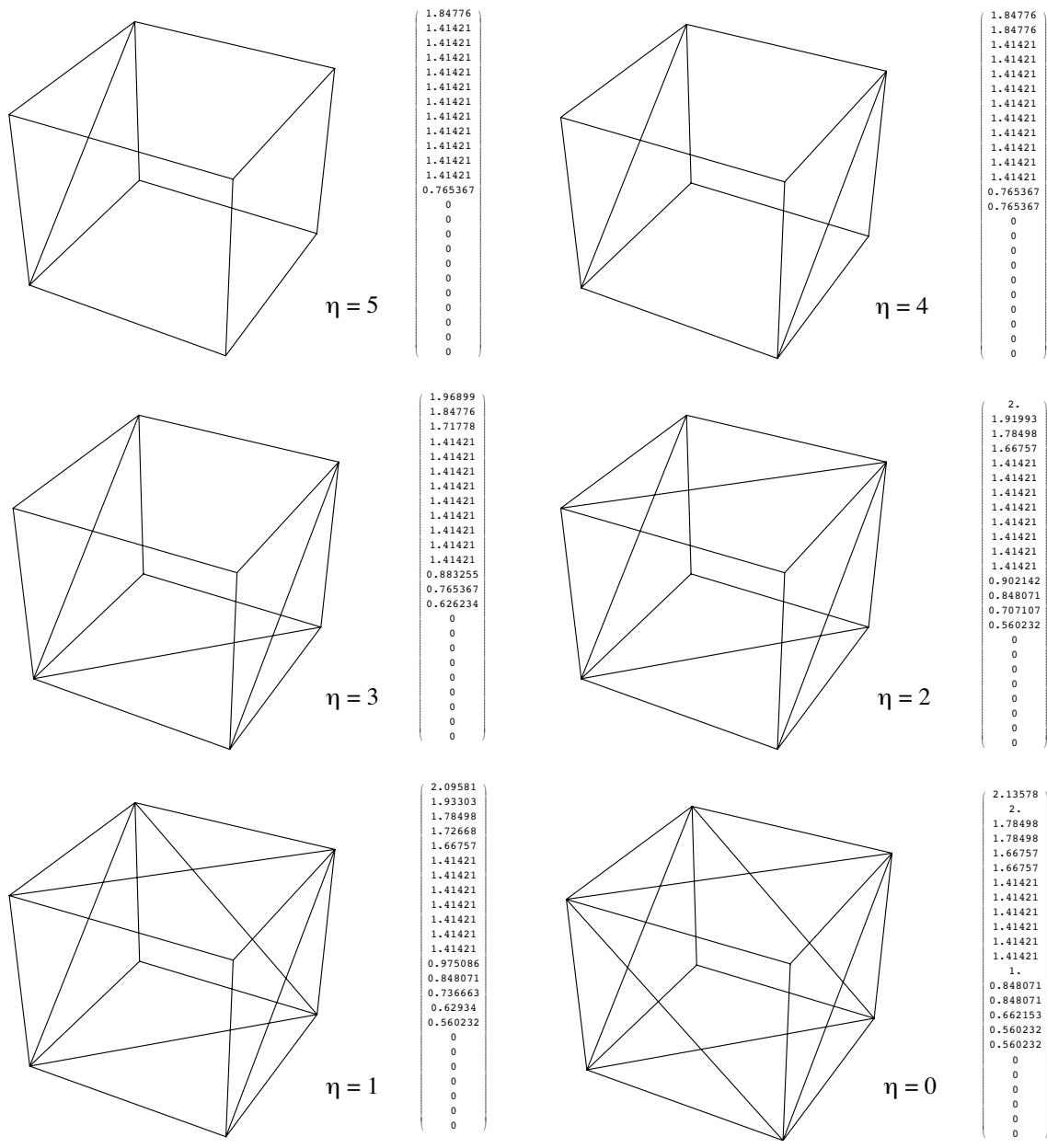


Figure 6.6: Adding diagonal springs to the cube affects the index of rigidity, η , in a methodical way.

masses connected by a single spring. So like the square, transverse restoring forces must be 2nd order or higher. Furthermore, the index of rigidity behaves in a useful and intuitively correct way. By trying various spring combinations and using the normal mode analysis we are able to quickly and easily determine which shapes are rigid, and which exhibit various degrees of floppiness. These results call for a more rigorous mathematical approach which will be touched upon in Appendix A. For now the focus remains vibration in the clathrin cage. The close look we have taken at semi-rigid systems will be vital in interpreting results obtained in the following chapter.

Chapter 7

Normal Modes of the Clathrin Cage

Recall that the clathrin cage is a large hexagonal barrel shape with rigid triskelions on each vertex. In order to use the machinery introduced in the previous chapters, we must simplify this complex biological protein into a set of point masses coupled by springs. It is hoped that as extreme as this simplification is, the results of a normal mode analysis will still yield useful results.

Applying the analytic and computational methods to this system is a test in patience and care, but the specifics of both approaches remain unchanged. Structurally, the clathrin cage has 36 vertices and is composed of 12 pentagons and 8 hexagons for a total of twenty faces. This means that we expect $3N = 108$ normal modes. In the analytic solution, the potential energy includes one term for every spring which works out to 54 terms. The Hessian matrix will have the dimensions 108×108 . This means that each eigenvector will also have 108 elements, and there will be 108 eigenvalues. Meanwhile, the computational solution requires the derivation of 36 equations which describe the force each vertex experiences (i.e. a 36 component force function). Due to the large size of these quantities, the reader is referred to the *Mathematica* code to see how they are generated.

7.1 Finding the equilibrium position

Unlike the previous cases, the equilibrium configuration of clathrin is not trivial, and its computation relies on a modification of the numerical methods previously discussed. We conjecture that since each vertex represents a triskelion in the physical system, the equilibrium distance between neighboring vertices must remain the same across the cage. From electron micrographs, we know how the various triskelions interconnect in the specific case we have chosen. These two facts motivate the use of our computational model to determine the minimum energy of the system. If we make a reasonable guess at the equilibrium position of each vertex and include a damping term in the force function, then running the simulation will relax the cage down to a configuration in which all the springs assume their rest length. Modifying the force function with a velocity dependent damping term yields:

$$F(\mathbf{r}, \mathbf{v}) = \begin{pmatrix} F(\mathbf{r}_1) = \sum_j T(\mathbf{r}_1, \mathbf{r}_j) - \gamma \mathbf{v}_1 \\ F(\mathbf{r}_2) = \sum_j T(\mathbf{r}_2, \mathbf{r}_j) - \gamma \mathbf{v}_2 \\ \vdots \\ F(\mathbf{r}_N) = \sum_j T(\mathbf{r}_N, \mathbf{r}_j) - \gamma \mathbf{v}_N \end{pmatrix} \quad (7.1)$$

where j ranges over the connected vertices, and γ is the damping coefficient.

We estimate the position of each particle by assuming that every vertex lies on a set of rings oriented in the x - y plane (See figure 7.1). Spherical coordinates are used to select 6 evenly spaced points on each ring. By varying each rings scale, rotation, and position along the z axis, we assemble a set of coordinates which roughly matches the symmetry of the true cage. Labeling these points in a careful way, (See figure 7.2), makes computing the force function a straightforward, though tedious, process. With these quantities known, we run the simulation and observe the clathrin cage settling to an equilibrium configuration (See figure 7.3). Note that we have chosen the usual rest length for each spring: $d = 1$. Our efforts result in table 7.1. With the equilibrium configuration of clathrin known, we can proceed

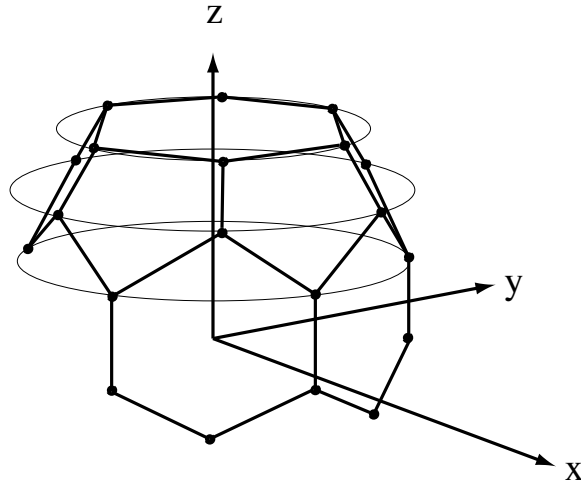


Figure 7.1: **We take a rough guess at the clathrin's equilibrium configuration by assuming the vertices lie on a set of rings oriented in the x-y plane. The bold lines show where the springs will connect in our simulation.**

with the usual analytic and computational methods.

7.2 Analytic results

Using the previous result, we compute the potential energy and the resulting Hessian matrix. Computing the normal frequencies reveals a surprise: there are exactly 54 vibrational modes and 54 zero-frequency modes in the clathrin cage (See table 7.2). Applying the formula for the index of rigidity here

$$\eta = (\text{zero-frequency modes}) - 6 = 48. \quad (7.2)$$

If this equation is at all indicative of the flexibility of a structure, then it appears our mass-spring model of the clathrin cage is far too great a simplification. What are we to make of the non-zero normal frequencies in light of this? We turn to the computational model for further insight.

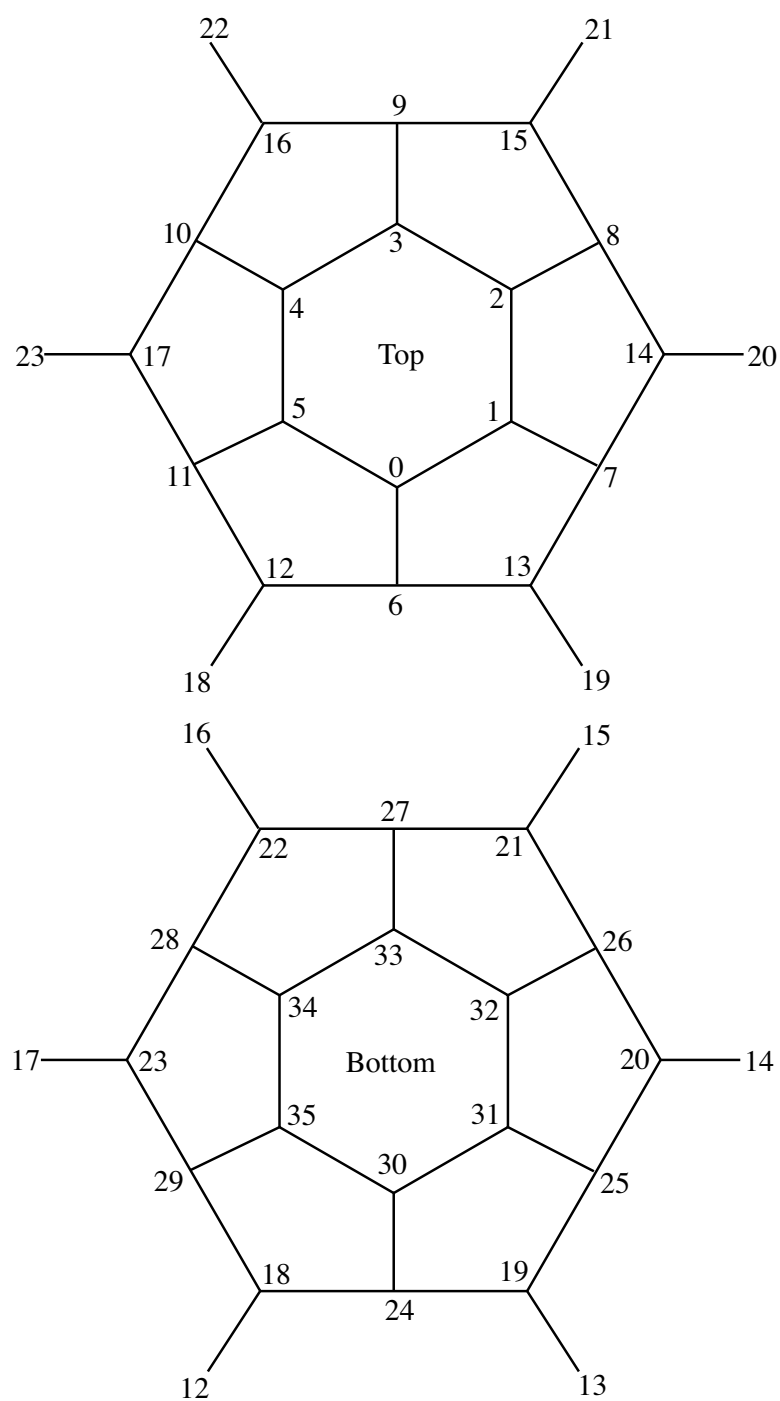


Figure 7.2: These diagrams demonstrate how the clathrin vertices are connected by springs, and is used in the construction of the force function. Each number represents the index, j , of the vector r_j .

Determining the clathrin equilibrium configuration

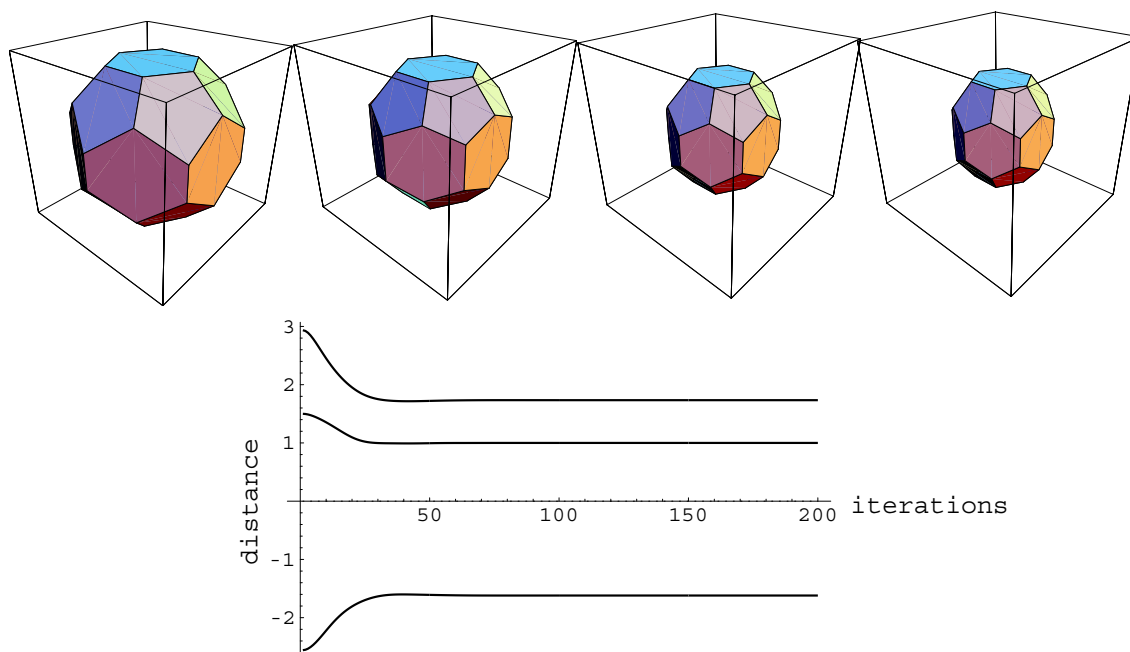


Figure 7.3: Taking a guess at clathrin's equilibrium position and including a damping term in its force function allows us to minimize the potential energy and find the true equilibrium position numerically. The plot shows values for a few q_i as they equilibrate.

| | x | y | z |
|-------|-----------|-----------|-----------|
| R[0] | 1. | 0 | 1.76901 |
| R[1] | 0.5 | 0.866025 | 1.76901 |
| R[2] | -0.5 | 0.866025 | 1.76901 |
| R[3] | -1. | 0 | 1.76901 |
| R[4] | -0.5 | -0.866025 | 1.76901 |
| R[5] | 0.5 | -0.866025 | 1.76901 |
| R[6] | 1.61969 | 0 | 0.984161 |
| R[7] | 0.809845 | 1.40269 | 0.984161 |
| R[8] | -0.809845 | 1.40269 | 0.984161 |
| R[9] | -1.61969 | 0 | 0.984161 |
| R[10] | -0.809845 | -1.40269 | 0.984161 |
| R[11] | 0.809845 | -1.40269 | 0.984161 |
| R[12] | 1.50165 | -0.866981 | 0.5 |
| R[13] | 1.50165 | 0.866981 | 0.5 |
| R[14] | 0 | 1.73396 | 0.5 |
| R[15] | -1.50165 | 0.866981 | 0.5 |
| R[16] | -1.50165 | -0.866981 | 0.5 |
| R[17] | 0 | -1.73396 | 0.5 |
| R[18] | 1.50165 | -0.866981 | -0.5 |
| R[19] | 1.50165 | 0.866981 | -0.5 |
| R[20] | 0 | 1.73396 | -0.5 |
| R[21] | -1.50165 | 0.866981 | -0.5 |
| R[22] | -1.50165 | -0.866981 | -0.5 |
| R[23] | 0 | -1.73396 | -0.5 |
| R[24] | 1.61969 | 0 | -0.984161 |
| R[25] | 0.809845 | 1.40269 | -0.984161 |
| R[26] | -0.809845 | 1.40269 | -0.984161 |
| R[27] | -1.61969 | 0 | -0.984161 |
| R[28] | -0.809845 | -1.40269 | -0.984161 |
| R[29] | 0.809845 | -1.40269 | -0.984161 |
| R[30] | 1. | 0 | -1.76901 |
| R[31] | 0.5 | 0.866025 | -1.76901 |
| R[32] | -0.5 | 0.866025 | -1.76901 |
| R[33] | -1. | 0 | -1.76901 |
| R[34] | -0.5 | -0.866025 | -1.76901 |
| R[35] | 0.5 | -0.866025 | -1.76901 |

Table 7.1: The equilibrium configuration of clathrin is found by including a damping term in the computational model and letting the system relax to its minimum energy. The various vectors, $R[i]$, correspond with figure 7.2

**Clathrin Normal Modes
Analytic Solution**

| | | | | | | | | |
|---------|---------|---------|----------|----------|---|---|---|---|
| 1.75182 | 1.64349 | 1.53251 | 1.20822 | 0.919395 | 0 | 0 | 0 | 0 |
| 1.7434 | 1.64349 | 1.50094 | 1.17654 | 0.882794 | 0 | 0 | 0 | 0 |
| 1.7434 | 1.64308 | 1.50094 | 1.17654 | 0.792697 | 0 | 0 | 0 | 0 |
| 1.73205 | 1.64308 | 1.4974 | 1.13439 | 0.736236 | 0 | 0 | 0 | 0 |
| 1.73205 | 1.62011 | 1.48041 | 1.13439 | 0.736236 | 0 | 0 | 0 | 0 |
| 1.71684 | 1.62011 | 1.47914 | 1.09444 | 0.63369 | 0 | 0 | 0 | 0 |
| 1.71684 | 1.56499 | 1.47914 | 1.09444 | 0 | 0 | 0 | 0 | 0 |
| 1.6778 | 1.56499 | 1.43628 | 1.06509 | 0 | 0 | 0 | 0 | 0 |
| 1.6778 | 1.56162 | 1.43628 | 1.0596 | 0 | 0 | 0 | 0 | 0 |
| 1.66189 | 1.56162 | 1.28946 | 0.972569 | 0 | 0 | 0 | 0 | 0 |
| 1.65263 | 1.55635 | 1.28946 | 0.972569 | 0 | 0 | 0 | 0 | 0 |
| 1.65263 | 1.53251 | 1.20822 | 0.919395 | 0 | 0 | 0 | 0 | 0 |

Table 7.2: Analytically computing the normal modes in a mass-spring model of the clathrin cage reveals 54 vibrational modes and 54 zero-frequency modes.

7.3 Computational results

The classical result provides a general solution for each q_i which, as usual, defines the simulation's initial conditions. It is possible to visualize either solution after stimulating any linear combination of the normal modes by adjusting the 108 component vector, α . Observing the two solutions provides information to help interpret the 54 zero-frequency modes. For very small oscillations, $\alpha < .01$, the Fourier transforms of both solutions are nearly identical (See figure 7.4). Therefore we conclude that the non-zero modes obtained from \mathbb{H} are experimentally confirmed to exist in the structure. However, increasing the amplitude of α does not preserve these stable oscillations. Instead of showing harmonic oscillation that is modified by the presence of nonlinear terms, as observed in the tetrahedron, larger amplitude vibrations do not preserve the geometry of the structure (See figure 7.5). Apparently the classical

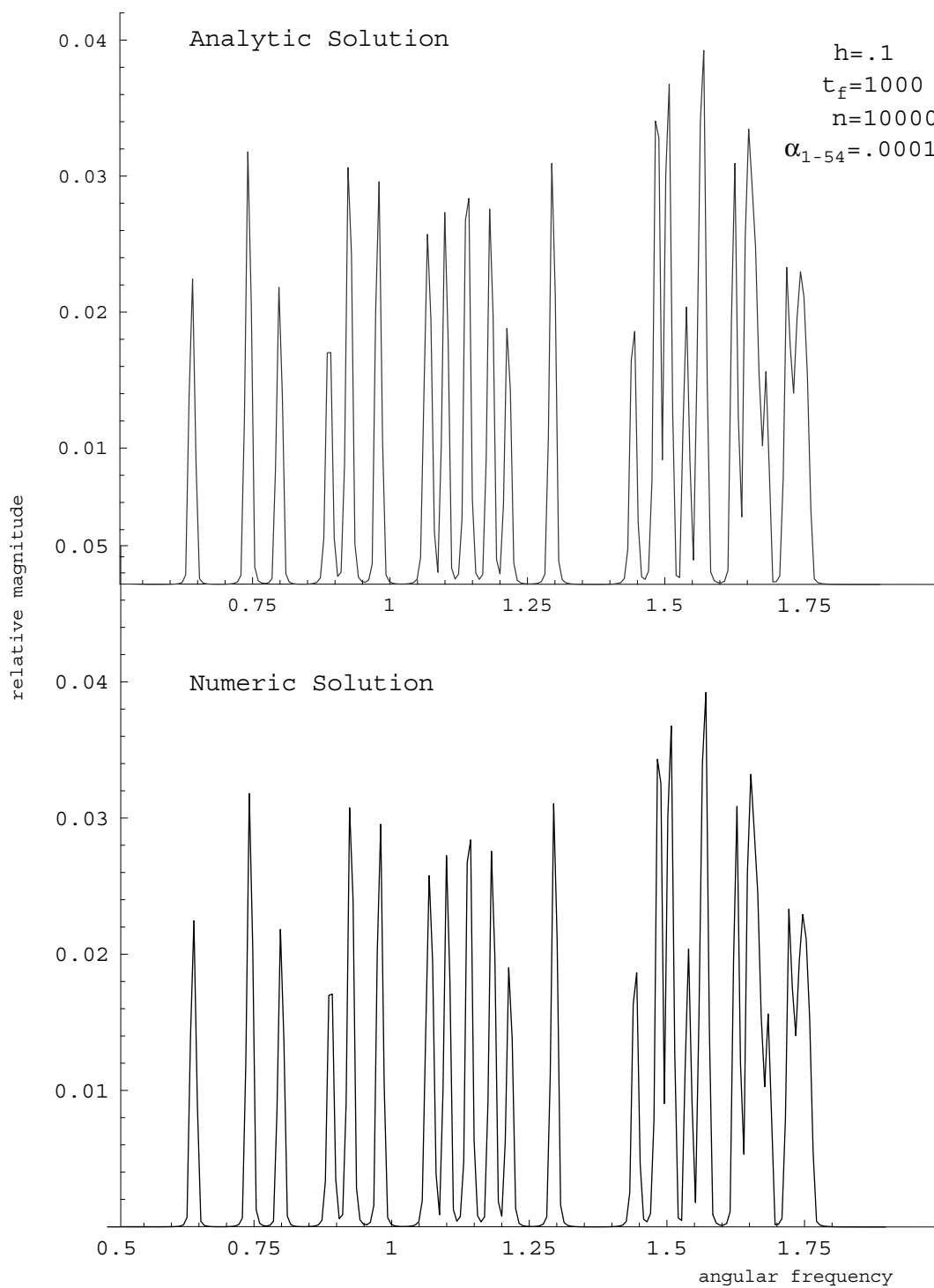


Figure 7.4: For very small oscillations, the analytic and numerical models agree. In these spectra, the 54 non-zero normal modes are equally excited with an amplitude .01% of the equilibrium distance.

Numerical solution for larger amplitudes

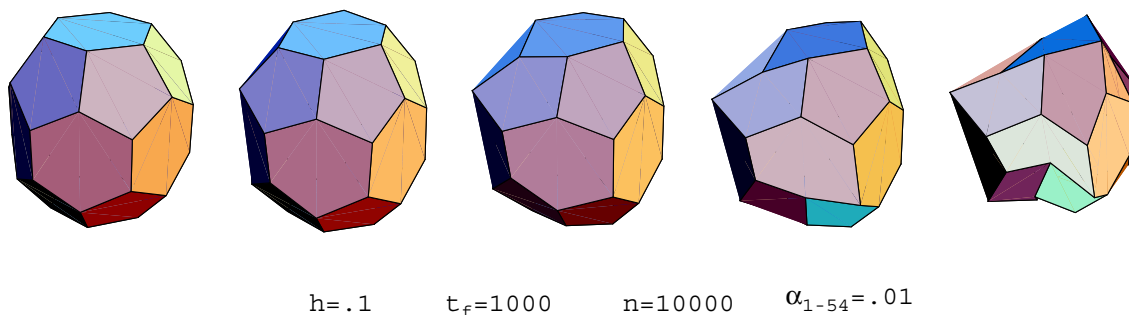


Figure 7.5: **Rather than simply introducing nonlinearities, increasing the amplitude of oscillation to 1% of each spring's rest length causes a complete destruction of the structure's symmetry.**

result is valid, but only for extremely small displacements from equilibrium. This suggests that η does assess, to some degree, the rigidity of a system. It is difficult to predict from the classical results alone how to interpret the 54 vibrational solutions. However, the numerical results are able to identify a regime in which the harmonic approximation holds. Increasing the amplitude to within 1% of each spring's rest length reveals that this harmonic approximation is not valid for the range of vibration we would like to consider. Thus our hypothesis that large amplitude oscillation plays a role in vesicular separation remains unanswered by this analysis. It seems that a mass-spring model of the clathrin cage ignores too many details of its structure. Future work may make the appropriate changes, such as the addition of torsion springs, to model the behavior more realistically.

Chapter 8

Conclusion

Although the normal modes of the clathrin cage were not definitively determined, the utility of both the classical and the numerical approach have been verified. What at first appeared to be anomalous factors of zero in ω have been shown to be an artifact of the unstable equilibrium that clathrin assumes when approximating rigid triskelions with point masses. Furthermore, this unexpected application of the analytic method may prove useful in assessing the rigidity of a system. One could further explore this possibility by combining both approaches and revisiting a simple structure like the cube. Chapter 6 illustrates the relationship between the index of rigidity, η , and the number of cross-bracing springs in place, but the next logical step has not yet been pursued. With the proven $RK4$ algorithm in place, we are already in a position where, by applying various loads to the cube, we could numerically compute its response. This would provide an experimental foundation for the intuitive notion that “more springs make a more rigid structure.” As much as the example of the cube demands more examination from a numerical standpoint, it also demands more from a mathematical one. We know that a large number of zero-frequency modes are caused by a lack of rigidity in the system. More precisely, the 2nd order terms that we retain from the Taylor expansion do not constrain the geometry of the object. Although it was not explored in earlier

chapters, varying the amplitude, α , of these zero-frequency modes provides insight into how the shape may be prone to deform. In the case of the clathrin cage, visualizing several of these deformation modes shows that the top and bottom of the clathrin cage can be pulled apart while the sides collapse in. This technique is hampered, however, by the fact that the zero-modes are a linear combination of translations, rotations, and deformations. A more useful approach would eliminate the translation and rotation modes before applying the vibrational analysis. This technique is discussed in Appendix A. A rigorous mathematical approach using these principles and focusing on semi-rigid systems could be of great benefit to the study of static systems. If the translational and rotational components are completely decoupled from the system prior to the normal mode analysis, then all the resulting zero-modes would have to be a linear combination of only deformations.

In addition to these untried approaches, the experimental data strongly hints at a fundamental relationship between the number of springs included in a system and the expected number of zero-frequency modes. Looking back at the cube and the clathrin cage, one notices that the zero-frequency modes correspond exactly with the number of degrees of freedom minus the number of included springs. Our model of clathrin contains 54 springs and has 108 degrees of freedom. It also has 54 zero-frequency modes. The same pattern is demonstrated by the cube model. The mathematical and experimental excursion necessary to answer this question is beyond the scope of the present work, but it cannot be ignored.

Other questions inspired by these results deal more directly with the originally posed problem of vesicular separation. For example, do all of the geometric variations assumed by clathrin exhibit semi-rigid behavior when modeled as point masses, and if so, do these semi-rigid characteristics play a role in vesicular separation? If the rigid geometry of the triskelions themselves are required for a stable structure, could the pinching off mechanism involve a relaxation of their torsional rigidity?

Although these new questions have been introduced, the original supposition that normal mode vibration plays a role in vesicular separation remains open to exploration. The machinery set forth in this text could be modified to include a torsional spring at each vertex. Research done on the triskelions themselves suggest that it may even be possible to obtain an experimental estimate of these torsional spring constants [3]. Coupling this with realistic estimates for the linear spring coefficients, the triskelion arm “rest length,” and an estimate of their mass would produce results which could be experimentally tested.

The work that has been accomplished is engaging for the questions it introduces as well as those it helps answer. The most obvious resolution reached is simply that mathematical rigor and numerical accuracy are both appropriate for the analysis of complex, many-body problems. Undoubtedly, pursuing these techniques to their logical conclusion will provide answers to the unexpected phenomena we have observed.

Appendix A:

Algebraic Considerations

In order to make the previous analysis of semi-rigid systems more rigorous, we would like to isolate those modes which correspond with deformations. Although it was not done previously, this is accomplished by using the appropriate equations of constraint to eliminate the translational and rotational modes of motion prior to the normal mode analysis. Wilson et al (see [11] p273-279) derive the appropriate equations of constraint to accomplish this. Three of these constrain the center of mass, and three constrain the rotation of the system. If m_j is the mass of each vertex, \mathbf{R}_j the equilibrium position of each vertex, and \mathbf{r}_j the generalized displacement of each vertex, these equations of constraint take the form

$$\sum_{j=0}^{N-1} m_j \mathbf{r}_j = 0 \quad (1)$$

and

$$\sum_{j=0}^{N-1} m_j \mathbf{R}_j \times \mathbf{r}_j = 0 \quad (2)$$

where \times denotes the cross product of the terms. Each of the above equations is actually 3 simultaneous equations, one for each dimension, so they can be used to eliminate 6 of the generalized coordinates. All zero-modes in the subsequent analysis would correspond with deformations of the structure, and furthermore these modes could be visualized as their magnitude, α , is adjusted. The resulting motion would

be decoupled from any translational or rotational modes, making their interpretation simpler and more revealing.

Another consequence of this is that the resulting Hessian, \mathbb{H} will reflect the reduction in 6 of the generalized coordinates by having dimensions $(3N - 6) \times (3N - 6)$. Recall that from Chapter 2, the equations of motion can be written in matrix form as

$$\mathbb{H}\mathbf{a} - \omega^2\mathbb{M}\mathbf{a} = 0. \quad (3)$$

We simplified this by letting the mass of each particle be 1, so that $\mathbb{M} \rightarrow \mathbb{I}$, the identity matrix. Alternatively, we could accomplish this by varying the individual masses and choosing a mass-weighted coordinate system (See [11] p14). Rewriting equation 3 in light of this, we have the familiar eigenvalue equation

$$\mathbb{H}\mathbf{a} = \omega^2\mathbf{a}. \quad (4)$$

For zero-frequency modes, ($\omega \rightarrow 0$), this becomes

$$\mathbb{H}\mathbf{a} = 0. \quad (5)$$

By definition, the vectors \mathbf{a} which solve this equation span the *null space* of \mathbb{H} , and the number of independent solutions correspond with the number of zero-frequency modes. Therefore, the rigidity of a system, η , is the dimension of the null space of \mathbb{H} after eliminating the translational and rotational modes using the above method. This provides one possible starting point for a more mathematically rigorous treatment of the deformation modes.

Appendix B: The Mathematica Code

Bibliography

- [1] C. J. Smith, N. Grigorieff and B. M.F.Pearse, “Clathrin coats at 21 Å resolution: a cellular assembly designed to recycle multiple membrane receptors,” *The EMBO Journal*, **17**: 4943–4953 (1998).
- [2] A. J. Jin and R. Nossal, “Topological Mechanisms Involved in the Formation of Clathrin-coated Vesicles,” *Biophysical Journal*, **65**: 1523–1537 (1993).
- [3] A. J. Jin and R. Nossal, “Rigidity of Triskelion Arms and Clathrin Nets,” *Biophysical Journal*, **78**: 1183–1194 (2000).
- [4] J. Powell and R. Crandall, “Lattice Dynamics for a Simplified DNA Model,” *Mathematica in Education 2*, **3**: 15–20 (1993).
- [5] R. Crandall, *Topics in Advanced Scientific Computation* (Springer-Verlag, New York, 1996), 215–222.
- [6] A. P. French, *Vibrations and Waves* (MIT, 1971), 136–141.
- [7] R. Landau and M. Paez, *Computational Physics* (John Wiley and Sons, New York, 1997), 117–129.
- [8] W. Press, S. Teukolsky, W. Vetterling and B. Flannery, *Numerical Recipes in C* (Cambridge University Press, 1992), 456–493 and 707–747.

- [9] H. Lodish, A. Berk, S. Zipursky, P. Matsudaira, D. Baltimore and J. Darnell, *Molecular Cell Biology* (W. H. Freeman and Company, 2000), 733–743.
- [10] J. B. Marion and S. T. Thornton, *Classical Dynamics of Particles and Systems* (Saunders College Publishing, 1995), 459–502.
- [11] E. Wilson jr, J. Decius, P. Cross, *Molecular Vibrations* (McGraw-Hill Book Company Inc., New York, 1955), 11–31, 54–74.
- [12] H. Goldstein, *Classical Mechanics* (Addison-Wesley Publishing Company, London, 1980), 243–274.